5 AL 9
DE OCTUBRE

XXI CONGRESO
ARGENTINO
DE CIENCIAS DE
LA COMPUTACIÓN

XIX ESCUELA
INTERNACIONAL
DE INFORMÁTICA

CACIC2015XXI

LIBRO DE ACTAS

Sede UNNOBA Junín

UNNOBA
Universidad Nacional
Noroeste ▪ Buenos Aires

RedUNCI

ESCUELA DE
TECNOLOGÍA

CEDI - Centro de Edición y Diseño

CACIC2015XXI

XXI CONGRESO ARGENTINO
DE CIENCIAS DE LA COMPUTACIÓN
CACIC 2015: OCTUBRE 2015, JUNÍN, ARGENTINA

Organizadores:
Red de Universidades con Carreras en Informática RedUNCI
Universidad Nacional del Noroeste de la Provincia de Buenos Aires
Escuela de Tecnología.

Fecha de Catalogación: 17/11/2015

Libro de Actas CACIC 2015 / Claudia Cecilia Russo ... [et al.]. - 1a ed . - Junín : Universidad Nacional del Noroeste de la Provincia de Buenos Aires. UNNOBA ; La Plata : RedUNCI, 2015.
Libro digital, PDF

Archivo Digital: descarga y online
ISBN 978-987-3724-37-4
1. Informática. 2. Educación. I. Russo, Claudia Cecilia
CDD 005.1

Autoridades de la RedUNCI

AUTORIDADES DE LA RED DE UNIVERSIDADES NACIONALES CON CARRERAS EN INFORMÁTICA

Coordinador Titular

De Giusti Armando (UNLP) 2014-2016

Coordinador Alterno

Feierherd Guillermo (UNTDF) 2014-2016

Junta Directiva

Horacio Kuna (UN Misiones) 2013-2015
Fernanda Carmona (UN Chilecito) 2013-2015
Jorge Finocchieto (U CAECE) 2013-2015
Raul Kantor (UN Rosario) 2013-2015
Estayno Marcelo (UNLZ) 2014-2016
Fabiana Piccoli (UNSL) 2014-2016
Dapozo Gladys (UNNE) 2014-2016
Padovani Hugo (UM) 2014-2016

SECRETARÍAS

Secretaría Administrativa

Prof. Javier Balladini (UNCOMA)

Secretaría Académica

Prof. Claudia Russo (UNNOBA)

Secretaría de Congresos, Publicaciones y Difusión

Prof. Patricia Pesado (UNLP)

Secretaría de Asuntos Reglamentarios

Prof. Osvaldo Spositto (UNLaM)

Autoridades de la UNNOBA

RECTORADO

Rector
Dr. Guillermo Tamarit
Vicerrectora
Mg. Danya V. Tavela
Guardasellos
Ing. Luis J. Lima

SECRETARÍAS

Secretaria General
Abog. Diego Batalla
Secretaria Académica
Abog. María Florencia Castro
Secretaría Económica Financiera
Cdora. Mariela García
Secretaría Legal y Técnica
Abog. Carlos Pérez
Secretaría de Extensión
Lic. Juan Pablo Itoiz
Secretaría de Investigación, Desarrollo y Transferencia
Dr. Jerónimo Ainchil
Secretaría de Cultura
Lic. Laura Durán
Instituto de Postgrado
Lic. Silvina Sansarricq

PROSECRETARÍAS

Prosecretaría General
Abog. Pablo G. Petraglia
Prosecretaría Económico Financiero
Cdor. Gustavo Bonet
Prosecretaría de Planeamiento y Coordinación Institucional
Cdora. Natalia Toloza
Prosecretaría de Tecnología de la Información y la Comunicación
Mg. Hugo Ramón

ESCUELA DE TECNOLOGÍA

Directora
Mg. Claudia Russo
Secretaría Académica
Lic. Mónica Sarobe
Prosecretaría Académica
Área de Diseño
DCV Florencia Antonini
Director del Departamento de Asignaturas Afines y Complementarias
DCV María de las Mercedes Ortín
Director del Departamento de Informática
Ing. Oscar Spada
Coordinador de Carreras de Ingeniería
Ing. Daniel Galli

Comité Organizador Local

Presidenta

Mg. Claudia Russo

Ahmad, Tamara
Anolles, Natalia
Di Cicco, Carlos
Gutierrez, Juan Pablo
Lencina, Paula
Mangold, Leonardo
Pérez, Daniela
Picco, María Linda Trinidad
Pompei, Sabrina
Rodríguez, Sabina
Serrano, Eliana
Spada, Oscar

XIX ESCUELA INTERNACIONAL DE INFORMÁTICA

Directora

Lic. Mónica Sarobe

Comité Académico

INTEGRANTES DEL COMITÉ ACADÉMICO, REPRESENTANTES DE LAS UNIVERSIDADES MIEMBRO DE LA RED UNCI

Escuela Superior Técnica, Facultad de Ingeniería del Ejército
Castro Lechstaler, Antonio

Universidad Abierta Interamericana
De Vicenzi, Marcelo

Universidad Adventista del Plata
Bournissen, Juan

Universidad Argentina John F. Kennedy
Panizzi, Marisa Daniela

Universidad Atlántida Argentina
Rathmann, Liliana

Universidad Austral
Robiolo, Gabriela

Universidad Autónoma de Entre Ríos
Aranguren, Silvia

Universidad CAECE
Finocchieto, Jorge

Universidad Católica Argentina
Grieco, Sebastián

Universidad de Buenos Aires
Echeverría, Adriana (Ingeniería)
Fernández Slezak, Diego (Exactas)

Universidad de Morón
Padovani, Hugo René

Universidad de Palermo
Cuckierman, Uriel

Universidad del Aconcagua
Giménez, Rosana

Universidad del CEMA
Guglianone, Ariadna

Universidad del Salvador
Zanitti, Marcelo

Universidad Gastón Dachary
Belloni, Edgardo

Universidad Nacional Arturo Jauretche
Morales, Martin

Universidad Nacional de Catamarca
Barrera, María Alejandra

Universidad Nacional de Chilecito
Carmona, Fernanda

Universidad Nacional de Córdoba
Brandán Briones, Laura

Universidad Nacional de Cuyo
Forradelas, Raymundo

Universidad Nacional de Entre Ríos
Tugnarelli, Mónica

Universidad Nacional de Jujuy
Paganini, José

Universidad Nacional de La Matanza
Spositto, Osvaldo Mario

Universidad Nacional de La Pampa
Alfonso, Hugo

Universidad Nacional de la Patagonia Austral
Márquez, María Eugenia

Comité Académico

Universidad Nacional de la Patagonia San Juan Bosco
Buckle, Carlos

Universidad Nacional de La Plata
De Giusti, Armando

Universidad Nacional de La Rioja
Nadal, Claudio

Universidad Nacional de Lanús
García Martínez, Ramón

Universidad Nacional de Lomas de Zamora
Estayno, Marcelo

Universidad Nacional de Lujan
Scucimarri, Jorge

Universidad Nacional de Misiones
Kuna, Horacio

Universidad Nacional de Río Cuarto
Arroyo, Marcelo

Universidad Nacional de Río Negro
Vivas, Luis

Universidad Nacional de Rosario
Kantor, Raúl

Universidad Nacional de Salta
Gil, Gustavo

Universidad Nacional de San Juan
Otazú, Alejandra

Universidad Nacional de San Luis
Esquivel, Susana

Universidad Nacional de Santiago del Estero
Durán, Elena

Universidad Nacional de Tierra del Fuego, Antártida e Islas del Atlántico Sur
Feierherd, Guillermo

Universidad Nacional de Tres de Febrero
Oliveros, Alejandro

Universidad Nacional de Tucumán
Luccioni, Griselda

Universidad Nacional de Villa María
Prato, Laura

Universidad Nacional del Centro de la Provincia de Buenos Aires
Acosta, Nelson

Universidad Nacional del Chaco Austral
Zachman, Patricia

Universidad Nacional del Comahue
Vaucheret, Claudio

Universidad Nacional del Litoral
Loyarte, Horacio

Universidad Nacional del Nordeste
Greiner, Cristina

Universidad Nacional del Noroeste de la Provincia de Buenos Aires
Russo, Claudia

Universidad Nacional del Oeste
Foti, Antonio

Universidad Nacional del Sur
Rueda, Sonia

Universidad Tecnológica Nacional
Bursztyn, Andrés

Comité Científico

COMITÉ CIENTÍFICO

Abásolo, María José *(Argentina)*
Acosta, Nelson *(Argentina)*
Aguirre, Jorge Ramió *(España)*
Alfonso, Hugo *(Argentina)*
Ardenghi, Jorge *(Argentina)*
Arroyo, Marcelo *(Argentina)*
Baldasarri, Sandra *(España)*
Balladini, Javier *(Argentina)*
Bertone, Rodolfo *(Argentina)*
Bría, Oscar *(Argentina)*
Brisaboa, Nieves *(España)*
Buckle, Carlos *(Argentina)*
Cañas, Alberto *(EE.UU)*
Casali, Ana *(Argentina)*
Castro Lechtaller, Antonio *(Argentina)*
Castro, Silvia *(Argentina)*
Cechich, Alejandra *(Argentina)*
Chavez, Edgar *(México)*
Coello Coello, Carlos *(México)*
Constantini, Roberto *(Argentina)*
Cuevas, Alfredo Simón *(Cuba)*
Cukierman, Uriel *(Argentina)*
De Giusti, Armando *(Argentina)*
De Giusti, Laura *(Argentina)*
De Vincenzi, Marcelo *(Argentina)*
Deco, Claudia *(Argentina)*
Depetris, Beatriz *(Argentina)*
Díaz, Javier *(Argentina)*
Dix, Juerguen *(Alemania)*
Doallo, Ramón *(España)*
Docampo, Domingo *(España)*
Echaiz, Javier *(Argentina)*
Esquivel, Susana *(Argentina)*
Estayno, Marcelo *(Argentina)*
Estevez, Elsa *(Argentina)*
Falappa, Marcelo *(Argentina)*
Feierherd, Guillermo *(Argentina)*
Fillottrani, Pablo *(Argentina)*
Finocchietto, Jorge *(Argentina)*
Fleischman, William *(Argentina)*
García Garino, Carlos *(Argentina)*
García Villalba, Javier *(España)*
Género, Marcela *(España)*
Giacomantone, Javier *(Argentina)*
Gómez, Sergio *(Argentina)*
Gröller, Eduard *(Austria)*
Guerrero, Roberto *(Argentina)*
Janowski, Tomasz *(Naciones Unidas)*
Kantor,Raul *(Argentina)*
Kuna, Horacio *(Argentina)*
Lanzarini, Laura *(Argentina)*
Leguizamón, Guillermo *(Argentina)*
López Gil, Fernando *(España)*
Loui, Ronald Prescott *(EEUU)*
Luque, Emilio *(España)*
Madoz, Cristina *(Argentina)*
Malberti, Alejandra *(Argentina)*

Comité Científico

Malbran,María *(Argentina)*
Manresa Yee, Cristina *(España)*
Marín, Mauricio *(Chile)*
Mas Sansó, Ramón *(España)*
Mon, Alicia *(Argentina)*
Motz, Regina *(Uruguay)*
Naiouf, Marcelo *(Argentina)*
Navarro Martín, Antonio *(España)*
Olivas Varela, José Ángel *(España)*

Actas del XXI Congreso Argentino de Ciencias de la Computación CACIC 2015

XVI Workshop Agentes
y Sistemas Inteligentes (WASI)

XV Workshop Procesamiento
Distribuido y Paralelo (WPDP)

XIII Workshop Computación Gráfica,
Imágenes y Visualización (WCGIV)

XIII Workshop Tecnología Informática
aplicada en Educación (WTIAE)

XII Workshop Ingeniería de Software (WIS)

XII Workshop Bases de Datos
y Minería de datos (WBDDM)

X Workshop Arquitectura,
Redes y Sistemas Operativos (WARSO)

VII Workshop Innovación
en Sistemas de Software (WISS)

VI Workshop Procesamiento de Señales
y Sistemas de Tiempo Real (WPSTR)

IV Workshop de Seguridad Informática (WSI)

IV Workshop de Innovación
en Educación en Informática (WIEI)

V ETHICOMP Latinoamérica

XVI Workshop Agentes y Sistemas Inteligentes (WASI)

7432 | Clasificación automática de evoluciones médicas multiclases en español
Marcela Riccillo (HIBA), David Perez (HIBA), Daniel Luna (HIBA), Fernando Campos (HIBA), Carlos Otero (HIBA), María Laura Gambarte (HIBA), Sonia Benítez (HIBA)

7445 | Representing Traffic Congestions on Moving Objects Trajectories
Mariano S. Kohan S. (UBA), Juan María Ale (UBA)

7450 | Immune Algorithm for Solving the Smooth Economic Dispatch Problem
Victoria Aragón (UNSL), Susana Esquivel (UNSL)

7520 | Predicción de Ingresos de Causas Penales utilizando Programación Genética Lineal
Alberto David Garcete Rodríguez (UNE), Benjamín Barán (UNE)

7528 | Optimización Basada en Colonias de Hormigas: Una aplicación a la distribución de sobres
Adrián Ojeda (UNI), Benjamín Barán (UNE), Horacio Kuna (UNaM)

7530 | Personalização de recursos com suporte à privacidade através de técnicas adaptativas
Paulo Roberto Massa (POLI-USB), João Jose Neto (POLI-USB)

7539 | Optimización basada en Colonias de Hormigas para el Problema del Vendedor Viajante con muchos objetivos contradictorios
Francisco Riveros (UNA), Nestor Benítez (UNA), Julio Paciello (UNA), Benjamín Barán (UNA)

7543 | A Preliminary Framework for Reasoning with Inconsistent Possibilistic Description Logics Ontologies with Disjunctive Assertions
Sergio Alejandro Gómez (UNS)

7574 | Imperialist Competitive Algorithm for the Flowshop Problem
Gabriela Minetti (UNLPAM), Carolina Salto (UNLPAM)

7611 | Integración de agentes inteligentes heterogéneos en el entorno T-World
Sergio Burdisso (UNSL), Guillermo Aguirre (UNSL), Marcelo Errecalde (UNSL)

7621 | A Desiderata for Modeling and Reasoning with Social Knowledge
Fabio R. Gallo (UNS), Natalia Abad (UNS), Gerardo I. Simari (UNS), Marcelo A. Falappa (UNS)

7634 | Algoritmos de Inteligencia de Enjambres Orientados a Map Reduce
Silvia Marta Molina (UNSL), Guillermo Leguizamón (UNSL)

7646 | Evaluación de dos nuevos algoritmos en el diseño de granjas eólicas
Fabricio Loor (UNSL), Guillermo Leguizamón (UNSL), Javier Apolloni (UNSL)

7658 | Clasificación de configuraciones de manos del Lenguaje de Señas Argentino con ProbSOM
Franco Ronchetti (UNLP), Facundo Quiroga (UNLP), César Estrebou (UNLP), Laura Lanzarini (UNLP)

Clasificación automática de evoluciones médicas multiclases en español

Marcela Riccillo[1], David Perez[1], Daniel Luna[1], Fernando Campos[1], Carlos Otero[1], María Laura Gambarte[1], Sonia Benítez[1]

[1]Departamento de Informática en Salud, Hospital Italiano de Buenos Aires HIBA
mail: marcela.riccillo@hospitalitaliano.org.ar

Abstract. En este trabajo presentamos una comparación de metodologías de clasificación para texto libre de narrativas médicas, en este caso evoluciones médicas multiclase. Comparamos el rendimiento de redes neuronales y máquinas de soporte vectorial con preprocesamientos para clasificar evoluciones de Diabetes, en Tipo 1, Tipo 2 y Otros (otro tipo de afección). Se compararon accuracy, sensitivity y specificity, mostrando beneficios en costos de entrenamiento y resultados de exactitud. Encontramos porcentajes mayores con redes neuronales sin preprocesamiento PCA y en el caso de SVM con dicho preprocesamiento (con menor costo de entrenamiento).

Keywords: Electronic Health Record, Support Vector Machine, Natural Language Processing, Neural Network, Diabetes Mellitus, Narrative Medicine, Principal Component Analyses

1 Introducción

En el Hospital Italiano de Buenos Aires [1], desde el año 1998 se ha implementado de manera gradual un Sistema de Información en Salud (SIS) a partir de un desarrollo "in house" que maneja la información médica y administrativa desde la captura hasta el análisis. Incluye una única Historia Clínica Electrónica (HCE) web, modular, orientada a problemas y centrada en el paciente. Conocida con el nombre de ITALICA, la HCE permite el registro de la atención en los ámbitos: ambulatorio, internación, emergencias y atención domiciliaria. ITALICA permite la solicitud de estudios complementarios, prescripción farmacológica y visualización de resultados que incluye un sistema de almacenamiento y transmisión de imágenes asociadas al paciente.

Dado que la HCE es orientada a problemas, cuenta con una lista estructurada de los problemas de los pacientes que posee un control terminológico [2] que normaliza el texto ingresado según ontologías de referencia. Luego, el médico evoluciona la situación de cada paciente asociando dicha evolución al problema correspondiente. La evolución es un campo que contiene textos libres de narrativa médica.

En la narrativa médica encontramos información valiosa sobre las características de cada paciente, evolución, medicamentos, antecedentes familiares, entre otros, que podría ser aprovechada para complementar y en algunos casos hasta para cotejar los

datos que aparecen en forma estructurada. Esto comprende por ejemplo [3] informes radiológicos, resúmenes de alta, reportes de patologías, historias de admisión, informes de exámenes físicos.

Los textos libres en general son complejos con datos de laboratorio (donde algunos son copiados de otros campos para enfatizar o ilustrar las indicaciones), comportamientos del paciente, sugerencias de tratamientos, interconsultas, entre otros. La complejidad lingüística dificulta la extracción automática de información, que luego podría utilizarse para reportes estadísticos o la correcta determinación de la patología presentada por el paciente.

El Procesamiento del Lenguaje Natural (NLP) [4] proporciona un medio para "desbloquear" esta importante fuente de datos, convirtiendo texto no estructurado en estructurado, en datos procesables para su uso en aplicaciones de soporte a las decisiones clínicas, control de calidad y monitoreo de la salud pública. Hay que tener en cuenta que los conceptos [5] pueden ser modificados por la negación (ej, "sin temblores significativos"), por la información temporal (ej, "admitido previamente por neumonía"), por el historial familiar (ej, "antecedentes familiares de enfermedades del corazón"), o modificadores que indican que el evento en realidad no ocurrió (ej, "expuesto a tuberculosis").

Es por eso que en este trabajo utilizamos técnicas que tomen en cuenta el contexto de las narrativas más que palabras específicas, es decir métodos de aprendizaje automático como Redes Neuronales Artificiales (ANN) y Máquinas de Soporte Vectorial (SVM). El objetivo del estudio es la comparación de metodologías de clasificación para el análisis multiclase de evoluciones médicas. Para esto aplicamos las metodologías a evoluciones en las cuales identificamos si corresponden a Diabetes Tipo 1, Diabetes Tipo 2 o no se relacionan con esta afección.

Esto podría servir para levantar alarmas en el caso de que el problema asociado a la evolución no sea el correspondiente o cuando las características demarcaran una posible ocurrencia de un problema que no esté registrado en la HCE del paciente.

2 Antecedentes

No sólo es importante la detección temprana de la enfermedad sino también identificar correctamente el tipo que corresponde, ya que los tratamientos pueden diferir según la sintomatología presentada.

La Diabetes Mellitus [6] es un desorden metabólico crónico caracterizado por niveles persistentemente elevados de glucosa en la sangre, como consecuencia de una alteración en la secreción y/o acción de la insulina. La Diabetes Mellitus tipo 1 (DBT1) se caracteriza por la destrucción de las células beta pancreáticas, que se traduce en un déficit absoluto de insulina y dependencia vital a la insulina exógena. La Diabetes Mellitus tipo 2 (DBT2) se caracteriza por resistencia insulínica, que habitualmente se acompaña de un déficit relativo de insulina.

Algunos trabajos analizan la detección o predicción de la enfermedad a través de variables estructuradas. Por ejemplo en el trabajo de Hera et al. [7] alimentan un modelo estadístico de regresión logística con variables como el nivel de glucemia para la detección de diabetes desconocida en pacientes coronarios. Khan et al. [8]

entrenan una red neuronal para el análisis de enfermedades renales a partir de la ocurrencia de síntomas (temperatura, sensación de náusea, dolores) para facilitar el diagnóstico del médico. En el estudio de Vassy et al. [9] utilizan variantes genómicas para la identificación de Diabetes Tipo 2 y en el de Yu et al. [10] aplican datos estructurados de contexto del paciente (como edad, peso, historial clínico) para detectar diabetes y pre-diabetes con la aplicación de técnicas de máquinas de soporte vectorial.

Sin embargo, en nuestro trabajo buscamos la detección de diabetes a partir de la información de contexto presente en la narrativa médica de las HCE. La narrativa médica siempre ha sido una parte vital de la medicina [11] y se puede entender como el puente entre la evidencia de estudios a gran escala y el arte de la medicina de la aplicación de este conocimiento al caso particular de cada paciente.

Existen varios trabajos donde se utilizan técnicas de procesamiento del lenguaje natural para la extracción de información en texto libre. Por ejemplo, en el de Chau, Xu y Chen [12] donde desarrollan un extractor basado en redes neuronales para diferenciar entidades en reportes policiales.

Pero si nos enfocamos en los trabajos con narrativas médicas, vemos por ejemplo que Breydo, Chu y Turchin [13] desarrollan un algoritmo para la detección de medicamentos inactivos para removerlos de la lista de las HCE. En Savoya et al. de Clínica Mayo [14] desarrollan un sistema llamado cTakes para Análisis de Texto y extracción del conocimiento, a partir de anotaciones semánticas de diversas estructuras de las oraciones que integran la narrativa.

Deng y Denecke [15] plantean la dificultad de los médicos de encontrar rápidamente la información del paciente siendo que la misma se encuentra distribuida en grandes cantidades de narrativa que se va incrementando a lo largo de su evolución. Para eso presentan una metodología de sumarización de la HCE a través de extracción de información, sentimental analysis, nubes de palabras y tecnologías aplicadas en creación de resúmenes.

Pero en el presente estudio no nos enfocamos en la extracción sino en la clasificación de documentos. En particular, clasificaciones de evoluciones médicas según si se relacionan con DBT Tipo 1, DBT Tipo 2 o representan casos de otras afecciones.

En el trabajo de Rodríguez, Calot y Merlino [16] el objetivo es la clasificación de texto médico en español. Los campos a clasificar son los de prescripciones médicas, que resultan campos de una oración o segmento de ésta. Para la representación de los tokens usan unigramas. Para resolver la clasificación comparan las metodologías de SVM y Naïve Bayes Multinomial.

En el trabajo de Wright et al. [17] categorizan notas de texto libre médico con SVM. Los conjuntos de entrenamiento y testeo fueron armados a partir de evoluciones médicas relacionadas o no con diabetes. La diferencia con nuestro estudio es que la clasificación que presentan es biclase, ya que entrenan la SVM para detectar Diabetes (cualquiera sea el tipo) o no.

3 Metodología

Nuestra metodología consistió en la comparación del entrenamiento de una Red Neuronal y una SVM multiclase para la detección y diferenciación de DBT Tipo 1, DBT Tipo 2 y Otros. Para esto formamos un conjunto de entrenamiento y uno de testeo a partir de evoluciones médicas.

En las HCE hay campos estructurados como por ejemplo los valores de laboratorio de estudios de pacientes. Pero se valora la posibilidad de que los médicos realicen anotaciones en texto libre. Uno de esos campos es de evoluciones.

Cada evolución es un texto complejo formado generalmente por varias oraciones. En cada una de ellas los médicos utilizan diferentes formas de expresión, nombran medicamentos, comportamientos de los pacientes, antecedentes propios o de familiares, frecuentemente con abreviaturas, comentarios y palabras escritas de diferentes maneras. Dependiendo de la patología y sintomatología, a veces los médicos vuelcan en las evoluciones información que ya está en otros campos. Pero muchas veces, expresan información que es valioso recuperar.

Es así que buscamos 340 ejemplos de evoluciones al azar donde el problema asociado fuera Diabetes Tipo 1 (130 casos), Diabetes Tipo 2 (130 casos) y Otros (es decir, casos que no fueran de diabetes) (80 casos).

En los casos de Diabetes, encontramos evoluciones referidas a controles de la enfermedad, cambios de medicación, complicaciones, sintomatología asociada, entre otros.

Fig. 1. Ejemplo de una evolución de DBT Tipo 1

Paciente de 9 años con diagnostico de DBT desde los 2 años
Regular control de la DBT
Cursa 4to grado con t de aprendisaje.
la madre refiere distención
Va a PSG 2/sem
Recibió ritalina a los 5 años por 2 mesesDisficultades en la motricidad fina.

Ef: normal
Plan: VC informe PSG
Sesiones TO

Fig. 2. Ejemplo de una evolución de DBT Tipo 2

Paciente de 63 años dbt tipo 2 de reciencinte dg Glucemia del 7-7-10 164 y del 20-7-10 TTOG Gluc basal 174 120´260 mg
Hb A1c 7.5%
Derivado por su m´wedica de cabecra para adecuar plan alñimentario
Hipotiroidismo en tto con levotiroxina
A fisica no realiza
Habitualmente saltea almuerzos y concentra la mayor parte de la ingesta por sía abundante consumo de frutas
Peso 102kg bmi 33
Imp dg obesidad G1
Plan alimentario hipoc con selección de H de c+ A fisica 150´semanles
Continúa control con su médica de cabecera

Encontramos que los médicos no siempre asocian a la evolución el problema exacto, es decir, que vimos ejemplos donde el texto se refería a DBT T1 y el problema asociado era DBT T2 y viceversa. Esto incrementa la importancia de este trabajo, ya que una incorrecta clasificación al momento del registro del paciente puede interferir en el tratamiento adecuado.

Los casos que no son de Diabetes refieren a evoluciones que pueden deberse a fracturas, análisis oftalmológicos, controles oncológicos, etc.

Fig. 3. Ejemplo de una evolución de Otros

Paciente de 87 años con diagnostico de cancer de pulmon estadio III En el dia de ayer consulto por guardia por agudizacion de disnea se medico con nebulizaciones con broncodilatadores y mejoro sintomatologia Se planifico nueva fibrobroncoscopia para el dia de mañana para evaluacion de stent y eventual aspirado de secreciones
Hoy concurre para realizar quimioterapia pero se pospone la misma hasta haber realizado el procedimiento previamente descripto

Con esta selección armamos 2 córpora con 250 evoluciones para entrenamiento (100 T1, 100 T2 y 50 otros) y 90 evoluciones para testeo (30 T1, 30 T2 y 30 otros).

3.1 Preprocesamiento

La representación que elegimos para modelar los datos fue la de Bag-of-Word (BOW bolsa de palabras). Para facilitar esta aproximación realizamos algunos preprocesamientos. Pasamos los textos a minúsculas (para evitar repeticiones de palabras), sacamos acentos (de la misma manera que en el caso de minúsculas, palabras acentuadas, sin acentos o mal acentuadas representan el mismo término).

También quitamos blancos extras, caracteres especiales como signos de puntuación, porcentajes (utilizados por ejemplo en la descripción de estudios) y números. Creamos una lista de "stopwords" a la cual exceptuamos conectores como "no" e "y". Esto es porque consideramos importante la información de negaciones y conjunciones, pero sacando las stopwords reducimos el tamaño de la BOW.

Otra técnica de reducción que probamos fue el "truncado" de palabras (stem en inglés), es decir dejar sólo la raíz de las mismas. Pero encontramos que esto afectaba los resultados disminuyendo la exactitud. Interpretamos que las palabras truncadas perdían información importante al entrenamiento.

3.2 Entrenamiento

Para la creación y entrenamiento de la Red Neuronal utilizamos el programa R [18] con la librería nnet. Para la realización de una salida multiclase codificamos los tipos de afección de la siguiente manera:

Tabla 1. Codificación multiclase para la red

Clase	Salida 1	Salida 2
DBT Tipo 1	1	0
DBT Tipo 2	0	1
Otros	0	0

Para reducir el tamaño de la BOW o sea, la cantidad de entradas a la red hicimos algunas pruebas preprocesando los datos con un Análisis de Componentes Principales (PCA). Ésta es una transformación lineal que reduce la dimensionalidad de los datos. Al reducir el espacio bajamos notablemente el tiempo de entrenamiento, permitiendo hacer pruebas rápidas con más cantidad de neuronas en la capa oculta.

Dado que las redes neuronales dependen del tiempo de entrenamiento, comparamos los resultados con otra metodología de clasificación como lo son las Máquinas de Soporte Vectorial. En las SVM el entrenamiento es mucho menos costoso en tiempo y baja la probabilidad de sobre entrenamiento. Para la creación y entrenamiento de la SVM usamos el programa R con la librería e1071.

4 Resultados

Para medir los resultados del aprendizaje medimos *accuracy*, *sensitivity* y *specificity* [19] estas últimas por cada clase, ya sea Clase 0 para otros, Clase 1 para DBT Tipo 1 y Clase 2 para DBT Tipo 2.

4.1 Redes Neuronales

Uno de los parámetros variables fue la cantidad de neuronas de la capa intermedia de la red. A medida que probamos con córpora más grandes, el tamaño de la BOW resultante se incrementaba por la cantidad de palabras diferentes que se pueden ver en las frases a analizar (como decíamos antes, las evoluciones están formadas muchas veces por varias oraciones o conceptos continuos). Es así que el agregado de una o varias neuronas ocultas hace que la cantidad de las conexiones de la red crezca considerablemente como así también el tiempo de entrenamiento.

Probamos entonces con diferentes cantidades de neuronas ocultas, primero comenzamos con 2 neuronas que representaron 7700 pesos sinápticos, lográndose una accuracy del 75,5%

Tabla 2. Resultados ANN con 2 n-ocultas.

Clase	Sensitivity	Specificity
General	**Accuracy**	**0.755**
Clase 0	0.7333	0.9666
Clase 1	0.7666	0.8666
Clase 2	0.7666	0.8000

Luego tratamos de mejorar el aprendizaje e incrementamos la cantidad de neuronas a 4. Esto duplicó la cantidad de pesos a un orden de 15400 y el tiempo de entrenamiento fue de unos 15 minutos aproximadamente.

Tabla 3. Resultados ANN con 4 n-ocultas.

Clase	Sensitivity	Specificity
General	**Accuracy**	**0.8555**
Clase 0	0.8333	0.9833
Clase 1	0.8666	0.8666
Clase 2	0.8666	0.9333

Se obtuvo una mejora considerable con una accuracy del 85,5%.

Cuando subimos la cantidad de neuronas a 5, se produjo una mejora del aprendizaje del conjunto de entrenamiento cuya accuracy llegó al 100% pero disminuyó la del testeo al 76,6%. Esto resulta en una situación de sobre entrenamiento que puede ocurrir en el caso de las redes neuronales. Con 6 neuronas intermedias, obtuvimos una accuracy del 78,8% con un tiempo de 35 minutos de aprendizaje y unos 23000 pesos sinápticos.

4.2 Redes Neuronales con PCA

Para las pruebas con PCA, comenzamos con 2 neuronas intermedias. Sin PCA la cantidad de conexiones de la red era de 7700, con PCA estuvo en el orden significativamente menor de 400. Con 6 neuronas la cantidad era de 23000, con PCA registramos 1200. Entrenamos ejemplos de hasta 50 neuronas ocultas en cuestión de unos pocos minutos (con 50 neuronas, la cantidad de pesos fue del orden de 10000).

Sin embargo, la accuracy no fue tan alta como la obtenida sin PCA. Con PCA y 2 neuronas, vimos un 53,33% (contra 75,5% sin PCA), y con 4 neuronas ocultas un 63,3% (contra 85,5% obtenido sin PCA).

Valores altos de accuracy fueron encontrados recién a las 25 y 30 neuronas intermedias (cantidades más grandes de neuronas bajaban nuevamente el porcentaje).

Tabla 4. Resultados ANN con PCA y 25 n-ocultas.

Clase	Sensitivity	Specificity
General	**Accuracy**	**0.7000**
Clase 0	0.6666	0.8500
Clase 1	0.7666	0.8333
Clase 2	0.6666	0.8666

Tabla 5. Resultados ANN con PCA y 30 n-ocultas.

Clase	Sensitivity	Specificity
General	**Accuracy**	**0.7777**
Clase 0	0.7333	0.9000
Clase 1	0.8000	0.8500
Clase 2	0.8000	0.9166

Tampoco se obtuvo una mejora en la specificity (cantidad de falsos negativos) en comparación a los resultados obtenidos sin PCA.

4.3 Máquinas de Soporte Vectorial

Para el caso de SVM, luego de varias pruebas elegimos como una mejor respuesta un kernel sigmoide. No encontramos diferencias significativas cambiando el valor del costo, por lo que no describimos variaciones de ese parámetro en este trabajo.

Realizamos variaciones con respecto al parámetro gamma del kernel. Al ser una SVM multiclase pudimos comparar también los valores obtenidos con las técnicas de predicción. Esto es, por alternativas, donde la SVM elige la clase ganadora clasificando de a dos clases por vez. O por probabilidades donde la clase ganadora es la de mayor probabilidad obtenida [20].

Tabla 6. Algunos resultados de accuracy SVM con varios gamma.

Gamma	Alternativas	Probabilidades
0.1	0.6889	0.5222
0.05	0.6222	0.6222
0.01	0.5333	0.6889
0.001	0.7333	0.7333

Como en el caso de las redes neuronales, probamos preprocesar los datos con la técnica de PCA. Encontramos que los valores de accuracy se incrementaban. Los tiempos de entrenamiento no variaron significativamente porque ya eran cortos, pero podría reducir tiempos a futuro con córpora más grandes. Los gamma que resultaron con mejores valores para SVM con PCA fueron los de 0.015 y 0.016 con probabilidades.

Tabla 7. Resultados SVM PCA prob gamma 0.015.

Clase	Sensitivity	Specificity
General	**Accuracy**	**0.8333**
Clase 0	0.9000	0.9500
Clase 1	0.7667	0.8833
Clase 2	0.8333	0.9167

Tabla 8. Resultados SVM PCA prob gamma 0.016.

Clase	Sensitivity	Specificity
General	**Accuracy**	**0.8444**
Clase 0	0.9000	0.9500
Clase 1	0.7667	0.9000
Clase 2	0.8667	0.9167

Si bien con la red neuronal sin PCA se logró una accuracy de 85,5%, la specificity de Clase 1 es levemente mejor (0.9 contra 0.86) con lo cual se podría seleccionar en el caso de buscar menos falsos positivos de DBT Tipo 1.

5 Conclusiones y trabajos futuros

En este trabajo presentamos una comparación de metodologías de clasificación para evoluciones médicas multiclase. En particular, clasificamos evoluciones relacionadas con Diabetes Tipo 1, Tipo 2 u otro tipo de afección. En el caso de DBT, no sólo es importante la detección temprana de la enfermedad sino también identificar correctamente el tipo que corresponde, ya que los tratamientos pueden diferir según la sintomatología presentada.

Por lo que observamos, obtuvimos una mejor accuracy con la red neuronal sin PCA, pero en poca diferencia con SVM con PCA. Si tomamos en cuenta grandes volúmenes de datos, las SVM tienen un tiempo de entrenamiento más corto (las redes pueden llegar a utilizar horas de aprendizaje con relativamente conjuntos no tan grandes). Y además, vimos que también podrían influir casos en los que se prioriza detectar menor cantidad de falsos positivos (specificity mayor).

Las SVM [21] implementan un principio de minimización del riesgo estructural en vez del riesgo empírico. Eso hace que minimicen un límite superior a la generalización de error en lugar de minimizar el error de entrenamiento, evitándose el sobre entrenamiento. Y el tiempo de aprendizaje se reduce también por el hecho de que las ANN pueden tener varios mínimos locales y las SVM sólo un mínimo global a alcanzar. Como conclusión podemos decir que las SVM con un preprocesamiento de PCA sería un clasificador útil en el caso de evoluciones médicas, que representan texto libre con anotaciones complejas en longitud, términos y formato.

En este trabajo utilizamos córpora no tan grandes a los fines de la comparación (en particular por los tiempos de la red, y la complejidad y longitud de las evoluciones). En trabajos futuros la idea es colectar un conjunto grande de datos para la aplicación y ajuste del método seleccionado de modo que la clasificación obtenida por esta metodología sirviera como entrada a los servicios de terminología para levantar eventuales alarmas sobre el problema asociado a cada evolución. Otros aspectos a desarrollar podrían ser el uso de otras representaciones de datos como como n-gramas y extender este estudio para otros tipos de problemas.

Referencias

1. González Bernaldo de Quirós F., Luna D., Baum A., Plazzotta F., Otero C., Benítez S.: Incorporación de tecnologías de la información y de las comunicaciones en el Hospital Italiano de Buenos Aires. Com. Económica para América Latina y el Caribe CEPAL (2012)
2. Gambarte M.L., Lopez Osornio A., Martinez M., Reynoso G., Luna D., González Bernaldo de Quirós F.: A Practical Approach to Advanced Terminology Services in Health Information Systems. Studies in Health Technology and Informatics Volume 129: MEDINFO 2007 Pages 621 – 625 (2007)
3. Jain N.L., Friedman C.: Identification of findings suspicious for breast cancer based on natural language processing of mammogram reports. Proceedings of the AMIA Annual Fall Symposium. 1997:829-833 (1997)
4. Doan S., Conway M., Phuong T.M., Ohno-Machado L.: Natural Language Processing in Biomedicine: A Unified System Architecture Overview. Clinical Bioinformatics Methods in Molecular Biology Volume 1168, 2014, pp 275-294 (2014)

5. Friedman C., Shagina L., Lussier Y., Hripcsak G.: Automated Encoding of Clinical Documents Based on Natural Language Processing. Journal of the American Medical Informatics Association : JAMIA. 2004;11(5):392-402. doi:10.1197/jamia.M1552 (2004)
6. http://www.supersalud.gob.cl/difusion/572/w3-propertyvalue-3130.html Superintendencia de Salud Chile (consultado en julio 2015)
7. Hera J.M., Vegas J.M., Hernández E., Lozano I., García-Ruiz J.M., Fernández-Cimadevilla O.C., Carro A., Avanzas P., Torres F., Bayón J., Menéndez T., Jiménez-Navarro M., Delgado E.: Rendimiento de la glucohemoglobina y un modelo de riesgo para la detección de diabetes desconocida en pacientes coronarios. Rev Esp Cardiol. 2011;64:759-65. - Vol. 64 Núm.09 DOI: 10.1016 (2011)
8. Khan I.Y., Zope P.H., Suralkar S.R.: Importance of Artificial Neural Network in Medical Diagnosis disease like acute nephritis disease and heart disease. International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 2, Issue 2, March 2013 ISSN: 2319-5967 (2013)
9. Vassy J.L., Hivert M.F., Porneala B., Dauriz M., Florez J.C., Dupuis J., Siscovick D.S., Fornage M., Rasmussen-Torvik L.J., Bouchard C., Meigs J.B.: Polygenic type 2 diabetes prediction at the limit of common variant detection. American Diabetes Association 2014 Jun;63(6):2172-82. doi: 10.2337/db13-1663 (2014)
10. Yu W., Liu T., Valdez R., Gwinn M., Khoury M.J.: Application of support vector machine modeling for prediction of common diseases: the case of diabetes and pre-diabetes. BMC Medical Informatics and Decision Mak 2010 doi: 10.1186/1472-6947-10-16 (2010)
11. Kalitzkus V., Matthiessen P.F.: Narrative-Based Medicine: Potential, Pitfalls, and Practice. The Permanente Journal. 2009;13(1):80-86 (2009)
12. Chau M., Xu J.J., Chen H.: Extracting Meaningful Entities from Police Narrative Reports. Proceeding Proceedings of the 2002 annual national conference on Digital government research Pages 1-5 Digital Government Society of North America (2002)
13. Breydo E.M., Chu J.T., Turchin A.: Identification of Inactive Medications in Narrative Medical Text. AMIA Annual Symposium Proceedings. 2008:66-70 (2008)
14. Savova G.K., Masanz J.J., Ogren P.V., et al. Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. Journal of the American Medical Informatics Association : JAMIA. 2010;17(5):507-513. doi:10.1136/jamia.2009.001560 (2010)
15. Deng Y., Denecke K.: Summarization of EHR using information extraction, sentiment analysis and word clouds. Deutsche Gesellschaft für Medizinische Informatik, Biometrie und Epidemiologie doi: 10.3205/14gmds067, urn:nbn:de:0183-14gmds0672 (2014)
16. Rodríguez J.M., Calot E., Merlino H.D.: Clasificación de Prescripciones Médicas en Español. XV Workshop de Agentes y Sistemas Inteligentes Proceedings XX Congreso Argentino de Ciencias de la Computación ISBN 978-987-3806-05-6 (2014)
17. Wright A., McCoy A.B., Henkin S., Kale A., Sittig D.F.: Use of a support vector machine for categorizing free-text notes: assessment of accuracy across two institutions. Journal of the American Medical Informatics Association : JAMIA. 2013;20(5):887-890. doi:10.1136/amiajnl-2012-001576 (2013)
18. The R Project Statistical Computing http://www.r-project.org/ (consultado en julio 2015)
19. Parikh R, Mathai A, Parikh S, Chandra Sekhar G, Thomas R. Understanding and using sensitivity, specificity and predictive values. Indian Journal of Ophthalmology. 2008;56(1):45-50 (2008)
20. LIBSVM: A Library for Support Vector Machines http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf (consultado en julio 2015)
21. Lin J.Y., Cheng C.T., Chau K.W.: Using support vector machines for long-term discharge predict. Hydrological Sciences–Journal–des Sciences Hydrologiques Vol 51, Issue 4 (2006)

Representing Traffic Congestions on Moving Objects Trajectories

Mariano Kohan and Juan M. Ale

Facultad de Ingeniería, Universidad de Buenos Aires
marianokohan@gmail.com, ale@acm.org

Abstract. The discovery of moving objects trajectory patterns representing a high traffic density have been covered on different works using diverse approaches. These models are useful for the areas of transportation planning, traffic monitoring and advertising on public roads. Besides of the important utility, these type of patterns usually do not specify a difference between a high traffic and a traffic congestion. In this work, we propose a model for the discovery of high traffic flow patterns and traffic congestions, represented in the same pattern. Also, as a complement, we present a model that discovers alternative paths to the severe traffic on these patterns. These proposed patterns could help to improve traffic allowing the identification of problems and possible alternatives.

Keywords: Moving objects, trajectories, road network, traffic flow, traffic congestion

1 Introduction

In the last years, there has been a high presence of works related to the data mining of the trajectory data generated by moving objects ([16]). From these works, there has been a lot of attention to the discovery of different type of traffic flow patterns. These patterns can be discovered from trajectories moving inside a road network like [7], [6] and [9] or with a free movement ([5], [11], [4]). Related to the first case, the concept of traffic congestions as a limitation of the road network is considered on more recent papers ([14], [1]). These works are useful on different areas like transport planning, traffic monitoring, carpooling, store locations and advertising on public roads.

In this work, we propose a model for the discovery of high traffic flow patterns in relation to traffic congestions. This relation is displayed according to the shared traffic, like presented in the hot routes model ([7]). Also we present, as a complement, a model that relates these patterns with alternative paths, according to a low traffic level and useful location inside the road network. These models present an increased utility allowing to be applied on additional cases, like the identification of paths inside the road network that require changes (for transport planning), and the redirection of the drivers that contribute to the different congestion areas into alternative paths (for traffic monitoring).

The rest of the paper is organized as follows. Next section comments the works related to this paper. Section 3 describes selected concepts considered from these works. The proposed models are introduced on Section 4. And Sections 5 and 6 gives the definitions and algorithms for each model. Finally, Section 7 concludes this paper with information about the next steps.

2 Related Work

Works about the discovery of traffic flow patterns are related to this paper. The model of hot routes ([7]) is used as the main inspiration for this work because of its balance between the aggregate information about the moving objects and theirs specific behavior (represented in the common traffic in a sequence of edges). Li and others on [7] comments about some alternative methods to discover traffic flow patterns. First, just the aggregate behavior of individuals can be considered connecting only edges in the graph with high traffic. [6] uses this method and complements it with the discovery of the temporal evolution of the patterns. Also, in [9] the model is oriented to the traffic analysis through edges clustering. Another method is to discover moving clusters formed by moving objects, where [5] and [8] are some examples. The third method is about the clustering of trajectories. In this group we can consider some patterns like hot motion paths ([11]), the discovery of the Most Popular Route (MPR) between two locations ([4]) and the distributed parallel clustering method MCR-ACA ([15]). Besides of representing these patterns a high traffic inside the road network, they usually do not consider the cases of traffic jams, where the traffic density is close to the network capacity.

Another group of papers are related to the analysis of traffic congestions. To discover this type of patterns, it is possible to consider the road network characteristics or the moving objects data. In the first case, we have works about representative patterns for the network segments ([2]), usage patterns of road networks ([14], [13]), and the visualization of traffic jams using a GIS map service ([12]). For the second case we can consider diverse patterns like slowly flocks ([10]), the transitions within regions ([17]) and Non-Recurrent Congestion events ([1]). Also, there is the work [3] that considers both types of data in the discovered patterns. The main difference with the current work is that this second group of papers only discover patterns related to traffic congestions, but without relation to patterns with lower levels of traffic.

3 Background Models

The concepts presented in some of the related works are used as a starting point for the models proposed in this paper. In [7], the hot routes patterns are presented. These are traffic flow patterns inside a road network. The road network is represented by a graph $G(V,E)$, where E is the set of edges representing a unit of road segment, and V

the set of vertices representing a street intersection. Also, T is the set of trajectories, with each element composed of an ID (*tid*) and a sequence of edges traveled through: $(tid, <e_1, ..., e_k>)$, where $e_i \in E$. The hot routes are built up with a sequence of edges, near each other but not necessarily adjacent, that "...share a high amount of traffic between them." ([7]). The distance between the edges is based on the number of edges inside the road network graph, according to the metric *ForwardNumHops*, which represents the minimum number of edges between the end vertex of two edges. Using this metric, the *Eps-neighborhood* ($N_{Eps}(r)$) of an edge r defines the set of closed edges. The shared traffic considers the same trajectory identification made by each moving object. This model is used as a base for the development of patterns considering traffic congestions.

The work [10] is about the detection of potential traffic jams with slowly flock patterns. In this case, the velocity of each moving object is considered on the discovery of the flock patterns. This idea of using the velocity to identify a traffic jam is applied on the first proposed model.

The discovery of representative patterns for the network segments is proposed on [2]. On this work, the network segments are characterized according to presented network features like length, direction, capacity and density. This last concept of *segment density D*(s)* allows to identify alternative segments for an edge according to a bounding rectangle (*BR*) covering the segment, and the direction for the edges inside this *BR*. For the second model proposed, this concept is used to discover alternative paths to the traffic congestions.

[6] presents a more general pattern, called dense routes. These patterns are discovered using only the number of objects on each edge of the road network, and adjacent edges are linked if the difference on the number of moving objects is bellow a maximum threshold. A similar idea for the algorithm described on the second model is considered for the discovery of the alternative paths.

4 Models Description

We consider the model of hot routes ([7]) to be most appropriate to discover patterns with heavy traffic in a city road network, because it represents a balance between an aggregate analysis and the behaviors of the individuals. But besides of represent a high density of moving objects in a road network, it does not consider some characteristics of this road network causing the appearance of traffic jams:

- capacity: it is associated with the edges in the road network and represents the maximum number of vehicles that are allowed to circulate into a road segment.
- velocity and time: related to the feature of capacity are the concepts of velocity and time. When the density of objects in a road segment is close to its capacity, the velocity of the moving objects starts to be decreased and the travel time is extended.
- when the velocity and time starts to be affected in an edge, the drivers from the vehicles might choose to continue its path on another alternative segments. This concept of alternative segments is designed as *Density (D*)* elsewhere.

In this paper, we propose two new models for the discovery of trajectory patterns considering these features:

1. the concept of velocity is considered in order to discover hot routes with jam sections: sections with a density close to its capacity. We call these patterns *jam routes*. This model is described on the next section.
2. the existence of paths that could be used as alternative to the traffic in a *jam route*, because of its location and low density values. These patterns are called *cold routes*. Section 6 presents this model.

5 Discovery of Jam Routes

The *jam route* pattern could be defined as a hot route with one or more subpaths identified as traffic jam. So, it is a path in a road network with heavy traffic (shared by the same objects inside a sliding window) and with one or more sectors having a traffic level close to its capacity.

In order to identify these subpaths the velocity is used. So, each trajectory is composed by its ID (*tid*) and a sequence of pairs representing each edge traveled with its respective mean velocity: $(tid, <(e_1, v_1), (e_2, v_2), \ldots, (e_k, v_k)>)$, where $e_i \in E$ and v_i is the *mean velocity* on e_i.

We consider the use of the velocity to identify traffic congestions is better than compare the density with the road capacity for two reasons:

- the data about the velocity for each moving object on each edge of the trajectory is easier to obtain than the capacity of each edge of the road network
- considering the road segments are part of a network, there also additional factors that could lead to congestions ([14])

The first concept to present is *speed*. It complements the *traffic* definition from [7] to consider the velocity in each edge.

Definition 1 (speed). The *speed(r)* for a given edge *r* is the mean of velocities v_i of the edge *r*.

In order to identify the edges affected by the conditions of a traffic jam the concept *directly traffic jam-reachable* is used.

Definition 2 (directly traffic jam-reachable). An edge *s* is *directly traffic jam-reachable* from an edge *r* with respect to parameters *Eps*, *MinTraffic* and *JamSpeed* if

1. $s \in N_{Eps}(r)$
2. $|traffic(r) \cap traffic(s)| \geq MinTraffic$
3. $speed(s) \leq JamSpeed$ *or* $speed(r) \leq JamSpeed$

This definition extends the concept of *directly traffic density-reachable* ([7]), to identify traffic jams but maintaining the condition of shared traffic between the edges.

Definition 3 (route traffic jam-reachable). An edge *s* is *route traffic jam-reachable* from an edge *r* with respect to parameters *Eps*, *MinTraffic* and *JamSpeed* if

1. *there is a chain of edges* $r_1, r_2, \ldots, r_n$ *with* $r_1=r$ *and* $r_n=s$, *where* r_i *is directly traffic jam-reachable from* r_{i-1} *or* r_i *is just directly traffic density-reachable from* r_{i-1}
2. *for every Eps consecutive edges in the chain,* $\left| traffic(r_i) \cap traffic(r_{i+1}) \cap \ldots \cap traffic(r_{i+Eps}) \right| \geq MinTraffic$

This definition augments the concept of *route traffic density-reachable* ([7]), allowing to propose a path that relates sections with heavy traffic and sections with traffic jams.

This concept is the base for the discovery of the *jam routes*.

5.1 Algorithm

The algorithm to discover the *jam routes* presents a structure of breadth-first search on the road network graph.

It starts out the discovery from the *hot routes starts* ([7]), verifying if the *speed* on each of these edges is below the *JamSpeed* threshold. In this case, the edge is marked as a *jam*. Next, these *hot routes starts* are extended recursively to form the *jam routes*. The extension is from the last edge, finding the edges inside the N_{Eps} that satisfy the definitions of *directly traffic jam-reachable* or *directly traffic density-reachable.* Then, on each of these possible split edge from the route, the definition of *route traffic jam-reachable* is evaluated (specifically the second condition). If this definition is validated, a new *jam route* is created with the new edge. And, if the added edge is *directly traffic jam-reachable*, it is marked as a *jam*.

The algorithm is called *JamFlowScan* and its pseudo-code is presented as follows:

```
Input: Road network G, object trajectory data T, Eps,
MinTraffic, JamSpeed
Output: Jam routes R

1: Initialize R to {}
2: Let H be the set of hot route starts in G according
to T
3: for every hot route start h in H do
4:   r = new Jam Route initialized to <h> /*mark edge
as "jam" if speed(h) <= JamSpeed */
5:   Add Extend_Jam_Routes(r) to R
6: end for
7: Return R
```

```
Procedure Extend_Jam_Routes(jam route r)
1:  Let p be the last edge in r
2:  Let Q be the set of directly traffic jam-reachable
neighbors of p U the set of directly traffic density-
reachable neighbors of p
3:  if Q is non-empty then
4:    Initialize JR to {}
5:    for every split in Q do
6:      if route traffic jam-reachable condition is
satisfied then
7:        Let r' be a copy of r
8:        Append split's edges to r'
9:        if directly traffic jam-reachable condition
is satisfied then
10:         mark split edge as "jam"
11:       end if
12:       Add Extend_Jam_Routes(r') to JR
13:     end if
14:   end for
15:   return JR
16: else
17:   Return {r}
18: end if
```

To verify the definitions used in the algorithm, the *traffic* set and *speed* for every edge is required. So, the object trajectory data *T* can be converted into table structure that relates each edge with the *tid* of the trajectories that belongs to, and the mean velocity on all the trajectories. The building of this table has linear complexity with respect to the trajectories data.

The jam routes are discovered applying the definitions from the model and identifying the traffic jams on the respective cases:

- initially after the identification of the *hot routes starts* (step 4)
- on the extension of the *jam route* for each split, following the identification of an edge as *route traffic jam-reachable* (steps 9-11 from `Extend_Jam_Routes`).

So, if a traffic congestion is found during the route building, it will be properly identified on the results. Also, the order used to extend the routes adds efficiency to the search but does not omit edges. Therefore, the set of *jam routes* discovered is complete and correct.

6 Discovering Cold Routes

The *cold route* pattern is a path in a road network with low traffic (so it does not affect the network capacity) and with a location inside the road network that allows to be chosen as an alternative path to the traffic present in a *jam route*.

To allow the identification of the alternative routes the concept *BR-neighborhood* $N_{BR}(s)$ is used. It is the same concept *segment density D*(s)* from [2] (but using a name following the conventions applied to this work): considers the vicinity area of a segment (with a bounding rectangle) and the direction. So, each edge $e \in E$ from the road network graph $G(V,E)$ will be associated with a label representing its direction.

The first concept to present is *cold traffic*. It allows to identify edges with low traffic and that could be considered alternatives to edges with traffic jam (d*irectly traffic jam-reachable*).

Definition 4 (cold traffic). An edge *s* is considered *cold traffic* with respect to parameters *BR* and *MaxTraffic* if:

1. $\left| traffic(s) \right| \leq MaxTraffic$
2. $s \in N_{BR}(s)$ *of directly traffic jam-reachable edge*

Additionally the concept of *directly cold traffic reachable* is presented.

Definition 5 (directly cold traffic reachable). An edge *s* is considered *directly cold traffic reachable* from an edge *r* with respect to parameter *MaxTraffic* if:

1. *s is adjacent to r: start(s) = end(r) or end(s) = start(r)*
2. $\left| traffic(r) \right| \leq MaxTraffic$
3. $\left| traffic(s) \right| \leq MaxTraffic$

Both concepts are related on the definition of *route cold traffic reachable.*

Definition 6 (route cold traffic reachable). An edge *s* is considered *route cold traffic reachable* from an edge *r* with respect to parameters *BR* and *MaxTraffic* if there is a chain of edges $r_1, r_2, \ldots, r_n$ *with* $r_1 = r$ *and* $r_n = s$, where:

1. *each* r_i *is directly cold traffic reachable from* r_{i-1}
2. *there exists almost one edge* r_i *that is cold traffic*

The concept of *route cold traffic reachable* allows the discovery of the *cold route* patterns.

6.1 Algorithm

Considering that *cold routes* are formed by edges with low traffic, it is better to discover them using a simple aggregate method.

The proposed algorithm starts the discovery process from the *jam routes* discovered by *JamFlowScan,* finding the *cold traffic* edges according to the N_{BR} of the *directly traffic jam-reachable* edges. Next, these edges are extended to both sides, evaluating the definition *directly cold traffic reachable* into the adjacent edges. With the two conditions of *route cold traffic reachable* satisfied, the new edge is added to the route, considering possible splits (representing different alternative paths).

The algorithm is called *ColdScan* and it pseudo-code is presented as follows:

```
Input: Road network G, object trajectory data T,
MaxTraffic, BR, JamRoutes (from JamFlowScan)
Output: Cold routes CR

1:  Initialize CR to {}
2:  Let CS the set of cold traffic edges in G according
to T and discovered Jam Routes
3:  for every cold traffic edge cs in CS do
4:    cr = new Cold Route initialized to <cs>
5:    Add Extend_Cold_Route_Forward(cs) to CRf
6:    for every route (extended forward from cs) crf in
CRf do
7:      Add Extend_Cold_Route_Backward(crf) to CR
8:    end for
9:  end for
10: return CR

Procedure Extend_Cold_Route_Forward (cold route cr)
1:  Let p be the last edge in cr
2:  Let S be the set of directly cold traffic reachable
edges from p with end(p) = start(s)
3:  if S is non-empty then
4:    Initialize CR to {}
5:    for every edge s in S
6:      Let cr' be a copy of cr
7:      Append edge s to the end of cr'
8:      Add Extend_Cold_Route_Forward(cr') to CR
9:    end for
10:   return CR
11: else
12:   return {cr}
13: end if

Procedure Extend_Cold_Route_Backward (cold route cr)
1:  Let p be the first edge in cr
2:  Let S be the set of directly cold traffic reachable
edges from p with start(p) = end(s)
3:  if S is non-empty then
4:    Initialize CR to {}
5:    for every edge s in S
6:      Let cr' be a copy of cr
7:      Append edge s to the beginning of cr'
8:      Add Extend_Cold_Route_Backward (cr') to CR
9:    end for
10:   return CR
11: else
12:   return {cr}
13: end if
```

The algorithm requires, to verify the definitions used, the *traffic* set for every edge. So, in this case a similar table structure built from the trajectories can be utilized, with a linear complexity with respect to the trajectory data.

ColdScan discovery process applies the definition of *route cold traffic reachable*, considering all the *jam routes* from *JamFlowScan*. Also, these routes are extended to both possible sides. So, the discovered set of *cold routes* is complete.

7 Conclusion and Future Work

In this paper we presented two models for the discovery of traffic flow patterns. The hot routes model for the discovery of high traffic routes is considered as a starting point for the development of patterns representing traffic jams and its alternative paths.

First, in the *jam routes* model the velocity of the moving objects is added in order to identify traffic jam sectors inside the patterns. The relation between these congestions and the high traffic density is according to the shared traffic in common. Next, starting with a vicinity concept the *cold routes* are presented as a path that could be used as an alternative to the traffic in the *jam routes*. These patterns are identified according to a low level of traffic and comparing its location in the road network graph with respect to the congestions in the *jam routes*. The algorithms for the discovery of the proposed models are presented in order to clarify further details.

This is a work in progress. The next step is the implementation of the presented models, in order to compare the discovered patterns with the obtained using some of the related models. This will allow to confirm the utility of these models.

References

1. Berk Anbaroglu, Benjamin Heydecker, and Tao Cheng. Spatio-temporal clustering for non-recurrent traffic congestion detection on urban road networks. Transportation Research Part C: Emerging Technologies, 48(0):47 – 65, 2014.
2. Farnoush Banaei-Kashani, Cyrus Shahabi, and Bei Pan. Discovering patterns in traffic sensor data. In: Proceedings of the 2Nd ACM SIGSPATIAL International Workshop on GeoStreaming, IWGS '11, pp. 10–16, New York, USA, 2011. ACM.
3. Kyoung Soo Bok, He Li, Jong Tae Lim, and Jae Soo Yoo. Discovering congested routes using vehicle trajectories in road networks. Advances in Multimedia, 2014.
4. Zaiben Chen, Heng Tao Shen, and Xiaofang Zhou. Discovering popular routes from trajectories. In: Data Engineering (ICDE), 2011 IEEE 27th International Conference on, pp 900–911, 2011.
5. Panos Kalnis, Nikos Mamoulis, and Spiridon Bakiras. On discovering moving clusters in spatiotemporal data. In: Claudia Bauzer Medeiros, Max J. Egenhofer, and Elisa Bertino, editors, Advances in Spatial and Temporal Databases, volume 3633 of Lecture Notes in Computer Science, pp. 364–381. Springer Berlin Heidelberg, 2005.

6. Ahmed Kharrat, Karine Zeitouni, Iulian Sandu-Popa, and Sami Faiz. Characterizing traffic density and its evolution through moving object trajectories. Fifth International IEEE Conference on Signal-Image Technologies and Internet-Based Systems, pp. 257–263, 2009.
7. Xiaolei Li, Jiawei Han, Jae-Gil Lee, and Hector Gonzalez. Traffic density-based discovery of hot routes in road networks. In: SSTD'07 - Proceedings of the 10th international conference on Advances in spatial and temporal databases, pp. 441–459, Berlin, Heidelberg, 2007. Springer-Verlag.
8. Wenting Liu, Zhijian Wang, and Jun Feng. Continuous clustering of moving objects in spatial networks. In: Ignac Lovrek, Robert J. Howlett, and Lakhmi C. Jain, editors, Knowledge-Based Intelligent Information and Engineering Systems, volume 5178 of Lecture Notes in Computer Science, pp. 543–550. Springer Berlin Heidelberg, 2008.
9. Irene Ntoutsi, Nikos Mitsou, and Gerasimos Marketos. Traffic mining in a road-network: How does the traffic flow? International Journal of Business Intelligence and Data Mining, 3:82–98(17), 2008.
10. Rebecca Ong, Fabio Pinelli, Roberto Trasarti, Mirco Nanni, Chiara Renso, Salvatore Rinzivillo, and Fosca Giannotti. Traffic jams detection using flock mining. In: Dimitrios Gunopulos, Thomas Hofmann, Donato Malerba, and Michalis Vazirgiannis, editors, Machine Learning and Knowledge Discovery in Databases, volume 6913 of Lecture Notes in Computer Science, pp. 650–653. Springer Berlin Heidelberg, 2011.
11. Dimitris Sacharidis, Kostas Patroumpas, Manolis Terrovitis, Verena Kantere, Michalis Potamias, Kyriakos Mouratidis, and Timos Sellis. On-line discovery of hot motion paths. In: Proceedings of the 11th International Conference on Extending Database Technology: Advances in Database Technology, EDBT '08, pp. 392–403, New York, NY, USA, 2008. ACM.
12. Anna Izabel J. Tostes, Fátima de L. P. Duarte-Figueiredo, Renato Assunção, Juliana Salles, and Antonio A. F. Loureiro. From data to knowledge: City-wide traffic flows analysis and prediction using Bing maps. In: Proceedings of the 2Nd ACM SIGKDD International Workshop on Urban Computing, UrbComp '13, pp. 12:1–12:8, New York, NY, USA, 2013. ACM.
13. Junjie Wang, Dong Wei, Kun M. He, Hang Gong, and Pu Wang. Encapsulating urban traffic rhythms into road networks. Scientific Reports, 4, 2014.
14. Pu Wang, Timothy Hunter, Alexandre M. Bayen, Katja Schechtner, and Marta C. Gonzalez. Understanding road usage patterns in urban areas. Scientific Reports, 2, 2012.
15. Jie Yang, Xiaoping Li, Dandan Wang, and Jia Wang. A group mining method for big data on distributed vehicle trajectories in wan. International Journal of Distributed Sensor Networks, 2014.
16. Yu Zheng. Trajectory data mining: An overview. ACM Transaction on Intelligent Systems and Technology, 2015.
17. Yu Zheng, Yanchi Liu, Jing Yuan, and Xing Xie. Urban computing with taxicabs. In: Ubicomp 2011. ACM, 2011.

Immune Algorithm for Solving the Smooth Economic Dispatch Problem

Victoria S. Aragón and Susana C. Esquivel

Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)
Universidad Nacional de San Luis
Ejército de los Andes 950 - (5700) San Luis, ARGENTINA
CONICET

Abstract. In this paper, an algorithm inspired on the T-Cell model of the immune system is presented, it is used to solve Economic Dispatch Problems with smooth objective function. The proposed approach is called IA_EDP_S, which stands for *Immune Algorithm for Economic Dispatch Problem for smooth objective function*, and it uses as differentiation process a redistribution power operator. The proposed approach is validated using five problems taken from the specialized literature. Our results are compared with respect to those obtained by several other approaches.

Keywords: Artificial immune systems , economic dispatch problem , metaheuristics

1 Introduction

The objective of Economic Dispatch Problem (EDP) is to minimize the total generation cost of a power system while satisfying several constraints associated to the system, such as load demands, ramp rate limits, maximum and minimum limits, and prohibited operating zones. The objective function type (smooth or non smooth) and the constraints which are considered in the problem will determine how hard is to solve the problem.

Over the last years, several methods have been proposed to solve the EDP. They can be divided in three main groups: classical, based on artificial intelligence (AI) and hybrid methods. Classical methods have been proposed to solve EDP, but they suffer from some limitations (for instance, the objective functions and the constraints must be differentiable). On the other hand, modern heuristic algorithms have proved to be able to deal with nonlinear optimization problems, e.g., EDPs. Surveys about these techniques can be found in [14] and [2].

In this paper, we propose an algorithm to solve EDPs which is inspired on the T cells from the immune system. Once the algorithm has found a feasible solution, it applies a redistribution power operator in order to improve the original solution with the aim of keeping such a solution feasible at a low computational cost.

The remainder of this paper is organized as follows. Section 2 defines the economic dispatch problem. In Section 3, we describe our proposed algorithm. In Section 4, we present the test problems used to validate our proposed approach and parameters settings. In Section 5, we present our results and we discuss and compare them with respect to other approaches. Finally, in Section 6, we present our conclusions and some possible paths for future research.

2 Problem Formulation

The schedule has to minimize the total production cost and involves the satisfaction of both equality and inequality constraints.

2.1 Objective Function

Minimize

$$TC = \sum_{i=1}^{N} F_i(P_i)$$

where TC is the fuel cost, N is the number of generating units in the system, P_i is the power of i^{th} unit (in MW) and F_i is the total fuel cost for the i^{th} unit (in \$/h).

An EDP with a smooth cost function represents the simplest cost function. It can be expressed as a single quadratic function: $Fi(P_i) = a_i P_i^2 + b_i P_i + c_i$, where a_i , b_i and c_i are the fuel consumption cost coefficients of the i^{th} unit.

2.2 Constraints

1. Power Balance Constraint: the power generated has to be equal to the power demand required. It is defined as: $\sum_{i=1}^{N} P_i = P_D$
2. Operating Limit Constraints: thermal units have physical limits about the minimum and maximum power that can generate:$P_{min_i} \leq P_i \leq P_{max_i}$, where P_{min_i} and P_{max_i} are the minimum and maximum power output of the i^{th} unit, respectively.
3. Power Balance with Transmission Loss: some power systems include the transmission network loss, thus Power Balance Constraint equation is replaced by:$\sum_{i=1}^{N} P_i = P_D + P_L$
   The P_L value is calculated with a function of unit power outputs that uses a loss coefficients matrix B, a vector B0 and a value B00:
   $PL = \sum_{i=1}^{N} \sum_{j=1}^{N} P_i B_{ij} P_j + \sum_{i=1} B0_i P_i + B00$
4. Ramp Rate Limits: they restrict the operating range of all on-line units. Such limits indicate how quickly the unit's output can be changed:$max(P_{min_j}, P_j^0 - DR_j) \leq P_j \leq min(P_{max_j}, P_j^0 + UR_j)$, where P_j^0 is the previous output power of the j^{th} unit(in MW) and, UR_j and DR_j are the up-ramp and down-ramp limits of the j^{th} unit (in MW/h), respectively.

5. Prohibited Operating Zones: they restrict the operation of the units due to steam valve operation conditions or to vibrations in the shaft bearing:
$$\begin{cases} P_{min_i} \leq P_i \leq P_{i,1}^l \\ P_{i,j1}^u \leq P_i \leq P_{i,j}^l, \quad j = 2, 3, ..., nj \\ P_{i,nj}^u \leq P_i \leq P_{max_i} \end{cases}$$
where nj is the number of prohibited zones of the i^{th} unit, $P_{i,j}^l$ and $P_{i,j}^u$ are the lower and upper bounds of the j^{th} prohibited zone.

3 Our Proposed Algorithm

In this paper, an adaptive immune system model based on the immune responses mediated by the T cells is presented. These cells present special receptors on their surface called T cell receptors (TCR: are responsible for recognizing antigens bound to major histocompatibility complex (MHC) molecules.) [6].

The model considers some processes that T cells suffer. These are proliferation (to clone a cell) and differentiation (to change the clones so that they acquire specialized functional properties); this is the so-called activation process.

IA_EDP_S (Immune Algorithm for Economic Dispatch Problem with Smooth Objective Function) is an adaptation of an algorithm inspired on the activation process [2], which is proposed to solve the EDP with Smooth Objective Function. IA_EDP_S operates on one population which is composed of a set of T cells.

For each cell, the following information is kept:

1. TCR: it identifies the decision variables of the problem ($TCR \in \Re^N$). Each thermal unit is represented by one decision variable.
2. $objective$: objective function value for TCR, ($TC(TCR)$).
3. $prolif$: it is the number of clones that will be assigned to the cell, it is N for all problems.
4. $differ$: it is the number of decision variables that will be changed when the differentiation process takes place (if applicable).
5. TP: it is the power generated by TCR ($\sum_{i=1}^{N} TCR_i$).
6. P_L: it is the transmission loss for TCR (if the problem does not consider transmission loss, then $P_L = 0$).
7. ECV: it is the equality constraint violation for TCR ($\mid TP - P_D - P_L \mid$). If $ECV > 0$, then the power generated is bigger than the demanded power, and if $ECV < 0$ then the power generated is lower than the required power.
8. ICS: it is the inequality constraints sum, $\sum_{i=1}^{nj} poz(TCR_i, i)$
$$poz(p, i) = \begin{cases} \min(p - PZ_{ll_i}, PZ_{ul_i} - p) & if p \in [PZ_{ll_i}, PZ_{ul_i}] \\ 0 & otherwise \end{cases}$$
where nj is the number of prohibited operating zones and $[PZ_{ll_i}, PZ_{ul_i}]$ is the prohibited range for the i^{th} thermal unit.
9. feasible: it indicates if the cell is feasible or not. A cell is considered as feasible if: 1) $ECV = 0$ for problems without transmission network loss and $0 \leq ECV < \epsilon$ for problems with transmission loss. This means that if a solution generates less than the demanded power, then it is considered

as infeasible ($ECV < 0$) and 2) $ICS = 0$ for problems which consider prohibited operating zones.

Differentiation for feasible cells - Redistribution Process

The idea is to take a value (called d) from one unit (say i) and assign it to another unit (variable). i^{th} unit is modified according to: $cell.TCR_i = cell.TCR_i - d$, where $d = U(prob*D, D)$, $D = \min(cell.TCR_i - ll_i, U(min, max)))$, $U(w_1, w_2)$ refers to a random number with a uniform distribution in the range (w_1,w_2), max is the maximum power that can be generated by the other units according to their current outputs (i.e. $max = \max_{n=1 \wedge n \neq i}^{N}(ul_n - cell.TCR_n)$, min is the minimum power that can be generated by the other units according to their current outputs (i.e. $min = \min_{n=1 \wedge n \neq i}^{N}(ul_n - cell.TCR_n)$).

d was designed to avoid: 1) that the i^{th} unit falls below its lower limit and 2) to take from the i^{th} unit more power of what other units can generate. Next, d has to increase the power of another unit (say k). In a random way k is selected considering $cell.TCR_k + d \leq ul_k$.

The main difference between IA_EDP_S and the algorithm proposed in [2] arises in the number of variables that are modified. This version just changes i and k while version [2] changes i and one o more variables. Note this operator only preserve the feasibility of solutions by taking into account the power balance constraints.

Differentiation for infeasible cells

For infeasible cells, the number of decision variables to be changed is determined by their differentiation level. This level is calculated as $U(1, N)$. Each variable to be changed is chosen in a random way and it is modified according to: $cell.TCR_i^{'} = cell.TCR_i \pm m$, where $cell.TCR_i$ and $cell.TCR_i^{'}$ are the original and the mutated decision variables, respectively. $m = U(0, 1)* \mid cell.ECV + cell.ICS \mid$. In a random way, it decides if m will be added or subtracted to $cell.TCR_i$. If the procedure cannot find a TCR_i' in the allowable range, then a random number with a uniform distribution is assigned to it ($cell.TCR_i^{'} = U(cell.TCR_i, ul_i)$ if m should be added or $cell.TCR_i^{'} = U(ll_i, cell.TCR_i)$, otherwise).

The algorithm works in the following way (see Algorithm 1). First, the TCRs are randomly initialized within the limits of the units (Step 1). Then, ECV and ICS are calculated for each cell (Step 2). Only if a cell is feasible, its objective function value is calculated (Step 3). Next, while a predetermined number of objective function evaluations had not been reached or if after 50 iterations the best value does not improve (Steps 4-6) the cells are proliferated and differentiated considering if they are feasible or infeasible. Finally, statistics are calculated (Step 8).

Algorithm 1 IA_EDP_S Algorithm

```
Initialize_Population();
Evaluate_Constraints();
Evaluate_Objective_Function();
while A predetermined number of evaluations has not been reached or Not improve
do
    Proliferation_Population();
    Differentiation_Population();
end while
Statistics();
```

4 Validation

IA_EDP_S performance was validated with five test problems, SYS_3U, SYS_6U, SYS_15U, SYS_18U and SYS_20U (see [2] for full description). Table 1 provides their most relevant characteristics and the maximum number of function evaluations. IA_EDP_S was implemented in Java (version 1.6.0_24) and the experiments were performed in an Intel Q9550 Quad Core processor running at 2.83GHz and with 4GB DDR3 1333Mz in RAM.

Table 1. Test Problems Characteristics

Problem	Thermal Units	P_L	Prohibited Zones	P_D (MW)	Evaluations
SYS_3U	3	No	No	850.0	1000
SYS_6U	6	Yes	Yes	1263.0	3000
SYS_15U	15	Yes	Yes	2630.0	20000
SYS_18U	18	No	No	365.0	40000
SYS_20U	20	Yes	No	2500.0	20000

The required parameters by IA_EDP_S are: size of population, number of objective function evaluations, and probability for redistribution operator. To analyze the effect of the first and third parameters on IA_EDP_S's behavior, we tested it with different parameters settings. Some preliminary experiments were performed to discard some values for the population size parameter. Hence, the selected parameter levels were: a) Population size (C) has four levels: 1, 5, 10 and 20 cells and b) Probability has three levels: 0.01, 0.1 and 0.5.

Thus, we have 12 parameters settings for five problems. They are identified as C<size>-Pr<Prob>, where C and Pr indicate the population size and the probability, respectively. For each problem, 100 independent runs were performed.

The box plot method was selected to visualize the distribution of the objective function values for each power system. This allowed us to determine the robustness of our proposed algorithm with respect to its parameters. Figures 1

to 3 show in the x-axis the parameter combinations and the y-axis indicates the objective function values for each problem. We can see that better results are reached with the lowest probability value and the highest population size. So, C=5and Pr=0.01 were used to compare the results got by IA_EDP_S with those produced by other approaches.

Considering the lowest number of objective function evaluations used by the other approaches (see [2]) we take as maximum number of function evaluations, 1000, 40000, 3000, 20000 and 20000 for SYS_3U, SYS_18U, SYS_6U, SYS_15U and SYS_20U, respectively. Also, we set ϵ=0.1 for those problems which consider loss transmission (e.d. SYS_6U, SYS_15U and SYS_20U).

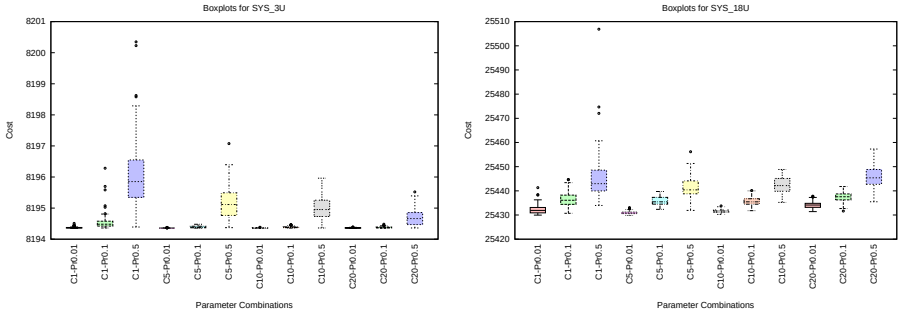


Fig. 1. Box plots for the test problems with the best parameters combination

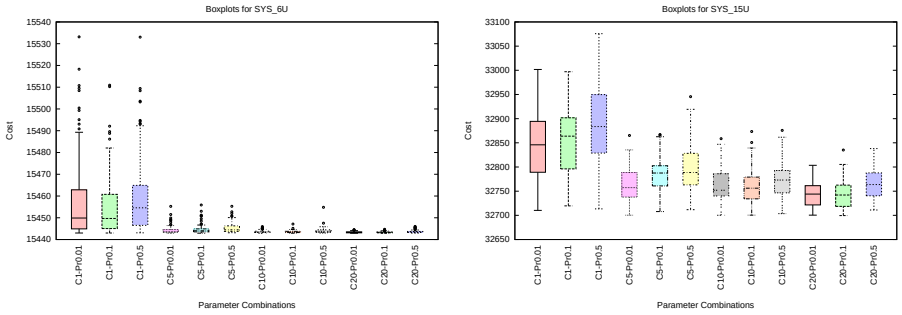


Fig. 2. Box plots for the test problems with the best parameters combination

5 Comparison of Results and Discussion

Table 2 shows: the best, worst, mean, median, standard deviation and number of function evaluations obtained by IA_EDP_S. Only four decimal digits are shown

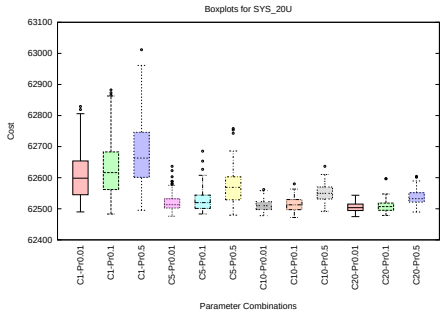


Fig. 3. Box plots for the test problem with the best parameters combination

due to space restrictions. For all the test problems, our proposed IA_EDP_S found feasible solutions in all the runs performed.

Problems which do not consider transmission loss, rate ramp limits or prohibited zones, i.e., SYS_3U and SYS_18U, do not seem to be a challenge for IA_EDP_S. The standard deviations obtained by IA_EDP_S are lower than 1. Additionally, the problem dimensionality does not seem to affect the performance of our proposed approach either.

For problems which consider transmission loss, rate ramp limits and prohibited zones, SYS_6U_a and SYS_15U, the standard deviations increase with the problem dimensionality.

For the only problem which considers transmission loss but not rate ramp limits or prohibited zones, SYS_20U, the standard deviation is lower than SYS_15U's standard deviation.

Table 2. Results obtained by IA_EDP_S

Problem	Best	Worst	Mean	Median	SD.	Ev.
SYS_3U	8194.3561	8194.3784	8194.3597	8194.3584	0.004	987.16
SYS_18U	25429.8005	25433.0655	25430.9312	25430.8415	0.614	35103.15
SYS_6U	15442.8962	15455.2466	15444.3082	15443.6071	1.877	1490.62
SYS_15U	32700.2971	32865.2657	32763.5364	32758.1897	35.765	18321.5
SYS_20U	62476.1186	62636.5875	62522.3703	62513.2753	30.371	8151.36

Eleven methods are compared with respect to IA_EDP_S. They are cited in Table 3. The running time of each algorithm is affected by both the hardware environment and the software environment. That is the reason why the main comparison criterion that we adopted for assessing efficiency was the number of objective function evaluations performed by each approach. For having a fair comparison of the running times of all the algorithms considered in our study, they should all be run in the same software and hardware environment (some-

thing that was not possible in our case, since we do not have the source code of several of them). Clearly, in our case, the emphasis is to identify which approach requires the lowest number of objective function evaluations to find solutions of a certain acceptable quality.

However, the running times are also compared in an indirect manner, to give at least a rough idea of the complexities of the different algorithms considered in our comparative study. For all test problems IA_EDP_S found the best cost in the lowest time. Except for SYS_3U, where fast-PSO just required 0.01 second and IA_EDP_S spent 0.18 seconds to find the best solution.

Table 3 summarizes the performance IA_EDP_S with respect to that of the other methods. As shown in Table 3, considering the best cost found, IA_EDP_S outperforms all other approaches. Considering running times, IA_EDP_S requires less than one second to find solutions with an acceptable quality for SYS_3U and SYS_6U. It requires less than 1.4 second for SYS_15U and SYS_18U. And it requires less than 2.1 second for SYS_20U.

We could not found an approach that report feasible solutions for SYS_20U, so IA_EDP_S obtained the best results.

Table 3. Comparison of results. The best values are shown in **boldface**.

Problem/ Algorithm	Best	Worst	Mean	Std.	Time(s)	Ev.
SYS_3U						
IEP[10]	**8194.35**	-	-	-	-	-
MPSO[9]	**8194.35**	-	-	-	-	-
IPSO [11]	**8194.35**	-	-	-	0.42	3000
ModPSO [12]	8194.40	-	-	-	-	-
fast-CPSO[4]	**8194.35**	-	-	-	0.01	3000
IA_EDP_S	**8194.35**	**8194.37**	**8194.35**	0.004	0.18	987
SYS_18U						
ICA-PSO [13]	25430.16	25462.34	25440.89	-	18.585	40000
IA_EDP_S	**25429.80**	**25433.06**	**25430.93**	0.614	1.168	35103
SYS_6U						
IHS[7]	15444.30	-	15449.86	4.531	-	100000
BBO [3]	15443.09	**15443.09**	**15443.09**	-	-	50000
ICA-PSO[13]	15443.24	15444.33	15443.97	-	-	20000
IA_EDP_S	**15442.89**	15455.24	15444.30	1.877	0.828	1490
SYS_15U						
CCPSO[8]	32704.45	**32704.45**	**32704.45**	0.0	16.2	30000
MDE[1]	32704.9	32711.5	32708.1	-	-	160000
SA-PSO [5]	32708.00	32789.00	32732.00	18.025	12.79	20000
IA_EDP_S	**32700.29**	32865.26	32763.53	35.76	1.328	18321
SYS_20U						
IA_EDP_S	**62476.11**	62636.58	62522.37	30.371	2.016	8151

6 Conclusions and Future Work

This paper presented an adaptation of an algorithm inspired on the T-Cell model of the immune system, called IA_EDP_S, which was used to solve economic dispatch problems. IA_EDP_S is able to handle the five types of constraints that are involved in an economic dispatch problem: power balance constraint with and without transmission loss, operating limit constraints, ramp rate limit constraint and prohibited operating zones.

At the beginning, the search performed by IA_EDP_S is based on a simple differentiation operator which takes an infeasible solution and modifies some of its decision variables by taking into account their constraint violation. Once the algorithm finds a feasible solution, a redistribution power operator is applied. This operator modifies two decision variables at a time, it decreases the power in one unit, and it selects other unit to generate the power that has been taken.

The approach was validated with five test problems having different characteristics and comparisons were provided with respect to some approaches that have been reported in the specialized literature. Our results indicated that dimensionality increases standard deviations when the same types of constraints are considered but prohibited zones have more impact on the performance than dimensionality. Our proposed approach produced competitive results in all cases, being able to outperform the other approaches while performing a lower number of objective function evaluations than the other approaches.

As part of our future work, we are interested in redesigning the redistribution operator in order to maintain the solutions' feasibility when a problem involves prohibited operating zones.

References

1. N. Amjady and H. Sharifzadeh. Solution of non-convex economic dispatch problem considering valve loading effect by a new modified differential evolution algorithm. *International Journal of Electrical Power and Energy Systems*, 32(8):893–903, 2010.
2. V.S. Aragon, S.C. Esquivel, and C.A. Coello Coello. An immune algorithm with power redistribution for solving economic dispatch problems. *Information Sciences*, 295(0):609 – 632, 2015.
3. A. Bhattacharya and P.K. Chattopadhyay. Biogeography-based optimization for different economic load dispatch problems. *IEEE Transactions on Power Systems*, 25(2):1064–1077, 2010.
4. Leticia Cecilia Cagnina, Susana Cecilia Esquivel, and Carlos A. Coello Coello. A fast particle swarm algorithm for solving smooth and non-smooth economic dispatch problems. *Engineering Optimization*, 43(5):485–505, 2011.
5. Cheng-Chien Kuo. A novel coding scheme for practical economic dispatch by modified particle swarm approach. *IEEE Transactions on Power Systems*, 23(4):1825–1835, 2008.
6. Leandro Nunes de Castro and Jonathan Timmis. *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer-Verlag, New York, 2002.

7. V.R. Pandi, K.B. Panigrahi, M.K. Mallick, A. Abraham, and S. Das. Improved harmony search for economic power dispatch. In *Hybrid Intelligent Systems, 2009. HIS '09. Ninth International Conference on*, volume 3, pages 403–408, 2009.
8. J.-B. Park, Y.-W. Jeong, J.-R. Shin, and K.Y. Lee. An improved particle swarm optimization for nonconvex economic dispatch problems. *IEEE Transactions on Power Systems*, 25(1):156–166, 2010.
9. Jong-Bae Park, Ki-Song Lee, Joong-Rin Shin, and K.Y. Lee. A particle swarm optimization for economic dispatch with nonsmooth cost functions. *Power Systems, IEEE Transactions on*, 20(1):34–42, 2005.
10. Y. M. Park, J. R. Won, and J. B. Park. A new approach to economic load dispatch based on improved evolutionary programming. *Eng. Intell. Syst. Elect. Eng. Commun.*, 6(2):103–110, June 1998.
11. P. Sriyanyong, Y.H. Song, and P.J. Turner. Particle swarm optimisation for operational planning: Unit commitment and economic dispatch. In KeshavP. Dahal, KayChen Tan, and PeterI. Cowling, editors, *Evolutionary Scheduling*, volume 49 of *Studies in Computational Intelligence*, pages 313–347. Springer Berlin Heidelberg, 2007.
12. S. Siva Subramani and P. Raja Rajeswari. A modified particle swarm optimization for economic dispatch problems with non-smooth cost functions. *International Journal of Soft Computing*, 3(4):326–332, 2008.
13. J.G. Vlachogiannis and K.Y. Lee. Economic load dispatch a comparative study on heuristic optimization techniques with an improved coordinated aggregation-based pso. *IEEE Transactions on Power Systems*, 24(2):991–1001, 2009.
14. Ling Wang and Ling po Li. An effective differential harmony search algorithm for the solving non-convex economic load dispatch problems. *International Journal of Electrical Power & Energy Systems*, 44(1):832 – 843, 2013.

Predicción de Ingresos de Causas Penales utilizando Programación Genética Lineal

Alberto David Garcete Rodríguez[1] y Benjamín Barán[2]

[1] adgr_x@hotmail.com – [2] bbaran@cba.com.py
Universidad Nacional del Este
Campus Km. 8 Acaray, Calle Universidad Nacional del Este y Rca. del Paraguay
Ciudad del Este – Paraguay

Resumen. Este trabajo propone una metodología de predicción de ingresos de causas penales utilizando Programación Genética Lineal (*Linear Genetic Programming* - LGP). El estudio se realizó en base a datos mensuales recogidos durante siete años (2007 a 2013), en los siete Juzgados Penales de Garantías de Ciudad del Este - Paraguay. La verificación del método propuesto se hizo mediante la comparación del método LGP implementado con modelos estadísticos conocidos como la regresión lineal, promedio móvil, suavizado exponencial y suavizado exponencial con tendencia, prediciendo valores sobre una serie de tiempo, de forma a comparar los errores promedios de cada metodología de predicción. Se utilizaron dos métricas de error: (1) el error cuadrático medio y (2) el error absoluto medio. Resultados experimentales demuestran la superioridad del LGP implementado sobre los demás métodos estadísticos para la predicción de ingresos de causas penales.

Palabras clave: Programación Genética - GP, Programación Genética Lineal - LGP, ingresos penales, predicción.

1 Introducción

El aumento de delitos comunes moviliza todo el aparato judicial afectando el número de juicios en los estrados judiciales que se va incrementando año a año [1] y [2].

Investigar el aumento de delitos resulta beneficioso para mejorar la eficiencia del sistema judicial y analizar los factores que pueden incidir en estos hechos delictivos. En consecuencia, este trabajo propone realizar por primera vez predicciones relativas al número de delitos utilizando un algoritmo evolutivo y compararlo con métodos estadísticos tradicionales como la regresión lineal, promedio móvil, suavizado exponencial y suavizado exponencial con tendencia, para permitir una adecuada predicción que facilite la planificación de los recursos necesarios para que el Poder Judicial pueda cumplir con sus funciones específicas, sean estos: recursos humanos, espacio físico o insumos, entre otras necesidades específicas. El trabajo fue elaborado en base a datos históricos que corresponden a un periodo de siete años transcurrido entre enero/2007 y diciembre/2013.

2 Trabajos previos

Azamathulla, Guven y Demir presentan en [3] la metodología LGP, como una herramienta alternativa en la predicción de profundidad para tuberías sumergidas. El modelo LGP propuesto se comparó con sistemas adaptativos de inferencia *neuro-fuzzy* (ANFIS), observando que las predicciones utilizando LGP logran resultados mejores que los obtenidos con ANFIS o con una tradicional ecuación de regresión.

En [4], Guven y Kisi estudiaron los datos climáticos diarios para estimar la evaporación utilizando la programación genética lineal. Las estimaciones del modelo LGP se compararon con los resultados de otras técnicas de predicción, concluyéndose que la técnica LGP es apropiada para el modelado del proceso de evaporación.

En [5], Shavandi y Ramyani utilizaron LGP para predecir la radiación solar global. Los resultados indicaron que los modelos LGP dan estimaciones precisas de la radiación solar global y superan significativamente al modelo tradicional.

A su vez, Alavi, Gandomi y Mollahasani presentan en [6] un modelo del LGP así como un algoritmo híbrido de búsqueda que combinaba el algoritmo LGP con *Simulated Annealing* (SA), denominado LGP/SA, logrando predecir características de rendimiento de un suelo estabilizado. En particular, los modelos basados en LGP encontraron valores más precisos que los modelos híbridos basados en LGP/SA.

3 Programación Genética (GP)

La computación evolutiva propone un conjunto de modelos computacionales, que desarrollan automáticamente programas de computadoras que evolucionan generación tras generación resolviendo determinados problemas [7]. En este contexto, la Programación Genética GP, es una técnica de aprendizaje automático que se utiliza para optimizar una *población* de programas utilizando una función aptitud (*fitness*) [11], responsable de evaluar la calidad de cada programa.

La potencialidad de la programación genética reside en que permite desarrollar programas de forma automatizada, una vez definido el objeto de este programa y la métrica de comparación entre diferentes alternativas (o soluciones). Su base biológica es la misma que la de los algoritmos genéticos, es decir, el principio de evolución de las especies propuesto por Darwin [12]. Las primeras propuestas de GP utilizaban árboles sintácticos aplicados a un lenguaje de programación para representar a los individuos [10]. Posteriormente, aparecieron versiones más sencillas de implementar, como la Programación Genética Lineal (*Linear Genetic Programming* - *LGP*) [3], tratado en la siguiente sección.

4 Programación Genética Lineal (LGP)

En la metodología LGP los programas son representados como una secuencia de instrucciones de un lenguaje imperativo de programación; un ejemplo sería:

r[0] = r[2] + r[6]

En este ejemplo, **r[i]** representa al registro **i**. El registro **r[0]** es el registro destino y los registros **r[2]** y **r[6]** son los registros donde están los *operandos* de la operación realizada, en este caso una suma. Para este trabajo, las instrucciones están compuestas de la siguiente forma:

- Un registro destino donde se almacena el resultado.
- Uno o dos registros *operandos* donde se almacenan los valores con los que se operará.
- Una operación (de un conjunto de operaciones definidas a priori).

Entonces una representación típica de un programa obtenido por un algoritmo LGP podría ser de la siguiente forma:

r[6] = r[3] + r[1]
r[1] = seno r[15]
r[0] = r[1] / r[2]

El Algoritmo 4.1 que sigue fue diseñado para atender las necesidades propias del problema de predicción de causas penales estudiado en el presente trabajo. Para ello, se ha realizado una adaptación del algoritmo presentado por Guven y Kisi en [4].

Algoritmo 4.1: ***Programación Genética Lineal implementada***

1: Inicializar aleatoriamente una población de programas.
2: Seleccionar los programas de la población y compararlos de acuerdo a su *fitness, usando* el *SelectorRuleta* [9].
3: Modificar los programas (individuos) utilizando uno o más de los siguientes operadores de variación:
•**Reproducción**: copia un programa sin cambiarlo.
•**Cruzamiento**: intercambia subestructuras (genes) entre 2 programas (o individuos).
•**Mutación**: se utilizan dos tipos de mutaciones. La *micro-mutación* que modificar un único elemento (registro u operador) de la instrucción mutada, y la *macro-mutación* que inserta o borra una instrucción completa.
4: Se construye una nueva población con los programas variados en el paso 3 y se calcula el *fitness de cada nuevo programa*.
5: Si el criterio de parada no se cumple, volver al paso 2.
6: Parar: el programa con mejor *fitness* en la última población representa la solución encontrada por el LGP implementado.

El conjunto de registros junto con el conjunto de operaciones o funciones son los componentes a partir de los cuales la metodología LGP, trata de construir un programa de computador para solucionar un problema. El número de registros se define a priori [8] y para este trabajo solo se utilizaron 22 registros. El registro r[0] es el que almacena la salida del algoritmo. Los registros r[10] al r[15] son de entrada, conteniendo información introducida desde la fuente de datos conocida, típicamente datos de la serie temporal estudiada. El conjunto de funciones escogidas para este trabajo incluye 4 funciones aritméticas, 3 exponenciales y 2 trigonométricas.

En este trabajo, la aptitud [9] de un programa se estima ponderando el error que se produce al predecir un valor futuro utilizando un programa generado por el algoritmo LGP (individuo cuyo *fitness* se desea estimar). Entonces, el mejor individuo (o programa) será aquel que tenga el menor error de predicción.

La función de error más utilizada es el error cuadrático medio [8], dado en (1). Así, se definió el error "e" dado por (1) como el promedio de los errores cuadrados para "N" valores de la muestra, donde "x_i" representa el valor real de la muestra "i" mientras que "p_i" representa el pronóstico (o estimación) para el instante discreto "i".

$$e = \frac{1}{N}\sum_{i=1}^{N}(x_i - p_i)^2 \qquad (1)$$

Cabe recordar además que las operaciones genéticas son aplicadas sobre una población que evoluciona de generación en generación. En consecuencia, se deben definir los parámetros que controlarán la ejecución del algoritmo LGP, como ser: el tamaño de la población (150 individuos), la longitud máxima de un individuo (200 instrucciones), las probabilidades de aplicación de los operadores genéticos de cruce y mutación (20% y 30% respectivamente); entre otros parámetros [3], cuyos detalles no se presentan por falta de espacio.

5 Métodos Clásicos de Predicción

Este trabajo utiliza también métodos estadísticos de predicción de series de tiempo para compararlos con LGP implementado. Estos métodos son brevemente presentados a continuación.

5.1 Regresión Lineal

En una regresión lineal, se asume que existe una relación entre una variable **x** (independiente) y otra variable **y** (dependiente) que se representa como una recta [13].

$$\mathbf{y = mx + b} \qquad (2)$$

Los valores de "b" y "m" se eligen de manera a minimizar el error dado en (1).

5.2 Promedio Móvil

Es el promedio aritmético de las "N" observaciones más recientes [13], dado por:

$$\mathbf{S_t = \frac{1}{N}\sum_{i=t}^{t-N+1} D_i} \qquad (3)$$

donde "$\mathbf{D_i}$" representa el dato disponible para el instante de tiempo "i". El inconveniente de este modelo es la pérdida de datos iniciales de la serie original [13].

5.3 Suavizado Exponencial

Es uno de los métodos más utilizados [14]. Utiliza un mecanismo de corrección que ajusta los pronósticos, donde las ponderaciones disminuyen exponencialmente y está dado por:

$$\mathbf{F_t} = \alpha\mathbf{D_{t-1}} + (\mathbf{1} - \alpha)\mathbf{F_{t-1}} \qquad (4)$$

donde α representa la constante de suavizado que determina la ponderación relativa en la observación. Alternativamente, si existe una clara tendencia en los datos de la sería temporal, este método puede ser mejorado estimando esta tendencia, en lo que se da en llamar *Suavizado Exponencial con Tendencia*, que se presenta a continuación.

5.4 Suavizado Exponencial con Tendencia

Además de utilizar el parámetro α arriba presentado, este método requiere un segundo parámetro β que suavice la tendencia. En la mayoría de las aplicaciones [14], se utiliza:

$$\beta \leq \alpha \qquad (5)$$

Las ecuaciones usualmente utilizadas con el método suavizado exponencial con tendencia son [14]:

$$\mathbf{S_t} = \alpha\mathbf{D_{t-1}} + (\mathbf{1} - \alpha)(\mathbf{S_{t-1}} + \mathbf{T_{t-1}}) \qquad (6)$$

$$\mathbf{T_t} = \beta(\mathbf{S_t} - \mathbf{S_{t-1}}) + (\mathbf{1} - \beta)\mathbf{T_{t-1}} \qquad (7)$$

$$\mathbf{F_t} = \mathbf{S_t} + \mathbf{T_t} \qquad (8)$$

6 Pruebas Experimentales y Principales Resultados

Las pruebas experimentales fueron realizadas comparando el método LGP, con los métodos estadísticos seleccionados (regresión lineal, promedio móvil, suavizado exponencial y suavizado exponencial con tendencia) [13] y [14] utilizando los datos disponibles para el periodo comprendido entre enero del 2007 y diciembre del 2013, extraídos del sitio web de la Excma. Corte Suprema de Justicia [1] y [2]. Las pruebas se realizaron con la siguiente distribución de datos:

Entrenamiento o prueba	:	72 meses.
Validación o Pronóstico	:	12 meses.
Total de datos disponibles	:	84 meses (enero/2007 a diciembre/2013)

Las evaluaciones de las metodologías implementadas se realizaron considerando dos métricas de desempeño: (1) el error cuadrático medio y (2) el error absoluto medio dado por:

$$e = \frac{1}{N}\sum_{i=1}^{N} |x_i - p_i| \qquad (9)$$

Procedimiento utilizado con el método LGP:

- La población de individuos es de estado de reposo, donde unos padres son reemplazados por los hijos a cada generación.
- El algoritmo realiza la selección de individuos utilizando el método de la ruleta [9]. A los individuos seleccionados se les aplican los operadores evolutivos (selección, cruzamiento y mutación), para generar los individuos de la siguiente población.
- El pronóstico se realizó con los últimos 12 meses, aplicando la regla obtenida con el mejor individuo generado por el algoritmo LGP utilizando los 72 meses de prueba como entrenamiento.

Reglas obtenidas con el método LGP

La aleatoriedad del método LGP para obtener soluciones distintas en cada corrida, ha generado buenos resultados en la mayoría de las ejecuciones realizadas. Como ejemplo, la Tabla 1 presenta tres reglas (o programas de predicción) obtenidas por el algoritmo LGP que fueron comparados con los métodos estadísticos arriba presentados.

Tabla 1. Tabla con tres individuos (o programas) propuestos por el algoritmo LGP implementado en este trabajo

Individuo A	***Individuo B***	***Individuo C***
r[9] = r[15] - r[21]	r[6] = |r[17]|	r[4] = r[14] + r[6]
r[0] = r [7] + r[0]	r[6] = r [18] ^ r[3]	r[9] = r[6] * r[4]
r[1] = r[15] / r[19]	r[3] = sen[12]	r[5] = r[7] + r[17]
r[7] = r[1] + r[1]	r[6] = r[16] * r[11]	r[3] = r[13] + r[10]
r[0] = |r[9]|	r[6] = r[17] - r[21]	r[3] = sen[15]
r[8] = r[21] + r[0]	r[2] = r[5] - r[7]	r[5] = r[7] + r[17]
r[8] = r[20] + r[0]	r[6] = cosen[9]	r[4] = |r[10]|
r[0] = |r[9]|	r[0] = |r[9]|	r[0] = r [13] ^ r[5]
r[0] = r [8] + r[1]	r[3] = sen[18]	r[8] = r[8] / r[7]
r[0] = r [7] + r[0]	r[3] = ln[14]	r[0] = r[7] * r[2]
r[8] = r[21] + r[0]	r[4] = ln[6]	r[0] = r[1] - r[0]
r[0] = r [7] + r[0]	r[9] = sen[13]	r[4] = cosen[5]
r[8] = r[21] + r[0]	r[9] = ln[16]	r[6] = r[13] - r[1]
r[0] = r [7] + r[0]	r[5] = r[10] * r[7]	r[7] = r [16] + r[16]
r[8] = r[21] + r[0]	r[5] = |r[16]|	r[3] = cosen[16]
r[0] = r [8] + r[1]	r[0] = cosen[11]	r[4] = r[4] * r[6]
r[7] = r[1] + r[1]	r[5] = cosen[19]	r[0] = r[7] * r[2]
r[0] = r [7] + r[0]	r[4] = |r[15]|	r[4] = r [14] + r[6]
r[8] = r[21] + r[0]	r[4] = |r[19]|	r[5] = r [18] + r[16]
r[0] = r[19] * r[8]	r[8] = r[14] * r[20]	r[7] = r [16] + r[16]
	r[2] = sen[21]	r[0] = r [19] + r[12]
	r[7] = r[3] / r[5]	
	r[4] = sen[14]	
	r[2] = r[7] / r[4]	
	r[7] = ln[3]	
	r[0] = r[2] - r[5]	
	r[3] = sen[0]	
	r[7] = r[8] + r[9]	
	r[0] = r[7] * r[20]	

La Tabla 2 presenta los errores de predicción para los 12 meses el año 2013 utilizando el error cuadrático medio. En las últimas filas se presentan: el error total, el error medio y el *Ranking* correspondiente, verificándose el excelente desempeño de los individuos LGP(A) y LGP(B) calculado por el LGP implementado, y destacando que el LGP(C) es el peor de todos, lo que demuestra que no siempre el LGP calcula una buena regla de estimación, debido a su aleatoriedad inherente.

Tabla 2. Error cuadrático medio para cada solución evaluada durante el periodo de validación, ordenados con los mejores resultados a la izquierda y los peores a la derecha (ver fila: *Ranking*)

Mes	LGP(A)	LGP(B)	S. Exp.	S. Exp. Tend.	Regresión	P. Móvil	LGP(C)
Enero	14361,47	35628,71	67667,37	52617,80	59413,70	81035,11	47552,66
Febrero	3764,75	6809,11	6573,60	316,19	28828,59	3927,11	15892,57
Marzo	19565,15	9093,21	24134,67	47484,53	1100,22	35344,00	76139,73
Abril	10631,37	408,67	3506,06	7544,58	3,50	16129,00	27867,05
Mayo	1002,25	7740,42	9307,00	6562,36	18201,12	7511,11	28583,22
Junio	4022,77	15009,00	7548,77	3351,48	27209,20	16469,44	26917,56
Julio	33926,80	65216,27	28944,29	17540,12	80651,84	32881,78	22519,72
Agosto	17413,24	34560,86	2216,35	98,19	52915,29	1111,11	4364,68
Setiembre	32860,20	15342,73	82801,86	116682,34	7212,46	98177,78	135375,64
Octubre	461,95	6413,29	747,92	658,03	13482,55	841,00	12754,15
Noviembre	4321,74	71,97	5415,65	6725,72	737,39	3844,00	12783,86
Diciembre	2118,48	8396,44	4204,92	4523,74	18822,61	13378,78	486,90
TOTAL Error:	144450,19	204690,68	243068,46	264105,08	308578,47	310650,22	411237,72
Error Medio:	12037,52	17057,56	20255,70	22008,76	25714,87	25887,52	34269,81
Ranking	1°	2°	3°	4°	5°	6°	7°

Al utilizar la métrica del error absoluto medio para comparar las mismas siete soluciones en el mismo año 2013, se obtiene la Tabla 3, donde nuevamente el LGP es el método que encuentra las mejores estimaciones con los individuos LGP(A) y LGP(B) que consistentemente resultan así los mejores en el Ranking, por lo que serían los recomendados por este trabajo para realizar predicciones del número de causas judiciales en los Juzgados de Ciudad del Este – Paraguay.

Finalmente, se hace notar que según cuál sea la métrica de comparación (error cuadrático medio o error absoluto medio), los métodos *Suavizado Exponencial* y *Suavizado Exponencial con Tendencia* pueden quedar terceros o cuartos en el Ranking. Algo similar ocurre con el *Promedio Móvil* y la *Regresión Lineal*, que pueden quedar quinto o sexto, pero sin ninguna variación en los primeros puestos del Ranking que consistentemente corresponden a los individuos LGP(A) y LGP(B) calculados con el LGP implementado. Así mismo, el último puesto en ambos Rankings corresponde al individuo LGP(C), confirmando que un LGP no siempre realiza buenas predicciones, por lo que resulta recomendable realizar varias corridas del método antes de escoger la solución que será efectivamente utilizada.

Tabla 3. Error absoluto medio para cada solución evaluada durante el periodo de validación, ordenados con los mejores resultados a la izquierda y los peores a la derecha (ver fila: *Ranking*)

Mes	LGP(A)	LGP(B)	S. Exp. Tend.	S. Exp.	P. Móvil	Regresión	LGP(C)
Enero	119,84	188,76	229,39	260,13	284,67	243,75	218,07
Febrero	61,36	82,52	17,78	81,08	62,67	169,79	126,07
Marzo	139,88	95,36	217,91	155,35	188,00	33,17	275,93
Abril	103,11	20,22	86,86	59,21	127,00	1,87	166,93
Mayo	31,66	87,98	81,01	96,47	86,67	134,91	169,07
Junio	63,43	122,51	57,89	86,88	128,33	164,95	164,07
Julio	184,19	255,37	132,44	170,13	181,33	283,99	150,07
Agosto	131,96	185,91	9,91	47,08	33,33	230,03	66,07
Setiembre	181,27	123,87	341,59	287,75	313,33	84,93	367,93
Octubre	21,49	80,08	25,65	27,35	29,00	116,11	112,93
Noviembre	65,74	8,48	82,01	73,59	62,00	27,15	113,07
Diciembre	46,03	91,63	67,26	64,85	115,67	137,20	22,07
TOTAL Error:	1149,95	1342,68	1349,69	1409,88	1612,00	1627,86	1952,26
Error Medio:	95,83	111,89	112,47	117,49	134,33	135,66	162,69
Ranking	1°	2°	3°	4°	5°	6°	7°

7 Aportes al Poder Judicial

Un punto relevante determinado por este estudio fue la tendencia creciente de los ingresos penales (ver Figura 1). El mismo algoritmo LGP puede de hecho ser utilizado para analizar esta preocupante situación.

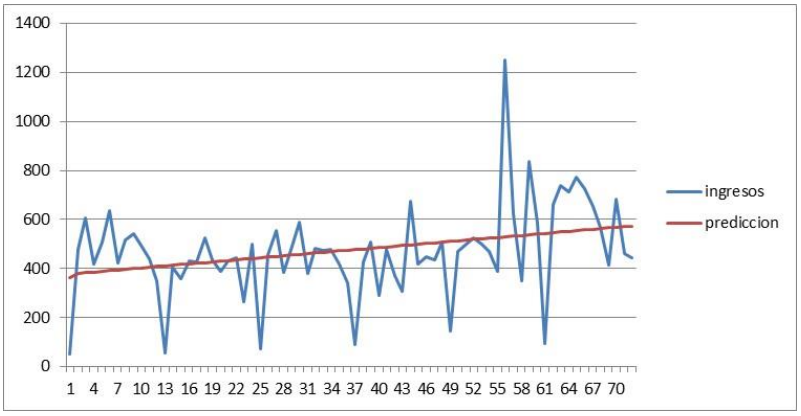


Fig. 1. Tendencia de crecimiento de los ingresos

La cantidad de causas estudiadas durante los 7 años son 38.305. El promedio anual de estas causas, arroja que cada despacho judicial estaría trabajando con un total aproximado de 5472 expedientes. Los datos del estudio demuestran que más de 6000 causas penales ingresaron en los últimos años y la tendencia es de un constante cre-

cimiento lo que significa que en un tiempo próximo, las autoridades deberán habilitar más juzgados penales de garantías en la localidad en cuestión, lo que conllevaría un gasto considerable, en base al costo que se genera en cada juzgado.

En consecuencia, este tipo de estudio realizado en diversas localidades aportaría a la Corte Suprema de Justicia con información relevante para ampliar en forma planificada los recursos que sean necesarios, minimizando así los contratiempos y demoras que hoy se vislumbra. Así, la herramienta propuesta permitirá advertir a las altas autoridades de las necesidades y tendencias antes de llegar a una situación extrema, proyecto con tiempo una solución adecuada, por ejemplo, invirtiendo más en prevención.

8 Conclusiones y trabajos futuros

El algoritmo LGP fue ejecutado varias veces y en la mayoría de las ejecuciones generaba resultados muy buenos al compararlos con otros métodos estadísticos más populares. Observando estos resultados y comparándolos con cada modelo matemático, se notó que las soluciones simulan y logran representar adecuadamente a las soluciones obtenidas por estos métodos matemáticos, con la diferencia relevante que el LGP logra generalmente mejores resultados.

El algoritmo LGP demuestra ser apto para ser aplicado como una metodología de predicción de los ingresos de causas penales de los Juzgados de Garantías de Ciudad del Este por la eficiencia obtenida en sus resultados, inclusive pudiendo ser extendido su aplicación para otras regiones del país. El éxito del LGP radica en que los resultados que genera se mantienen con un promedio de error bajo, quedando siempre entre los primeros lugares en cada predicción realizada. Cabe destacar que según sea la métrica de error considerada, el *Ranking* de comparación entre algoritmos presentados en las Tablas 2 y 3 puede diferir, por lo que no se logra establecer un único *Ranking.* Con todo, consistentemente, las mejores alternativas fueron siempre calculadas con el algoritmo LGP.

Las variantes que se le pueden aplicar al algoritmo LGP para su ejecución permiten obtener resultados experimentales exitosos para modelos de predicciones tanto lineales como no lineales, otro de los motivos por los que un LGP genera soluciones eficientes en comparación a los métodos estadísticos clásicos que demuestran limitaciones para predicciones de series de tiempos no lineales [8].

Por último, observando el crecimiento de los delitos y el incremento en la estructura de recursos humanos de la Excma. Corte Suprema de Justicia [1] y [2], se puede mencionar que con este tipo de trabajo, se podría realizar una reingeniería de recursos, alocando infraestructura y recurso humano donde más se los necesite, con razonable previsión. En consecuencia, aplicando este tipo de técnicas se obtendrían mejores resultados en el ámbito de la justicia, ya que ajustando los procesos se podrá ofrecer un mejor servicio en el área en la que se encuentra especializada, LA JUSTICIA.

Como posibles trabajos futuros para continuar con lo hasta aquí realizado, se propone:

1. Análisis de los parámetros óptimos de inicialización y de los valores de entrada a ser utilizados por un algoritmo LGP.
2. Comparación del algoritmo LGP con otros modelos no lineales de predicción, como las redes neuronales.
3. Estudiar problemas con múltiples salidas.
4. Implementación del algoritmo LGP con enfoques multi-objetivo.
5. Considerar aplicaciones concretas a otros juzgados y diferentes áreas del sector público que necesitan realizar estimaciones para mejorar su planificación en base a predicciones cada vez mejores.

Referencias

1. http://www.csj.gov.py/, Sitio De La Excma. Corte Suprema De Justicia, 2014.
2. http://www.pj.gov.py/, Sitio del Poder Judicial, 2014.
3. Azamathulla, H. M., Guven, A., & Demir, Y. K. Linear genetic programming to scour below submerged pipeline. Ocean Engineering, 38(8), 995-1000, 2011.
4. Guven, A., & Kişi, Ö. Daily pan evaporation modeling using linear genetic programming technique. Irrigation science, 29(2), 135-145, 2011.
5. Shavandi, H., & Ramyani, S. S. A linear genetic programming approach for the prediction of solar global radiation. Neural Computing and Applications, 23(3-4), 1197-1204, 2013.
6. Alavi, A. H., Gandomi, A. H., & Mollahasani, A. A Genetic Programming-Based Approach for the Performance Characteristics Assessment of Stabilized Soil. In Variants of Evolutionary Algorithms for Real-World Applications (pp. 343-376). Springer Berlin Heidelberg. 2012.
7. Koza, J. R. Human-competitive results produced by genetic programming. Genetic Programming and Evolvable Machines, 11(3-4), 251-284, 2010.
8. R. Sánchez, J. Martinez y B. Barán. "Economic Time-Series Forecasting Using Linear Genetic Programming", Computational Intelligence in Economics and Finance - CIEF. 11th Joint Conference on In-formation Sciences JCIS' Kylin Villa – Shenzhen, China, 2008.
9. Koza, J. R. Herarchical Genetic Algorithms Operating on Populations of Computer Programs. N. S. Sridharan, Editor, Proceedings Of 11th International Joint Conference On Artificial Intelligence, San Mateo, Morgan Kaufmann, California, 1989.
10. Koza, J. R. Introduction to genetic programming tutorial: from the basics to human-competitive results. Proceedings of the 12th annual conference companion on Genetic and evolutionary computation, 2137-2262, Julio 2010. ACM.
11. Poli, R., & Koza, J. Genetic Programming, 143-185. Springer US, 2014.
12. Darwin, Ch. The Origin of Species By Means Of Natural Selection or the Preservation of Favored Races in the Struggle for Life. Random House, New York, 1993.
13. Hernández, R. Estadística, Segunda Edición, Madrid España, 1997.
14. Triola, Mario F., Estadística Elemental, Séptima Edición, México 2000.

Optimización basada en Colonias de Hormigas para el Problema del Vendedor Viajante con muchos objetivos contradictorios

Francisco Riveros[1], Néstor Benítez[1], Julio Paciello[1] y Benjamín Barán[1,2]

[1] Universidad Nacional de Asunción
{friverosn,nestorbdietrich,juliopaciello}@gmail.com,
bbaran@pol.una.py

[2] Universidad Nacional del Este

Abstract. Considerando el conocido problema de disminución de rendimiento que presentan los algoritmos evolutivos cuando resuelven problemas denominados *many-objective*, este trabajo propone la utilización de una variante de la Optimización basada en Colonias de Hormigas que denominamos λ base-p. La nueva propuesta fue sometida a distintas pruebas experimentales sobre instancias del Problema del Vendedor Viajante con muchos objetivos utilizando la métrica del Hipervolumen. La Asignación de λ base-p fue comparada con algoritmos MOACO del estado del arte y con el algoritmo evolutivo NSGA2, demostrando que logra calcular un mejor Hipervolumen cuando se resuelven problemas de muchos objetivos contradictorios (*many-objective*).

Keywords: MOACO, TSP, *many-objective*, Hipervolumen, NSGA2

1 Introducción

Los algoritmos *Multi-Objective Ant Colony Optimization* (MOACO) fueron tradicionalmente utilizados para resolver problemas con múltiples objetivos contradictorios, denominados *Multi-Objective Optimization Problems* (MOP), considerados como uno de los métodos con mejor desempeño para resolver el Problema del Vendedor Viajante (TSP-*Travelling Salesman Problems*) [1] . Trabajos anteriores como [2] , [3] y [4] utilizaron los algoritmos MOACO para resolver los denominados MOP del mundo real, y lo hicieron de manera efectiva para 2 o 3 objetivos. Sin embargo, esta efectividad puede verse afectada cuando el número de objetivos aumenta, como pasa con la mayoría de los algoritmos evolutivos [7]. La problemática tratada en este trabajo es el desempeño de los algoritmos MOACO cuando son utilizados para resolver problemas *many-objective* (problemas con 4 o más objetivos contradictorios). En este contexto, se implementó una nueva estrategia de MOACO para resolver problemas *many-objective* ya que las implementaciones actuales no contemplan resolver estos problemas con muchos objetivos. En las siguientes secciones se describe el TSP *many-objective*,

los algoritmos MOACO considerados en el presente trabajo, la estrategia propuesta por este trabajo para considerar problemas *many-objective* y finalmente, se presentan los resultados experimentales, las conclusiones y trabajos futuros.

2 TSP *many-objective*

El TSP puede ser representado por un grafo ponderado completamente conexo $G = (N, A)$, donde N representa un conjunto de $c = |N|$ nodos y A un conjunto de aristas que conectan los nodos de N. En un caso mono-objetivo, cada arista tiene asociado una función de costo d_{ij}, que representa la distancia entre los nodos i y j. El TSP consiste en encontrar el ciclo Hamiltoniano que minimice la distancia recorrida desde un nodo inicial, pasando por cada nodo exactamente una vez y volviendo finalmente al nodo inicial [1]. Para formular el TSP se puede considerar una variable dicotómica $x_{i,j}$ para todo $(i, j) \in A$, de forma que tome el valor 1 si el arco forma parte del ciclo Hamiltoniano y 0 en otro caso, entonces, el TSP consiste en:

$$minimizar \sum_{(i,j)\in A} d_{i,j}x_{i,j} \tag{1}$$

Para el caso del TSP *many-objective*, son consideradas k funciones de costos, teniendo cada arista (i, j) asociado un conjunto de k distancias $d^1_{ij}, d^2_{ij}..., d^k_{ij}$. El problema consiste en minimizar las k funciones de costos al mismo tiempo [8], es decir:

$$minimizar \left[\sum_{(i,j)\in A} d^1_{i,j}x_{i,j}; \sum_{(i,j)\in A} d^2_{i,j}x_{i,j}; ...; \sum_{(i,j)\in A} d^k_{i,j}x_{i,j} \right] \tag{2}$$

3 Algoritmos MOACO

Los algoritmos *Ant Colony Optimization* (ACO) [12] se inspiran en el comportamiento de colonias de hormigas reales para resolver problemas de optimización combinatoria. Se basan en una colonia de hormigas artificiales, es decir, colonia de agentes computacionales simples que trabajan de manera cooperativa y se comunican mediante rastros de feromonas artificiales. Para aplicar un ACO al problema aquí tratado, se utiliza un conjunto de agentes llamados hormigas. Las hormigas construyen las soluciones del problema recorriendo el grafo $G = (N, A)$, partiendo desde un nodo inicial. Para cada cambio de nodo, una hormiga considera los parámetros de visibilidad $(\eta_{i,j})$ y feromonas $(\tau_{i,j})$ de cada arco utilizando una política probabilística que está dada por la siguiente fórmula:

$$p_{i,j} = \begin{cases} \frac{\tau^{\alpha}_{i,j}\eta^{\beta}_{i,j}}{\sum_{x\in J_i}\tau^{\alpha}_{i,x}\eta^{\beta}_{i,x}} & \text{si } j \in J_i \\ 0 & \text{en otro caso} \end{cases} \tag{3}$$

donde J_i representa a todos los posibles nodos que pueden ser visitados desde el nodo i (esto es, los nodos que todavía no han sido visitado por una hormiga);

mientras que α y β son parámetros definidos a priori que reflejan la importancia relativa de las feromonas y la visibilidad respectivamente. La actualización de las feromonas está dado por:

$$\tau_{i,j} = (1-\rho)\tau_{i,j} + \rho\Delta\tau \tag{4}$$

donde ρ es el coeficiente que se utiliza para representar la evaporación y $\Delta\tau$ es la cantidad de feromonas que se va sumando a los arcos que forman parte de una solución escogida.
Un MOACO es una extensión de la metaheurística ACO para resolver problemas multi-objetivo. En general, las ecuaciones se modifican, conforme se muestra a continuación [4]:

$$p_{i,j} = \begin{cases} \frac{\tau_{i,j}^{\alpha}(\eta_{i,j}^{1}\eta_{i,j}^{2}\ldots\eta_{i,j}^{k})^{\beta}}{\sum_{x\in J_i}\tau_{i,x}^{\alpha}(\eta_{i,x}^{1}\eta_{i,x}^{2}\ldots\eta_{i,x}^{k})^{\beta}} & \text{si } j \in J_i \\ 0 & \text{en otro caso} \end{cases} \tag{5}$$

$$\Delta\tau = \frac{1}{\sum_{l=1}^{k}\sum_{(i,j)\in A} d_{i,j}^{l} x_{i,j}} \tag{6}$$

donde k es la cantidad de objetivos. En el Algoritmo 1 se puede observar el pseudo-código de un MOACO genérico:

Algoritmo 1. Pseudo-código de un MOACO genérico .

```
procedure MOACO()
   inicializarParametros()
   while not condicionParada()
       generacion = generacion + 1
       for ant = 1 to m //m es la cantidad de hormigas
             construirSolucion()
             evaluarSolucion()
             actualizarCPFeromonas() // definido en (4)
       end for
    end while
return ConjuntoPareto
end procedure

procedure construirSolucion()
   sol = {}
   while existenEstadosNoVisitados()
        siguiente = seleccionarSiguienteEstado() // definido en (5)
        sol = sol U siguiente
        marcarVisitado(siguiente)
        if (actualizacionPasoAPaso)
           actualizarFeromonasPasoAPaso() // por ejemplo, usando (4)
        end if
   end while
end procedure
```

Actualmente existen diversas implementaciones de MOACOs que son objeto de constantes investigaciones [5], [6]. Cada implementación incluye diferentes enfoques de actualización de feromonas, cantidad de tablas de feromonas, cantidad de hormigas, etc. Para este trabajo se seleccionaron tres algoritmos MOACO del estado del arte: el *Multi-objective Ant System* (MAS) , el *Multi-objective Ant Colony System* (MOACS) y el *Multi-objective Max-Min Ant System* (M3AS), en base a su buen comportamiento resolviendo el problema TSP experimentalmente demostrado en [4]. Cabe recordar que en un MOACO, cada hormiga construye una solución mediante la transición de estados dado por ejemplo en (5). En el caso específico del MOACS, se utiliza la selección pseudo-aleatoria dada por (7) que considera el uso de la probabilidad de transición ilustrada en la ecuación (8), que también es utilizada como regla de transición por las variantes M3AS y MAS.

$$j = \begin{cases} max_{j \in J_i}\{\tau_{i,j}^{\alpha}(\eta_{i,j}^{1\lambda_1}\eta_{i,j}^{2\lambda_2}...\eta_{i,j}^{k\lambda_k})^{\beta}\} & \text{si } q < q_0 \\ \hat{\text{p}} & \text{en otro caso} \end{cases} \tag{7}$$

donde la variable $\hat{\text{p}}$ se calcula mediante la probabilidad $p_{i,j}$ que se define como:

$$p_{i,j} = \begin{cases} \frac{\tau_{i,j}^{\alpha}(\eta_{i,j}^{1\lambda_1}\eta_{i,j}^{2\lambda_2}...\eta_{i,j}^{k\lambda_k})^{\beta}}{\sum_{x \in J_i}\tau_{i,x}^{\alpha}(\eta_{i,x}^{1\lambda_1}\eta_{i,x}^{2\lambda_2}...\eta_{i,x}^{k\lambda_k})^{\beta}} & \text{si } j \in J_i \\ 0 & \text{en otro caso} \end{cases} \tag{8}$$

y las variables λ_1 a λ_k son parámetros utilizados para ponderar la importancia relativa de cada objetivo. Típicamente, cada hormiga puede utilizar un conjunto diferente de parámetros λ_1 a λ_k de forma a guiar la búsqueda de cada hormiga a regiones diferentes del espacio objetivo, lo que explica el reconocido éxito del algoritmo MOACS [4]. Todos los MOACOs implementados fueron objetos de varias pruebas experimentales utilizando las instancias del TSP *many-objective* generadas considerando 2, 4, 8 y 10 objetivos. Las pruebas experimentales fueron ejecutadas en un contexto de minimización simultánea de todas las funciones objetivos (costos). Luego de un análisis previo, realizado a las implementaciones de MOACOs aplicados al TSP *many-objective*, se pudo observar: (1) un deterioro de sus rendimientos al aumentar el número de funciones objetivos; y (2) el MOACS resultaba levemente superior a las demás alternativas. En consecuencia, se decidió modificar los parámetros de ejecución del MOACS de forma a hacerlo aún más competitivo al resolver problemas *many-objective*.

4 Métrica de rendimiento

Para la comparación de los resultados experimentales, fue utilizada la métrica del Hipervolumen [9], reconociendo su amplia adopción entre los investigadores del área de optimización multi-objetivo [7]. El Hipervolumen considera el tamaño de la región de dominancia en el espacio objetivo combinando las métricas de distancia, distribución y extensión en un solo valor [9]. Dado un conjunto de soluciones $Z = \{z_1, z_2, ..., z_R\}$ con R soluciones, y un punto de referencia y_{ref},

el Hipervolumen se define como [9]:

$$H(Z) = H(Z, y_{ref}) = \Lambda(\bigcup_{i=1}^{R} H(z_i)) \tag{9}$$

El Hipervolumen de un Frente Pareto X se define como la unión de las porciones que están limitadas por el punto de referencia y_{ref} denominado punto anti-óptimo o peor posible y los puntos del Frente Pareto evaluado. Considerando un problema de dos funciones objetivos con dos Frentes Pareto conocidos $X = \{a, b, c, d, e\}$, $X' = \{a', b', c', d', e'\}$ y el punto de referencia y_{ref}, si el Hipervolumen de X es mayor al de X', entonces, el Frente Pareto X no es peor que el Frente Pareto X' [10].

5 Implementación propuesta

En las reglas de transición de las fórmulas (7) y (8) podemos notar un parámetro de gran importancia que se utiliza para asignar una cierta prioridad a cada uno de los objetivos tenidos en cuenta, este parámetro es denominado λ. Típicamente, el parámetro λ es un valor complementario; de modo que si se aumenta para un objetivo, se debe disminuir para del otro y viceversa. Con esta asignación de λ se busca ponderar más la información de un objetivo que la de otro para así especializarse en una zona del espacio de búsqueda. Por ejemplo, para problemas bi-objetivos [2], se puede escoger los valores de λ_1 y λ_2 utilizando la siguiente relación:

$$\lambda_2 = m - \lambda_1 + 1 \tag{10}$$

donde m representa el número de hormigas y λ_1 toma valores enteros entre 1 y m. Cabe mencionar que para un problema de k objetivos $f_1, f_2, ..., f_k$, se debe asignar valores a $\lambda_1, \lambda_2, ..., \lambda_k$ por cada hormiga, donde λ_1 pondera a f_1, λ_2 pondera a f_2 y así sucesivamente. En general, cada λ toma uno de m valores posibles por lo que se necesitaran k^m hormigas para tener todas las posibles permutaciones, lo cual resulta inviable al crecer el número de objetivos k. Para abordar esta problemática, este trabajo propone una nueva estrategia denominada Asignación de λ base-p.

5.1 Asignación de λ base-p

Esta nueva estrategia aplica una restricción a los valores posibles que puede tomar cada λ que queda así restringido por un parámetro $p \in \mathbb{N}$. Dados los objetivos $f_1, f_2, ..., f_k$, donde k es la cantidad de objetivos, cada hormiga debe escoger aleatoriamente sus valores $\lambda_1, \lambda_2, ..., \lambda_k$. Con este propósito, cada hormiga selecciona en forma aleatoria un número entero $\tilde{n}$ en base decimal, tal que, $\tilde{n} \in [0, (p^k - 1)]$, donde p es el parámetro de esta estrategia que representa la cantidad de valores posibles que puede tomar cada λ. El número entero $\tilde{n}$ es convertido a su representación en base p, de esta manera se obtiene un número con hasta k dígitos, donde cada dígito corresponde a un valor del parámetro λ.

En este trabajo se utilizó $p = 3$, por lo que cada λ_i puede tomar los valores: 0 (visibilidad no considerada), 1 (peso medio) y 2 (peso alto). Como ejemplo, consideremos un problema con $k = 8$ objetivos y $p = 3$. Una hormiga seleccionaría en forma aleatoria un número entero $\tilde{n}$ entre 0 y $(3^8 - 1)$. Suponiendo que escoja $\tilde{n} = 4589$, se realiza la conversión de $\tilde{n}$ a su representación en base 3 dando por resultado el número 20021222; entonces, cada dígito de éste número es asignado a cada valor de λ, es decir, $\lambda_1 = 2, \lambda_2 = 0, \lambda_3 = 0, \lambda_4 = 2, \lambda_5 = 1, \lambda_6 = 2, \lambda_7 = 2$ *y* $\lambda_8 = 2$. Como se puede notar, la estrategia de Asignación de λ base-p mantiene la escalabilidad con un mayor número de objetivos, aparte de ser una estrategia simple de implementar y comprender.

6 Resultados experimentales

En esta sección se presentan los resultados experimentales obtenidos al comparar el MOACS base-p propuesto en este trabajo con los algoritmos del estado del arte antes citados (MOACS, M3AS y MAS) utilizando la recogida métrica del Hipervolumen presentada en la sección 4. Todas las pruebas fueron ejecutadas en un ordenador Genuine Intel 2.3 GHz con una arquitectura de 64 bits y 15 GB RAM, bajo el sistema operativo Ubuntu 12.04.2 LTS.

A falta de instancias conocidas en la literatura para el TSP *many-objective*, se generaron en forma aleatoria problemas para 2, 4, 8 y 10 objetivos cuidando que la correlación entre dos matrices de costo de un mismo problema no supere el valor de 0.1. Con esto, los objetivos minimizados simultáneamente resultaron contradictorios o al menos poco correlacionados.

Para la ejecución de las pruebas se modificó el algoritmo original MOACS, implementando la Asignación de λ base-p, creando así, una nueva variante de MOACO denominada en adelante MOACS base-p, principal aporte de este trabajo. Dicha variante fue comparada con los mejores algoritmos MOACO del estado del arte según [4], MOACS, M3AS y MAS, además del NSGA2 (*Non Sorting Genetic Algorithm version 2*) considerado el principal algoritmo de referencia del estado del arte en problemas multi-objetivo [7], [11].

Para realizar un análisis de la evolución de los resultados, se definió como criterio de parada, 10.000 generaciones y se obtuvieron los resultados para 2.500, 5.000, 7.500 y 10.000 generaciones transcurridas.

Considerando la parametrización reportada en [4], para los MOACOs se utilizaron los siguientes parámetros: $m = 10$ hormigas, $\alpha = 1$, $\beta = 2$, $\rho = 0.3$, $\tau_0 = 0.1$, $\tau_{max} = 0.9$, $\tau_{min} = 0.1$ y $q_0 = 0.5$. Para el NSGA2, se utilizaron los siguientes parámetros: tamaño de población igual a 10, Probabilidad de Mutación de 0.8 y Probabilidad de cruzamiento de 0.98. Estos valores fueron seleccionados conforme [11].

Los resultados presentados más abajo corresponden al promedio de 4 ejecuciones de cada uno de los algoritmos mencionados. Se utilizaron instancias del TSP con 2, 4, 8 y 10 objetivos y cada instancia fue configurada de manera a considerar tres tamaños de problemas: 50, 75 y 100 ciudades.

En la Figura 1 se puede observar la evolución del Hipervolumen de los Frentes Pareto calculados por cada uno de los algoritmos comparados, cuando resuelven una instancia del TSP con 75 ciudades, considerando solamente 2 objetivos (este problema no puede todavía ser considerado *many-objective*). Se puede notar que el MOACS base-p es levemente superior a los demás algoritmos sin importar el número de generaciones transcurridas, aunque el MAS tiene un desempeño muy próximo.

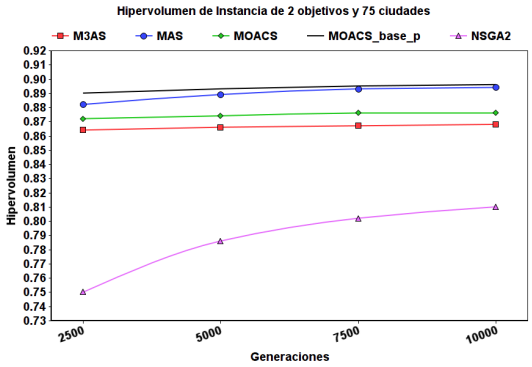


Fig. 1: Hipervolumen obtenido al resolver la Instancia con 2 objetivos y 75 ciudades

La Figura 2 muestra la evolución del Hipervolumen para cada uno de los algoritmos comparados, cuando resuelven una instancia del TSP con 75 ciudades, considerando 4 objetivos simultáneos (ya puede ser considerado un problema *many-objective*). Se puede notar que la variante propuesta es claramente superior a los demás algoritmos sin importar el número de generaciones transcurridas, verificandose la ventaja de utilizar la Asignación de λ base-p propuesta en este trabajo.

Las Figuras 3 y 4 muestran la evolución del Hipervolumen para cada uno de los algoritmos comparados, cuando resuelven una instancia del TSP con 75 ciudades, considerando 8 y 10 objetivos respectivamente, pudiéndose notar una clara superioridad del algoritmo propuesto en este trabajo que mejora su desempeño notoriamente con respecto a los demás algoritmos del estado del arte al aumentar el número de objetivos, confirmando la ventaja que significa utilizar el algoritmo propuesto.

Este mismo comportamiento fue observado al resolver instancias con 50 y 100 ciudades. Para instancias de más de 2 objetivos el MOACS base-p presenta de manera significativa el mejor Hipervolumen de entre todos los algoritmos estudiados en cualquier cantidad de generaciones. Por falta de espacio solo se presentará a continuación los resultados obtenidos utilizando instancias de 50 ciudades, resumidos en las Tablas 1 y 2.

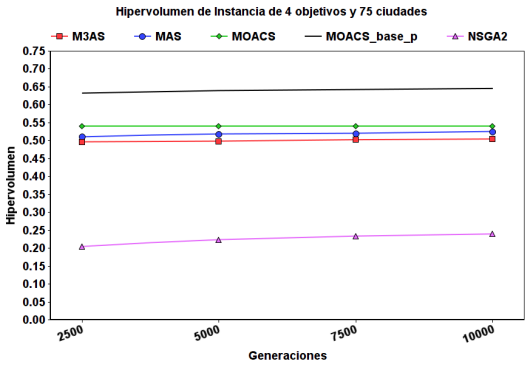


Fig. 2: Hipervolumen obtenido al resolver la Instancia con 4 objetivos y 75 ciudades

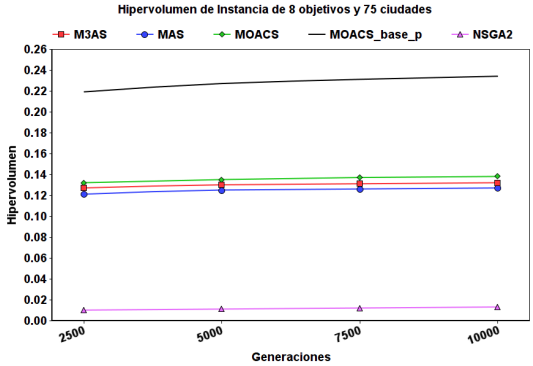


Fig. 3: Hipervolumen obtenido al resolver la Instancia con 8 objetivos y 75 ciudades

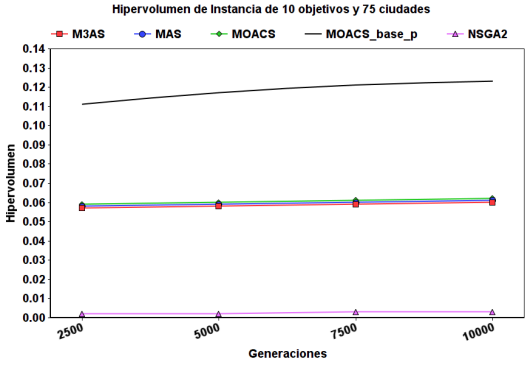


Fig. 4: Hipervolumen obtenido al resolver la Instancia con 10 objetivos y 75 ciudades

Objetivos	2				4			
Generaciones	2.500	5.000	7.500	10.000	2.500	5.000	7.500	10.000
M3AS	0,893	0,895	0,896	0,897	0,643	0,648	0,65	0,651
MAS	0,903	0,911	0,912	0,912	0,633	0,638	0,645	0,656
MOACS	0,898	0,899	0,9	0,901	0,653	0,656	0,658	0,66
MOACS_base_p	0,915	0,916	0,917	0,917	0,743	0,748	0,751	0,753
NSGA2	0,743	0,78	0,796	0,806	0,201	0,215	0,225	0,233

Tabla 1: Hipervolumen obtenido al resolver la Instancias con 50 ciudades para 2 y 4 objetivos

Objetivos	8				10			
Generaciones	2.500	5.000	7.500	10.000	2.500	5.000	7.500	10.000
M3AS	0,289	0,293	0,295	0,296	0,178	0,181	0,182	0,183
MAS	0,284	0,288	0,292	0,294	0,181	0,186	0,188	0,19
MOACS	0,292	0,295	0,297	0,298	0,178	0,182	0,184	0,186
MOACS_base_p	0,411	0,421	0,426	0,431	0,279	0,289	0,294	0,298
NSGA2	0,009	0,011	0,012	0,013	0,002	0,002	0,002	0,003

Tabla 2: Hipervolumen obtenido al resolver la Instancias con 50 ciudades para 8 y 10 objetivos

7 Conclusiones y trabajos futuros

Este trabajo presentó por primera vez en la literatura un estudio del comportamiento de los algoritmos MOACO aplicados a problemas *many-objective*. Se introdujo una nueva y simple variante del MOACS denominada MOACS base-p, que fue empíricamente comparada contra los algoritmos MOACO de mejor rendimiento al resolver el TSP (MOACS, M3AS, MAS) [4] y el algoritmo evolutivo de referencia para resolver problemas multi-objetivo, el NSGA2 [7]. Todos estos algoritmos fueron utilizados para resolver el TSP con 2, 4, 8 y 10 objetivos; almacenando los Frentes Pareto para diferentes cantidades de generaciones, y promediando los resultados de varias ejecuciones. Estos resultados fueron comparados utilizando la reconocida métrica de Hipervolumen, ampliamente utilizada [7].

Descrita la problemática existente en la asignación del parámetro λ en los algoritmos MOACO cuando resuelven problemas *many-objective*, se propuso como solución, una nueva estrategia de asignación de λ denominada Asignación de λ base-p, en la cual se basa el MOACS base-p.

Resultados experimentales demostraron que el MOACS base-p es superior a los mejores MOACO del estado del arte así como al algoritmo de referencia NSGA2, incluso con solo 2 objetivos aunque claramente su ventaja aumenta en la medida que crece el número de objetivos, conforme se verificó experimentalmente hasta 10 objetivos, quedando como trabajo futuro considerar un mayor número de objetivos para lo cual se deberá reconsiderar el uso de la métrica de rendimiento dada la complejidad del cálculo del Hipevolumen que crece ex-

ponencialmente con el número de objetivos. Otras extensiones del presente trabajo serían: implementar el esquema de Asignación de λ base-p en otras implementaciones de MOACO y compararlas con otra metaheurística como el *Particle Swarm Optimization* (PSO), extendiendo el análisis comparativo a otros casos de estudios como son el *Quadratic Assignment Problem* (QAP)[5], el *Vehicle Routing Problem with Time Windows* (VRPTW)[2]; u otros problemas multiobjetivo que puedan ser extendidos para el caso *many-objective*; por último, se propone analizar otros parámetros de los algoritmos MOACO, como por ejemplo la utilización de grupos de tablas de feromonas donde cada tabla de feromonas corresponderá a un grupo de objetivos.

Referencias

1. Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G., Shmoys, D.B.: The travelling salesman problem. Wiley, New York (1985)
2. Barán, B., Schaerer, M.: A multiobjective Ant Colony System for Vehicle Routing Problems with Time Windows. Proc. Twenty first IASTED International Conference on Applied Informatics, pp. 97–102. Insbruck, Austria (2003)
3. Pinto, D., Barán, B.: Solving Multiobjective Multicast Routing Problem with a new Ant Colony Optimization approach. In Proceedings of the 3rd international IFIP/ACM Latin American conference on Networking, pp. 11-19, Cali, Colombia (2005)
4. Paciello, J., Martínez, H., Lezcano, C., Barán, B.: Algoritmos de Optimizacin multiobjetivos basados en colonias de hormigas. XXXII Conferencia Latinoamericana de Informática - CLEI'2006. Santiago de Chile (2006)
5. Martínez, H., Paciello, J., Barán, B.: Equipo distribuido de algoritmos ACO multiobjetivos. VIII Argentine Symposium on Artificial Intelligence - ASAI, en el marco de las 35 Jornadas Argentinas de Informática e Investigación Operativa - 35 JAIIO. Mendoza, Argentina (2006)
6. Paciello, J., Martínez, H., Barán, B.: Aplicación de un equipo de algoritmos ACO al VRPTW bi-objetivo. XIII Conferencia Latino-IberoAmericana de Investigación Operativa - CLAIO'2006. Montevideo, Uruguay (2006)
7. von Lücken, C., Barn, B., Brizuela, C.: A survey on multy-objective evolucionay algorithms for many-objective problems. Computational Optimization and Applications 58, no. 3, pp. 707–756, (2014)
8. Ishibuchi, H., Tsukamoto, N., Nojima, Y.: Evolutionary many-objective optimization: A short review. In IEEE Congress on Evolutionary Computation CEC'2008, pp. 2419-2426 (2008)
9. Zitzler, E.: Evolutionary algorithms for multiobjective optimization: Methods and applications. Vol. 63. Ithaca: Shaker, (1999)
10. While, L., Hingston, P., Barone, L., Huband, S.: A faster algorithm for calculating hypervolume. Evolutionary Computation, IEEE Transactions on, 10 no. 1, pp. 29–38 (2006)
11. Optymalizacji, H. M., Kot, K., Pietraszko, A.: Heurystyczne metody wielokryterialnej optymalizacji kombinatorycznej. Akademia Górniczo-Hutnicza Thesis (2009)
12. Barán, B., Gómez, O.: Reasons of ACO's Success in TSP, ANTS'2004 - Fourth International Workshop on Colony Optimization and Swarm Intelligence. Bruselas, Bélgica. (2004)

Personalização de recursos com suporte à privacidade através de técnicas adaptativas

Paulo Roberto Massa Cereda e João José Neto

Escola Politécnica, Departamento de Engenharia de
Computação e Sistemas Digitais, Universidade de São Paulo
Av. Prof. Luciano Gualberto, s/n, Travessa 3, 158, CEP: 05508-900
São Paulo, SP – Brasil
paulo.cereda@usp.br, jjneto@usp.br

Resumo Provedores de serviços disponibilizam diversos serviços personalizados aos usuários, possibilitando que tais ofertas tornem-se mais aderentes aos interesses de seus visitantes, mesmo sob risco de violação de privacidade. Existem mecanismos presentes na literatura que buscam equacionar privacidade e personalização, mas que, em geral, trabalham de forma estática, isto é, não admitem atualizações contextuais. Este artigo apresenta uma extensão dos trabalhos existentes, utilizando técnicas adaptativas, para melhoria da oferta de serviços personalizados, guarantindo a privacidade do usuário.

1 Introdução

Com o crescimento e difusão da Internet, os provedores de serviços passaram a oferecer diversos serviços personalizados aos usuários. Como conseqüência imediata, as informações relativas aos acessos desses usuários tornaram-se significativas para os websites, de tal forma a serem coletadas sem consentimento explícito. Por outro lado, a privacidade do usuário tornou-se determinante durante tal navegação e, na maioria das vezes, decisiva para a permanência em um determinado website.

A personalização de recursos é um artifício que oferece conforto e proximidade aos usuários, mas depende de algumas informações acerca da utilização do serviço. Assim, ao restringir o envio de suas informações pessoais, não será possível ao usuário usufruir dos serviços personalizados.

Na tentativa de equacionar tal dicotomia, foram propostos mecanismos que propiciem uma navegação com privacidade e, ao mesmo tempo, garantam que o usuário possa receber algum serviço personalizado. De modo particular, o servidor MASKS, apresentado em detalhes na Seção 4, fundamenta-se na idéia de grupos e máscaras, na qual o usuário utiliza-se de máscaras para ocultar sua identidade e rotular-se por um grupo de interesses, permitindo, dessa forma, oferecer serviços personalizados sem conhecimento de sua identidade real. Outros mecanismos operam de maneira análoga.

Tais mecanismos de privacidade e personalização, em geral, trabalham de forma estática, isto é, não admitem atualizações contextuais. Do ponto de vista

de privacidade, a estaticidade dos modelos não tem impacto sobre a identidade real do usuário, mas prejudica significativamente a oferta de serviços personalizados em médio e longo prazo; máscaras e grupos costumam sofrer desgastes conceituais no tempo, tornando-se obsoletos ou demasiadamente genéricos.

Diante de tais impasses, este artigo apresenta uma extensão do formalismo adaptativo introduzido em [4], com a inclusão de componentes, utilizando técnicas adaptativas, no servidor MASKS original para melhoria da oferta de serviços personalizados, garantindo a privacidade do usuário.

2 Privacidade e personalização

Privacidade, em linhas gerais, é o direito de um indivíduo em resguardar suas informações pessoais de outrem. No contexto da Web, a privacidade pode ser caracterizada como a capacidade do usuário em manter controle das informações pessoais [6]. Em [5], o autor afirma que, "para ter privacidade, uma pessoa precisa ter controle sobre as informações existentes sobre si mesma e exercer este controle de forma consistente com seus interesses e valores pessoais". Portanto, durante a navegação do usuário pela Internet, essas informações devem ser divulgadas apenas com o consentimento do mesmo.

Com o advento e consolidação do comércio eletrônico, as questões relacionadas à privacidade tornaram-se ainda mais significativas. É virtualmente impraticável completar uma transação sem revelar quaisquer dados de caráter pessoal; um usuário pode não desejar informar esses dados caso julgue que sua privacidade está sendo invadida ou posta em risco [1].

Personalização pode ser descrita como tornar algo pessoal, individual, dependente das características e dos interesses humanos. Ao personalizar um objeto de acordo com um usuário, cria-se uma relação de afinidade. De acordo com [7], "um produto ou serviço pode atender as necessidades fundamentais de uma pessoa por suas funcionalidades e características primárias. Além disso, um serviço, através da personalização, pode possuir determinadas características que o torna mais parecido com um indivíduo. Essas qualidades secundárias são consideradas tão importantes que em muitos casos a escolha do produto ou serviço é regida somente através delas." Dessa forma, é imperativo que informações relevantes sejam obtidas para que os serviços sejam personalizados [10].

Algumas vantagens de sistemas personalizados incluem melhorias na navegabilidade, reconhecimento de visitas reincidentes e direcionamento de conteúdo [2]. Para a coleta de informações, em geral, utilizam-se análises de dados em formulários e de navegação do usuário; de posse de tais dados, técnicas de mineração de dados são aplicadas para determinar preferências e tendências [3].

3 Dispositivos adaptativos

Um dispositivo guiado por regras $AD = (ND_0, AM)$ é considerado adaptativo sempre que, para todos os passos de operação $k \geq 0$ (k é o valor de um contador embutido T iniciado em zero e que é incrementado de uma unidade toda vez

que uma ação adaptativa não nula é executada), AD segue o comportamento do dispositivo subjacente ND_k até que a execução de uma ação adaptativa não nula inicie o passo de operação $k+1$ através de mudanças no conjunto de regras; em outras palavras, a execução uma ação adaptativa não nula em um passo de operação $k \geq 0$ faz o dispositivo adaptativo AD evoluir do dispositivo subjacente ND_k para ND_{k+1}.

O dispositivo adaptativo AD inicia sua operação na configuração c_0, com o formato inicial definido por $AD_0 = (C_0, AR_0, S, c_0, A, NA, BA, AA)$. No passo k, um estímulo de entrada move AD para uma configuração seguinte e inicia seu passo de operação $k+1$ se, e somente se, uma ação não-adaptativa for executada; dessa forma, estando o dispositivo AD no passo k, com o formato $AD_k = (C_k, AR_k, S, c_k, A, NA, BA, AA)$, a execução de uma ação adaptativa não nula leva a $AD_{k+1} = (C_{k+1}, AR_{k+1}, S, c_{k+1}, A, NA, BA, AA)$, onde $AD = (ND_0, AM)$ é um dispositivo adaptativo com um dispositivo subjacente inicial ND_0 e um mecanismo adaptativo AM, ND_k é o dispositivo subjacente de AD no passo de operação k, NR_k é o conjunto de regras não adaptativas de ND_k, C_k é o conjunto de todas as configurações possíveis para ND no passo de operação k, $c_k \in C_k$ é a configuração inicial no passo k, S é o conjunto de todos os eventos possíveis que são estímulos de entrada para AD, $A \subseteq C$ é o subconjunto as configurações de aceitação (da mesma forma, $F = C - A$ é o subconjunto de configurações de rejeição), BA e AA são conjuntos de ações adaptativas (ambos contendo a ação nula, $\epsilon \in BA \cap AA$), NA, com $\epsilon \in NA$, é o conjunto de todos os possíveis símbolos de saída de AD como efeito da aplicação de regras do dispositivo, AR_k é o conjunto de regras adaptativas definido pela relação $AR_k \subseteq BA \times C \times S \times C \times NA \times AA$, com AR_0 definindo o comportamento inicial de AD, AR é o conjunto de todas as possíveis regras adaptativas para AD, NR é o conjunto de todas as possíveis regras não-adaptativas subjacentes de AD, e AM é o mecanismo adaptativo, $AM \subseteq BA \times NR \times AA$, a ser aplicado em um passo de operação k para cada regra em $NR_k \subseteq NR$. Regras adaptativas $ar \in AR_k$ são da forma $ar = (ba, c_i, s, c_j, z, aa)$ indicando que, em resposta a um estímulo de entrada $s \in S$, ar inicialmente executa a ação adaptativa anterior $ba \in BA$; a execução de ba é cancelada se esta elimina ar do conjunto AR_k; caso contrário, a regra não-adaptativa subjacente $nr = (c_i, s, c_j, z)$, $nr \in NR_k$ é aplicada e, finalmente, a ação adaptativa posterior $aa \in AA$ é executada [9].

Ações adaptativas podem ser definidas em termos de abstrações chamadas funções adaptativas, de modo similar às chamadas de funções em linguagens de programação usuais [9]. A especificação de uma função adaptativa deve incluir os seguintes elementos: *(a)* um nome simbólico, *(b)* parâmetros formais que referenciarão valores passados como argumentos, *(c)* variáveis que conterão valores de uma aplicação de uma ação elementar de inspeção, *(d)* geradores que referenciam valores novos a cada utilização, e *(e)* o corpo da função propriamente dita.

São definidos três tipos de ações adaptativas elementares que realizam testes nas regras ou modificam regras existentes, a saber: *(a) ação adaptativa elementar de inspeção:* a ação não modifica o conjunto de regras, mas permite a inspeção

deste para a verificação de regras que obedeçam um determinado padrão. *(b) ação adaptativa elementar de remoção:* a ação remove regras que correspondem a um determinado padrão do conjunto corrente de regras. *(c) ação adaptativa elementar de inclusão:* a ação insere uma regra que corresponde a um determinado padrão no conjunto corrente de regras.

Tais ações adaptativas elementares podem ser utilizadas no corpo de uma função adaptativa, incluindo padrões de regras que utilizem parâmetros formais, variáveis e geradores disponíveis.

4 Servidor de mascaramento MASKS

O servidor de mascaramento MASKS, proposto por [8], atua como um serviço de proxy estendido, de tal forma a preservar a privacidade de anonimato do usuário e permitir que este usufrua a personalização de serviços. A privacidade do usuário é garantida através do conceito de máscaras – identificações temporárias que um usuário pode assumir.

A atribuição de máscaras utiliza o princípio de associação em grupos, de acordo com cada requisição do usuário. Em outras palavras, quando o usuário faz uma requisição, esta é associada a um determinado grupo, que representa um tópico de interesse; dessa forma, o solicitante da requisição será sempre um grupo, e não mais um indivíduo, divulgando apenas dados sobre interesses comuns [8]. Existem situações, entretanto, em que o usuário terá a necessidade de fornecer suas informações (por exemplo, ao realizar uma compra). Essas informações, obviamente, não serão características comuns de grupo, e o usuário deverá desabilitar o mascaramento explicitamente.

A arquitetura do servidor MASKS possui dois componentes principais: o agente de privacidade e o servidor de máscaras; este último possui dois subcomponentes: seletor e gerenciador de máscaras. O processo de atribuição de máscaras inicia-se quando o usuário faz uma requisição a um determinado website. O agente de privacidade intercepta esta requisição, cifra-a e a envia ao servidor de máscaras. O seletor escolhe o melhor grupo de acordo com a requisição recebida, para que o servidor possa efetivamente enviar a requisição mascarada para o website. A associação de máscaras é feita a cada requisição, pois um usuário pode demonstrar interesses diversos durante uma única sessão, e também para proteger a privacidade, uma vez que o usuário não disponibiliza nenhuma informação além da requisição [8]. Os grupos podem possuir diversas máscaras, uma para cada website que esteja relacionado ao tema de interesse associado ao grupo.

O servidor de máscaras é o componente da arquitetura do MASKS responsável pelo gerenciamento e seleção dos grupos e de atribuição das máscaras, sendo um intermediário entre o agente de privacidade e os websites. Seu componente principal é o seletor, responsável pela seleção de um grupo de interesse que mais adequar-se à requisição do usuário [8].

O seletor utiliza uma árvore de categorias definida pelo *Open Directory Project*, distribuída livremente, classificada e atualizada por voluntários. A justificativa para seu uso desta árvore reside no fato de que os algoritmos de agrupamento

de informações e a própria extração de dados necessitam de conhecimento prévio acerca do usuário, o que não é recomendável neste caso. Como a classificação da árvore é feita por voluntários, ela reflete melhor a semântica dos grupos. Cada nó da árvore de categorias representa um grupo e é formado por um conjunto de páginas relacionadas, um conjunto de termos que o caracterizam (palavras-chaves) e as máscaras referentes a esse grupo. Um nó pode ter filhos, que representam uma especialização semântica de um grupo, e ligações, que são especializações que referenciam outros grupos da árvore.

O seletor utiliza um algoritmo de seleção de grupo que deve retornar o grupo semântico que mais corresponda à requisição do usuário. Resumidamente, suas etapas são: *(a)* tentar determinar o grupo através dos termos de consulta do endereço da requisição, através de uma tabela de termos; *(b)* caso não existam termos de consulta, é feita a busca pela existência de algum grupo semântico associado ao endereço da requisição, na tabela de conteúdo; *(c)* caso o endereço da requisição não exista na tabela, o algoritmo tenta determinar o grupo utilizando termos presentes no endereço, também na tabela de termos; *(d)* em último caso, o algoritmo retorna o grupo raiz. As quatro etapasgarantem a privacidade ao usuário, inclusive no pior caso,no qual não é possível determinar o grupo semântico, retornando, dessa forma, o grupo raiz, mais genérico e abrangente.

O servidor de máscaras permite ainda que os websites utilizem *cookies* para personalizar serviços aos usuários. Entretanto, um *cookie* fornecido por um website pertencerá ao grupo associado à requisição, e não mais ao usuário. Requisições subsequentes serão tratadas utilizando informações comuns ao grupo, não sendo possível individualizá-las.

5 Uma versão adaptativa do servidor MASKS

A escolha do grupo semântico é feita pelo componente seletor, utilizando uma árvore de categorias. A árvore em questão, uma vez carregada pelo servidor de mascaramento, é utilizada apenas para consulta. Quando não é possível determinar qual grupo associar a uma requisição, o grupo raiz é então selecionado para garantir a privacidade e o mínimo de personalização [8].

A árvore de categorias é estática e não permite manipulação de nós. A eficácia do seletor e do próprio servidor de mascaramento dependem, portanto, da estrutura da árvore de categorias; quando as ocorrências da escolha do nó raiz (como grupo semântico) tornarem-se freqüentes e em intervalos relativamente pequenos, é uma razão para acreditar que a árvore não apresenta uma estrutura adequada para o contexto utilizado.

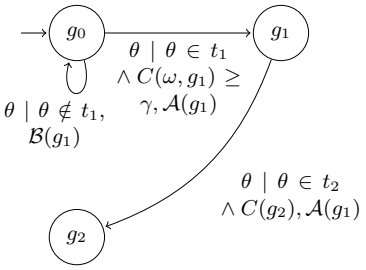
Em [4], os autores apresentam um formalismo para representação do algoritmo de seleção de grupo e a árvore de categorias como um dispositivo adaptativo. A nova versão permite que o modelo semântico modifique-se autonomamente ao longo do tempo, utilizando técnicas adaptativas, de acordo com observação e aprendizado sobre o contexto; dessa forma, a requisição do usuário sempre corresponderá ao melhor grupo disponível, evitando, inclusive, que o nó

raiz seja sempre selecionado para um determinado assunto, caso este não conste na árvore de categorias.

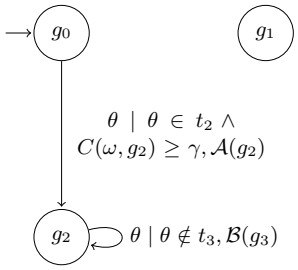
Entretanto, o formalismo introduzido em [4] apresenta dois impasses. O primeiro consiste na ausência de um tratamento adequado quando múltiplos grupos são escolhas válidas para o seletor – o formalismo original admite uma escolha aleatória com probabilidade $\frac{1}{n}$ para n grupos; na prática, o melhor grupo é aquele que contém informações semânticas adicionais mais aderentes à requisição do que os demais. O segundo impasse consiste no crescimento monotônico da árvore de categorias; o formalismo admite apenas a inserção de novos grupos semânticos ao longo do tempo, mas não considera a remoção de grupos que não contribuem potencialmente para a categorização contextual da árvore (ausência de *prunning*). Em um intervalo de tempo suficientemente grande, a árvore tornar-se-á complexa e demasiadamente específica, comprometendo significativamente o desempenho e a personalização de serviços.

Este artigo apresenta uma extensão do formalismo introduzido em [4], com a inclusão de componentes, utilizando técnicas adaptativas, no servidor MASKS original para melhoria da oferta de serviços personalizados, guarantindo a privacidade do usuário. A Figura 1 ilustra uma representação do algoritmo de seleção de grupo e da árvore como um dispositivo adaptativo. Em *(a)*, é apresentado um exemplo de configuração inicial do dispositivo adaptativo, com os seguintes elementos: θ é o termo consultado, G é o conjunto de grupos da árvore de categorias, $g_i \in G, i \in \mathbb{N}$ é um grupo particular do conjunto de grupos, $g_0 \in G$ representa o nó raiz, t_n é o conjunto de termos do grupo n, $T = \bigcup_{i=0}^{n} t_i$ é o conjunto de todos os termos indexados pela árvore de categorias, Ω é o conjunto enumerável de todos os possíveis contextos, $\omega \in \Omega$ representa o contexto corrente, $C\colon \Omega \times G \mapsto \mathbb{R}$ é uma função semântica que atribui um valor real dado o contexto corrente e um grupo de interesse, $\mathcal{A}$ é a função adaptativa a ser executada se o termo pesquisado existe no conjunto de termos do grupo corrente e o contexto é válido, $\theta \in t_n \wedge C(g_n)$, e $\mathcal{B}$ é a função adaptativa a ser executada caso o termo não exista no conjunto de termos do grupo corrente, $\theta \notin t_n$. É importante observar que o novo modelo realiza uma pré-seleção e potencialmente reduz o espaço de possibilidades de escolha de grupo ao inserir um limite real (*threshold*) γ que determina a pertinência semântica de um grupo ao contexto corrente; dessa forma, caso existam múltiplos grupos como escolhas válidas para o seletor, garante-se que o subconjunto R de grupos escolhidos sempre conterá elementos que sejam os mais aderentes possíveis ao contexto corrente, $R = \{g_n \in G \mid \theta \in t_n \wedge C(\omega, g_n) \geq \gamma\}$. Neste caso, a escolha aleatória com probabilidade $\frac{1}{n}$ para n grupos terá um resultado mais significativo do que o formalismo originalmente apresentava. Em *(b)*, é ilustrado um exemplo de configuração final do dispositivo adaptativo. Partindo da raiz g_0, o único estado alcançável é g_2, portanto ele será o grupo escolhido. Caso não existam estados alcançáveis, o grupo escolhido será o nó raiz, tal qual ocorre no algoritmo de seleção de grupo.

O formalismo original disponibiliza uma função adaptativa $\mathcal{C}$ responsável pela criação de um novo grupo semântico com um termo associado [4]. Entretanto, o



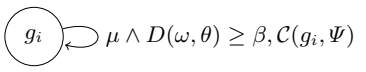
(a) Exemplo de configuração inicial do dispositivo adaptativo.



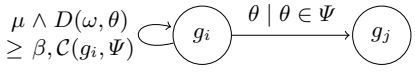
(b) Exemplo de configuração final do dispositivo adaptativo.

Figura 1. Representação do algoritmo de seleção de grupo e da árvore como um dispositivo adaptativo. Em *(a)*, um exemplo de configuração inicial do dispositivo adaptativo; em *(b)*, um exemplo de configuração final.

contexto corrente não era considerado, o que poderia provocar a criação de grupos altamente específicos e potenticialmente identificáveis (causando violação de privacidade [12]). A Figura 2 apresenta uma extensão da função adaptativa $\mathcal{C}$ no novo modelo. Em *(a)*, a função é aplicada no grupo selecionado para mascarar a requisição do usuário; os elementos componentes são: μ é um símbolo especial, indicador de criação de um novo grupo a partir do grupo corrente, $D\colon \Omega \times \Theta \mapsto \mathbb{R}$ é uma função semântica que atribui um valor real dado o contexto e termo de pesquisa correntes, $g_i \in G$ é o grupo corrente, $\Psi \subseteq T$ é o conjunto de termos associados ao novo grupo, e $\mathcal{C}$ é a função adaptativa que criará um novo grupo, uma especialização semântica de seus ancestrais. É importante observar a existência de um limite real β que determina a criação de um novo grupo semântico; dessa forma, limita-se a criação de um grupo de acordo com o contexto e o termo de pesquisa, evitando um crescimento desordenado da árvore de categorias e uma eventual exposição de dados confidenciais. Em *(b)*, é ilustrada a criação de um novo grupo semântico g_j, descendente de g_i. A capacidade de inserção de novos grupos permite que os grupos estejam coesos com o contexto corrente.



(a) Aplicação da função adaptativa $\mathcal{C}$ no grupo selecionado.



(b) Exemplo de criação de um novo grupo semântico.

Figura 2. Função adaptativa $\mathcal{C}$ responsável por uma eventual criação de um novo grupo semântico com um termo associado.

Dois novos componentes são necessários para estender o servidor de mascaramento original: controle e limpeza. Opcionalmente, um terceiro componente – monitoramento – pode compôr a arquitetura do novo servidor.

O componente de controle, como o próprio nome indica, é responsável pelo gerenciamento dos termos obtidos na requisição e pela eventual criação de novos grupos semânticos. Este possui uma tabela de termos pesquisados (Figura 3-a) que é utilizada para a decisão de criação de um novo grupo (função semântica D). A cada requisição, após a escolha do melhor grupo feita pelo seletor, o módulo de controle obtém os termos da requisição e faz uma avaliação dos mesmos, comparando-os com uma tabela de termos já existentes. Cada termo θ_i presente na tabela possui um valor v_{θ_i} referente à sua ocorrência nas requisições; esse valor incrementa conforme os termos surgem nas requisições até que se atinja um limite τ de pontuação $P(\theta_i)$ (Figura 3-b) – em um determinado período de tempo e, então, um novo grupo será criado utilizando o termo em questão. A tabela de termos armazena o número de ocorrências (hits) de um dado termo e a data da sua primeira ocorrência; assim, é possível avaliar se o termo θ_i tem uma procura razoável, fundamentando-se em um intervalo de datas para, então, disparar a função adaptativa $\mathcal{C}$, se, e somente se, $P(\theta_i) \geq \tau$. Se uma requisição possuir um conjunto Y de termos, somente o termo com maior pontuação será considerado, isto é, $y \in Y \mid \max\{P(y') \mid y' \in Y\} = P(y)$.

Termo	Hits	Início
futebol	15	28/01/2015
praia	8	01/02/2015
carnaval	25	13/02/2015

(a)

Cálculo da pontuação do termo θ_i

$$P(\theta_i) = \frac{\text{número de ocorrências de } \theta_i}{\text{data corrente} - \text{início de } \theta_i}$$

(b)

Figura 3. *(a)* Exemplo de uma tabela de termos. *(b)* Fórmula utilizada para cálculo da pontuação de um termo θ_i.

Uma vez criado um novo grupo semântico, o termo relacionado a ele deve receber um sinalizador, informando que este termo já foi avaliado; assim, requisições subseqüentes contendo tal termo não causarão a criação de grupos semânticos repetidos.

O componente de limpeza é responsável pela manutenção da tabela de termos, realizando a remoção dos termos que apresentarem uma pontuação muito baixa, inferior a um limite π, tal que $\pi \ll \tau$, $P(\theta_i) < \pi$, em um determinado período de tempo. A remoção de termos da tabela segue a política *least frequently used*; quanto mais vezes um dado é acessado, maior é a sua relevância e, portanto, terá menores chances de ser removido [11]. Aplicando tal política, a tabela de termos conterá apenas os termos mais relevantes e com chances reais de tornarem-se grupos. O termo θ que obtiver a pontuação mais baixa será removido, $\theta \in \Theta \mid \min\{P(\theta') \mid \theta' \in \Theta\} = P(\theta)$.

O componente opcional de monitoramento realiza ajustes automáticos dos limites definidos pelo modelo (a saber, γ, β, τ e π), de acordo com observações empíricas do sistema em execução. O componente é definido como um dispo-

sitivo adaptativo, ilustrado na Figura 4, que monitora o contexto corrente e executa uma função adaptativa de ajuste $\mathcal{D}$ conforme a ocorrência de padrões comportamentais (por exemplo, muitas requisições atribuídas ao nó raiz indicam a necessidade de criação de novos grupos semânticos). Cada valor pode ser alterado independentemente, de acordo com o contexto em análise.

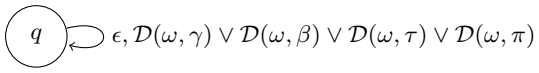


Figura 4. Dispositivo adaptativo para monitoramento do sistema, realizando ajustes nos limites definidos pelo modelo.

Dois experimentos foram realizados para analisar o comportamento de cada servidor em uma situação real de uso. Os testes duraram quatros semanas e envolveram usuários conectados à Internet através dos servidores avaliados. Os resultados são ilustrados na Figura 5.

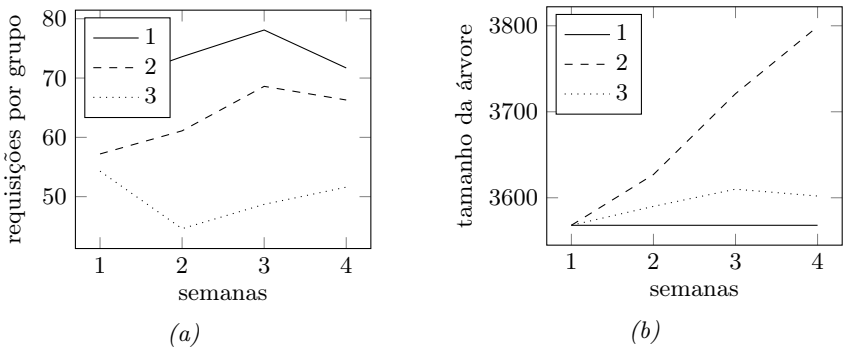


Figura 5. Resultados dos dois experimentos realizados. Legenda: (1) MASKS original, (2) versão proposta por [4], (3) versão proposta neste trabalho.

O primeiro experimento consistiu no cálculo da média de requisições associadas a grupos semânticos, cujo resultado é apresentado na Figura 5-a; é possível observar que o servidor implementando as melhorias propostas neste artigo obteve um número baixo de requisições por grupo em relação aos demais, representando especificações semânticas que podem admitir serviços personalizados mais aderentes. O servidor MASKS original apresentou um número alto de requisições associadas a grupos por não oferecer suporte a alterações contextuais (a maioria das requisições era associada ao nó raiz). O segundo experimento consistiu na mensuração do tamanho da árvore de categorias, conforme ilustra a Figura 5-b; o servidor MASKS original possui um tamanho constante por não admitir atualizações na árvore, enquanto a versão proposta por [4] extrapola a

criação de novos grupos semânticos; o servidor aqui proposto obteve um tamanho de árvore que permita atualizações contextuais sem, entretanto, criar grupos altamente específicos e potencialmente violar a privacidade de seus usuários.

6 Conclusões

Este artigo apresentou uma extensão do formalismo introduzido em [4], com a inclusão de componentes, utilizando técnicas adaptativas, no servidor MASKS original para melhoria da oferta de serviços personalizados, guarantindo a privacidade do usuário. Os resultados obtidos nos experimentos indicaram que as técnicas adaptativas utilizadas mostraram-se adequadas para tratar padrões comportamentais e ciência de contexto sem exigir um custo computacional considerável [3]. Adicionalmente, tal capacidade de percepção contextual contribui para o aumento do grau de satisfação do usuário em relação à utilização do serviço.

As técnicas adaptativas aqui empregadas proporcionam soluções consistentes que atendem às necessidades inerentes de diversos domínios de aplicação, proporcionando soluções computacionais viáveis para problemas complexos, mantendo a simplicidade e a eficiência em tempo e espaço.

Referências

1. Ackwerman, M.S., Cranor, L.: Privacy critics, safeguarding users' personal data. Tech. rep. (1999), web techniques
2. Cereda, P.R.M.: Servidor web adaptativo. In: WTA 2010: Workshop de Tecnologia Adaptativa. Escola Politécnica, Universidade de São Paulo, São Paulo, Brasil (2010)
3. Cereda, P.R.M., José Neto, J.: Adaptive data mining: Preliminary studies. IEEE Latin America Transactions 12(7), 1258–1270 (October 2014)
4. Cereda, P.R.M., Zorzo, S.D.: Formalismo com autômato adaptativo em mecanismo de privacidade e personalização. In: Memoria de la XXXIII Conferencia Latinoamericana en Informatica, CLEI 2007 (2007)
5. Fernandes, C.H.: A privacidade na sociedade da informação. Tech. rep. (2003)
6. Friedman, B., Khan, P.H., Howe, D.C.: Trust online. Communications of the ACM (2000)
7. Grande, R.E.: Sistema de integração de técnicas de proteção de privacidade que permitem personalização. Master's thesis, Universidade Federal de São Carlos (2006)
8. Ishitani, L., Almeida, V., Meira, W.: Masks: bringing anonymity and personalization together. IEEE Security and Privacy Magazine 3(1), 18–23 (2003)
9. José Neto, J.: Adaptive rule-driven devices: general formulation and case study. In: International Conference on Implementation and Application of Automata (2001)
10. Koch, M.: User representation in e-commerce and collaboration applications. Tech. rep., Department of Informatics, Technische Universitaet Muenchen (2003)
11. Robinson, J.T., Devarakonda, M.V.: Data cache management using frequency-based replacement. In: Proceedings of the 1990 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems. pp. 134–142 (1990)
12. Warren, S.D., Brandeis, L.D.: The right to privacy. Harvard Law Review 4(5) (1890)

Optimización basada en Colonias de Hormigas para el Problema del Vendedor Viajante con muchos objetivos contradictorios

Francisco Riveros[1], Néstor Benítez[1], Julio Paciello[1] y Benjamín Barán[1,2]

[1] Universidad Nacional de Asunción
{friverosn,nestorbdietrich,juliopaciello}@gmail.com,
bbaran@pol.una.py

[2] Universidad Nacional del Este

Abstract. Considerando el conocido problema de disminución de rendimiento que presentan los algoritmos evolutivos cuando resuelven problemas denominados *many-objective*, este trabajo propone la utilización de una variante de la Optimización basada en Colonias de Hormigas que denominamos λ base-p. La nueva propuesta fue sometida a distintas pruebas experimentales sobre instancias del Problema del Vendedor Viajante con muchos objetivos utilizando la métrica del Hipervolumen. La Asignación de λ base-p fue comparada con algoritmos MOACO del estado del arte y con el algoritmo evolutivo NSGA2, demostrando que logra calcular un mejor Hipervolumen cuando se resuelven problemas de muchos objetivos contradictorios (*many-objective*).

Keywords: MOACO, TSP, *many-objective*, Hipervolumen, NSGA2

1 Introducción

Los algoritmos *Multi-Objective Ant Colony Optimization* (MOACO) fueron tradicionalmente utilizados para resolver problemas con múltiples objetivos contradictorios, denominados *Multi-Objective Optimization Problems* (MOP), considerados como uno de los métodos con mejor desempeño para resolver el Problema del Vendedor Viajante (TSP-*Travelling Salesman Problems*) [1] . Trabajos anteriores como [2] , [3] y [4] utilizaron los algoritmos MOACO para resolver los denominados MOP del mundo real, y lo hicieron de manera efectiva para 2 o 3 objetivos. Sin embargo, esta efectividad puede verse afectada cuando el número de objetivos aumenta, como pasa con la mayoría de los algoritmos evolutivos [7]. La problemática tratada en este trabajo es el desempeño de los algoritmos MOACO cuando son utilizados para resolver problemas *many-objective* (problemas con 4 o más objetivos contradictorios). En este contexto, se implementó una nueva estrategia de MOACO para resolver problemas *many-objective* ya que las implementaciones actuales no contemplan resolver estos problemas con muchos objetivos. En las siguientes secciones se describe el TSP *many-objective*,

los algoritmos MOACO considerados en el presente trabajo, la estrategia propuesta por este trabajo para considerar problemas *many-objective* y finalmente, se presentan los resultados experimentales, las conclusiones y trabajos futuros.

2 TSP *many-objective*

El TSP puede ser representado por un grafo ponderado completamente conexo $G = (N, A)$, donde N representa un conjunto de $c = |N|$ nodos y A un conjunto de aristas que conectan los nodos de N. En un caso mono-objetivo, cada arista tiene asociado una función de costo d_{ij}, que representa la distancia entre los nodos i y j. El TSP consiste en encontrar el ciclo Hamiltoniano que minimice la distancia recorrida desde un nodo inicial, pasando por cada nodo exactamente una vez y volviendo finalmente al nodo inicial [1]. Para formular el TSP se puede considerar una variable dicotómica $x_{i,j}$ para todo $(i, j) \in A$, de forma que tome el valor 1 si el arco forma parte del ciclo Hamiltoniano y 0 en otro caso, entonces, el TSP consiste en:

$$minimizar \sum_{(i,j)\in A} d_{i,j}x_{i,j} \tag{1}$$

Para el caso del TSP *many-objective*, son consideradas k funciones de costos, teniendo cada arista (i, j) asociado un conjunto de k distancias $d^1_{ij}, d^2_{ij}..., d^k_{ij}$. El problema consiste en minimizar las k funciones de costos al mismo tiempo [8], es decir:

$$minimizar \left[\sum_{(i,j)\in A} d^1_{i,j}x_{i,j}; \sum_{(i,j)\in A} d^2_{i,j}x_{i,j}; ...; \sum_{(i,j)\in A} d^k_{i,j}x_{i,j} \right] \tag{2}$$

3 Algoritmos MOACO

Los algoritmos *Ant Colony Optimization* (ACO) [12] se inspiran en el comportamiento de colonias de hormigas reales para resolver problemas de optimización combinatoria. Se basan en una colonia de hormigas artificiales, es decir, colonia de agentes computacionales simples que trabajan de manera cooperativa y se comunican mediante rastros de feromonas artificiales. Para aplicar un ACO al problema aquí tratado, se utiliza un conjunto de agentes llamados hormigas. Las hormigas construyen las soluciones del problema recorriendo el grafo $G = (N, A)$, partiendo desde un nodo inicial. Para cada cambio de nodo, una hormiga considera los parámetros de visibilidad $(\eta_{i,j})$ y feromonas $(\tau_{i,j})$ de cada arco utilizando una política probabilística que está dada por la siguiente fórmula:

$$p_{i,j} = \begin{cases} \frac{\tau^{\alpha}_{i,j}\eta^{\beta}_{i,j}}{\sum_{x\in J_i}\tau^{\alpha}_{i,x}\eta^{\beta}_{i,x}} & \text{si } j \in J_i \\ 0 & \text{en otro caso} \end{cases} \tag{3}$$

donde J_i representa a todos los posibles nodos que pueden ser visitados desde el nodo i (esto es, los nodos que todavía no han sido visitado por una hormiga);

mientras que α y β son parámetros definidos a priori que reflejan la importancia relativa de las feromonas y la visibilidad respectivamente. La actualización de las feromonas está dado por:

$$\tau_{i,j} = (1-\rho)\tau_{i,j} + \rho\Delta\tau \tag{4}$$

donde ρ es el coeficiente que se utiliza para representar la evaporación y $\Delta\tau$ es la cantidad de feromonas que se va sumando a los arcos que forman parte de una solución escogida.
Un MOACO es una extensión de la metaheurística ACO para resolver problemas multi-objetivos. En general, las ecuaciones se modifican, conforme se muestra a continuación [4]:

$$p_{i,j} = \begin{cases} \frac{\tau_{i,j}^{\alpha}(\eta_{i,j}^{1}\eta_{i,j}^{2}\ldots\eta_{i,j}^{k})^{\beta}}{\sum_{x\in J_i}\tau_{i,x}^{\alpha}(\eta_{i,x}^{1}\eta_{i,x}^{2}\ldots\eta_{i,x}^{k})^{\beta}} & \text{si } j \in J_i \\ 0 & \text{en otro caso} \end{cases} \tag{5}$$

$$\Delta\tau = \frac{1}{\sum_{l=1}^{k}\sum_{(i,j)\in A} d_{i,j}^{l} x_{i,j}} \tag{6}$$

donde k es la cantidad de objetivos. En el Algoritmo 1 se puede observar el pseudo-código de un MOACO genérico:

Algoritmo 1. Pseudo-código de un MOACO genérico .

```
procedure MOACO()
   inicializarParametros()
   while not condicionParada()
       generacion = generacion + 1
       for ant = 1 to m //m es la cantidad de hormigas
             construirSolucion()
             evaluarSolucion()
             actualizarCPFeromonas() // definido en (4)
       end for
    end while
return ConjuntoPareto
end procedure

procedure construirSolucion()
   sol = {}
   while existenEstadosNoVisitados()
        siguiente = seleccionarSiguienteEstado() // definido en (5)
        sol = sol U siguiente
        marcarVisitado(siguiente)
        if (actualizacionPasoAPaso)
           actualizarFeromonasPasoAPaso() // por ejemplo, usando (4)
        end if
   end while
end procedure
```

Actualmente existen diversas implementaciones de MOACOs que son objeto de constantes investigaciones [5], [6]. Cada implementación incluye diferentes enfoques de actualización de feromonas, cantidad de tablas de feromonas, cantidad de hormigas, etc. Para este trabajo se seleccionaron tres algoritmos MOACO del estado del arte: el *Multi-objective Ant System* (MAS) , el *Multi-objective Ant Colony System* (MOACS) y el *Multi-objective Max-Min Ant System* (M3AS), en base a su buen comportamiento resolviendo el problema TSP experimentalmente demostrado en [4]. Cabe recordar que en un MOACO, cada hormiga construye una solución mediante la transición de estados dado por ejemplo en (5). En el caso específico del MOACS, se utiliza la selección pseudo-aleatoria dada por (7) que considera el uso de la probabilidad de transición ilustrada en la ecuación (8), que también es utilizada como regla de transición por las variantes M3AS y MAS.

$$j = \begin{cases} max_{j \in J_i}\{\tau_{i,j}^{\alpha}(\eta_{i,j}^{1\lambda_1}\eta_{i,j}^{2\lambda_2}...\eta_{i,j}^{k\lambda_k})^{\beta}\} & \text{si } q < q_0 \\ \hat{\text{p}} & \text{en otro caso} \end{cases} \tag{7}$$

donde la variable $\hat{\text{p}}$ se calcula mediante la probabilidad $p_{i,j}$ que se define como:

$$p_{i,j} = \begin{cases} \frac{\tau_{i,j}^{\alpha}(\eta_{i,j}^{1\lambda_1}\eta_{i,j}^{2\lambda_2}...\eta_{i,j}^{k\lambda_k})^{\beta}}{\sum_{x \in J_i}\tau_{i,x}^{\alpha}(\eta_{i,x}^{1\lambda_1}\eta_{i,x}^{2\lambda_2}...\eta_{i,x}^{k\lambda_k})^{\beta}} & \text{si } j \in J_i \\ 0 & \text{en otro caso} \end{cases} \tag{8}$$

y las variables λ_1 a λ_k son parámetros utilizados para ponderar la importancia relativa de cada objetivo. Típicamente, cada hormiga puede utilizar un conjunto diferente de parámetros λ_1 a λ_k de forma a guiar la búsqueda de cada hormiga a regiones diferentes del espacio objetivo, lo que explica el reconocido éxito del algoritmo MOACS [4]. Todos los MOACOs implementados fueron objetos de varias pruebas experimentales utilizando las instancias del TSP *many-objective* generadas considerando 2, 4, 8 y 10 objetivos. Las pruebas experimentales fueron ejecutadas en un contexto de minimización simultánea de todas las funciones objetivos (costos). Luego de un análisis previo, realizado a las implementaciones de MOACOs aplicados al TSP *many-objective*, se pudo observar: (1) un deterioro de sus rendimientos al aumentar el número de funciones objetivos; y (2) el MOACS resultaba levemente superior a las demás alternativas. En consecuencia, se decidió modificar los parámetros de ejecución del MOACS de forma a hacerlo aún más competitivo al resolver problemas *many-objective*.

4 Métrica de rendimiento

Para la comparación de los resultados experimentales, fue utilizada la métrica del Hipervolumen [9], reconociendo su amplia adopción entre los investigadores del área de optimización multi-objetivo [7]. El Hipervolumen considera el tamaño de la región de dominancia en el espacio objetivo combinando las métricas de distancia, distribución y extensión en un solo valor [9]. Dado un conjunto de soluciones $Z = \{z_1, z_2, ..., z_R\}$ con R soluciones, y un punto de referencia y_{ref},

el Hipervolumen se define como [9]:

$$H(Z) = H(Z, y_{ref}) = \Lambda(\bigcup_{i=1}^{R} H(z_i)) \tag{9}$$

El Hipervolumen de un Frente Pareto X se define como la unión de las porciones que están limitadas por el punto de referencia y_{ref} denominado punto anti-óptimo o peor posible y los puntos del Frente Pareto evaluado. Considerando un problema de dos funciones objetivos con dos Frentes Pareto conocidos $X = \{a, b, c, d, e\}$, $X' = \{a', b', c', d', e'\}$ y el punto de referencia y_{ref}, si el Hipervolumen de X es mayor al de X', entonces, el Frente Pareto X no es peor que el Frente Pareto X' [10].

5 Implementación propuesta

En las reglas de transición de las fórmulas (7) y (8) podemos notar un parámetro de gran importancia que se utiliza para asignar una cierta prioridad a cada uno de los objetivos tenidos en cuenta, este parámetro es denominado λ. Típicamente, el parámetro λ es un valor complementario; de modo que si se aumenta para un objetivo, se debe disminuir para del otro y viceversa. Con esta asignación de λ se busca ponderar más la información de un objetivo que la de otro para así especializarse en una zona del espacio de búsqueda. Por ejemplo, para problemas bi-objetivos [2], se puede escoger los valores de λ_1 y λ_2 utilizando la siguiente relación:

$$\lambda_2 = m - \lambda_1 + 1 \tag{10}$$

donde m representa el número de hormigas y λ_1 toma valores enteros entre 1 y m. Cabe mencionar que para un problema de k objetivos $f_1, f_2, ..., f_k$, se debe asignar valores a $\lambda_1, \lambda_2, ..., \lambda_k$ por cada hormiga, donde λ_1 pondera a f_1, λ_2 pondera a f_2 y así sucesivamente. En general, cada λ toma uno de m valores posibles por lo que se necesitaran k^m hormigas para tener todas las posibles permutaciones, lo cual resulta inviable al crecer el número de objetivos k. Para abordar esta problemática, este trabajo propone una nueva estrategia denominada Asignación de λ base-p.

5.1 Asignación de λ base-p

Esta nueva estrategia aplica una restricción a los valores posibles que puede tomar cada λ que queda así restringido por un parámetro $p \in \mathbb{N}$. Dados los objetivos $f_1, f_2, ..., f_k$, donde k es la cantidad de objetivos, cada hormiga debe escoger aleatoriamente sus valores $\lambda_1, \lambda_2, ..., \lambda_k$. Con este propósito, cada hormiga selecciona en forma aleatoria un número entero $\tilde{n}$ en base decimal, tal que, $\tilde{n} \in [0, (p^k - 1)]$, donde p es el parámetro de esta estrategia que representa la cantidad de valores posibles que puede tomar cada λ. El número entero $\tilde{n}$ es convertido a su representación en base p, de esta manera se obtiene un número con hasta k dígitos, donde cada dígito corresponde a un valor del parámetro λ.

En este trabajo se utilizó $p = 3$, por lo que cada λ_i puede tomar los valores: 0 (visibilidad no considerada), 1 (peso medio) y 2 (peso alto). Como ejemplo, consideremos un problema con $k = 8$ objetivos y $p = 3$. Una hormiga seleccionaría en forma aleatoria un número entero $\tilde{n}$ entre 0 y $(3^8 - 1)$. Suponiendo que escoja $\tilde{n} = 4589$, se realiza la conversión de $\tilde{n}$ a su representación en base 3 dando por resultado el número 20021222; entonces, cada dígito de éste número es asignado a cada valor de λ, es decir, $\lambda_1 = 2, \lambda_2 = 0, \lambda_3 = 0, \lambda_4 = 2, \lambda_5 = 1, \lambda_6 = 2, \lambda_7 = 2$ *y* $\lambda_8 = 2$. Como se puede notar, la estrategia de Asignación de λ base-p mantiene la escalabilidad con un mayor número de objetivos, aparte de ser una estrategia simple de implementar y comprender.

6 Resultados experimentales

En esta sección se presentan los resultados experimentales obtenidos al comparar el MOACS base-p propuesto en este trabajo con los algoritmos del estado del arte antes citados (MOACS, M3AS y MAS) utilizando la recogida métrica del Hipervolumen presentada en la sección 4. Todas las pruebas fueron ejecutadas en un ordenador Genuine Intel 2.3 GHz con una arquitectura de 64 bits y 15 GB RAM, bajo el sistema operativo Ubuntu 12.04.2 LTS.

A falta de instancias conocidas en la literatura para el TSP *many-objective*, se generaron en forma aleatoria problemas para 2, 4, 8 y 10 objetivos cuidando que la correlación entre dos matrices de costo de un mismo problema no supere el valor de 0.1. Con esto, los objetivos minimizados simultáneamente resultaron contradictorios o al menos poco correlacionados.

Para la ejecución de las pruebas se modificó el algoritmo original MOACS, implementando la Asignación de λ base-p, creando así, una nueva variante de MOACO denominada en adelante MOACS base-p, principal aporte de este trabajo. Dicha variante fue comparada con los mejores algoritmos MOACO del estado del arte según [4], MOACS, M3AS y MAS, además del NSGA2 (*Non Sorting Genetic Algorithm version 2*) considerado el principal algoritmo de referencia del estado del arte en problemas multi-objetivos [7], [11].

Para realizar un análisis de la evolución de los resultados, se definió como criterio de parada, 10.000 generaciones y se obtuvieron los resultados para 2.500, 5.000, 7.500 y 10.000 generaciones transcurridas.

Considerando la parametrización reportada en [4], para los MOACOs se utilizaron los siguientes parámetros: $m = 10$ hormigas, $\alpha = 1$, $\beta = 2$, $\rho = 0.3$, $\tau_0 = 0.1$, $\tau_{max} = 0.9$, $\tau_{min} = 0.1$ y $q_0 = 0.5$. Para el NSGA2, se utilizaron los siguientes parámetros: tamaño de población igual a 10, Probabilidad de Mutación de 0.8 y Probabilidad de cruzamiento de 0.98. Estos valores fueron seleccionados conforme [11].

Los resultados presentados más abajo corresponden al promedio de 4 ejecuciones de cada uno de los algoritmos mencionados. Se utilizaron instancias del TSP con 2, 4, 8 y 10 objetivos y cada instancia fue configurada de manera a considerar tres tamaños de problemas: 50, 75 y 100 ciudades.

En la Figura 1 se puede observar la evolución del Hipervolumen de los Frentes Pareto calculados por cada uno de los algoritmos comparados, cuando resuelven una instancia del TSP con 75 ciudades, considerando solamente 2 objetivos (este problema no puede todavía ser considerado *many-objective*). Se puede notar que el MOACS base-p es levemente superior a los demás algoritmos sin importar el número de generaciones transcurridas, aunque el MAS tiene un desempeño muy próximo.

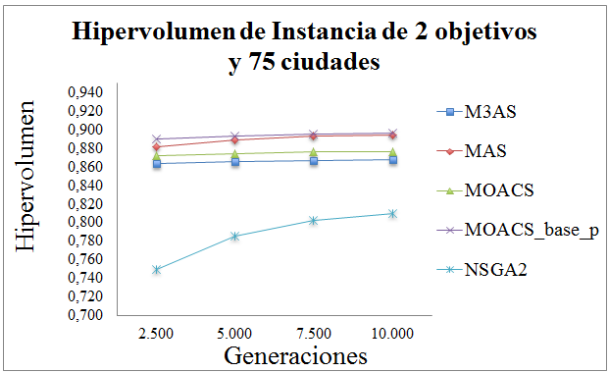


Fig. 1: Hipervolumen obtenido al resolver la Instancia con 2 objetivos y 75 ciudades

La Figura 2 muestra la evolución del Hipervolumen para cada uno de los algoritmos comparados, cuando resuelven una instancia del TSP con 75 ciudades, considerando 4 objetivos simultáneos (ya puede ser considerado un problema *many-objective*). Se puede notar que la variante propuesta es claramente superior a los demás algoritmos sin importar el número de generaciones transcurridas, verificandose la ventaja de utilizar la Asignación de λ base-p propuesta en este trabajo.

Las Figuras 3 y 4 muestran la evolución del Hipervolumen para cada uno de los algoritmos comparados, cuando resuelven una instancia del TSP con 75 ciudades, considerando 8 y 10 objetivos respectivamente, pudiéndose notar una clara superioridad del algoritmo propuesto en este trabajo que mejora su desempeño notoriamente con respecto a los demás algoritmos del estado del arte al aumentar el número de objetivos, confirmando la ventaja que significa utilizar el algoritmo propuesto.

Este mismo comportamiento fue observado al resolver instancias con 50 y 100 ciudades. Para instancias de más de 2 objetivos el MOACS base-p presenta de manera significativa el mejor Hipervolumen de entre todos los algoritmos estudiados en cualquier cantidad de generaciones. Por falta de espacio solo se presentará a continuación los resultados obtenidos utilizando instancias de 50 ciudades, resumidos en las Tablas 1 y 2.

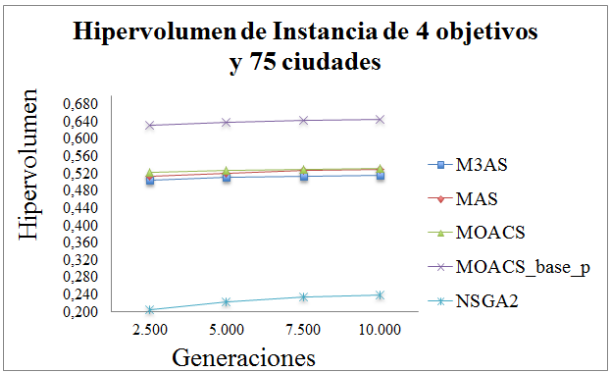


Fig. 2: Hipervolumen obtenido al resolver la Instancia con 4 objetivos y 75 ciudades

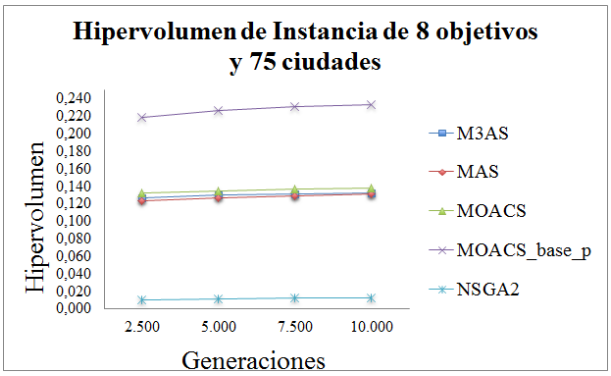


Fig. 3: Hipervolumen obtenido al resolver la Instancia con 8 objetivos y 75 ciudades

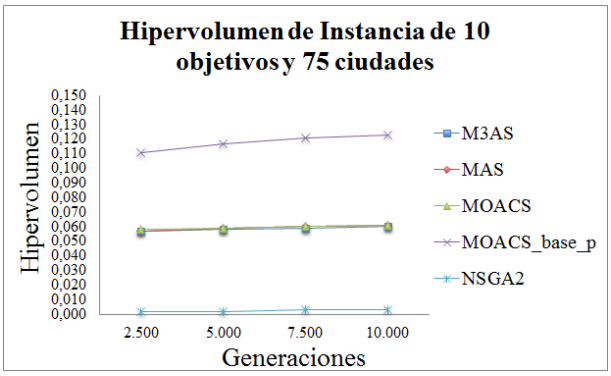


Fig. 4: Hipervolumen obtenido al resolver la Instancia con 10 objetivos y 75 ciudades

Objetivos	2				4			
Generaciones	2.500	5.000	7.500	10.000	2.500	5.000	7.500	10.000
M3AS	0,893	0,895	0,896	0,897	0,643	0,648	0,65	0,651
MAS	0,903	0,911	0,912	0,912	0,633	0,638	0,645	0,656
MOACS	0,898	0,899	0,9	0,901	0,653	0,656	0,658	0,66
MOACS_base_p	0,915	0,916	0,917	0,917	0,743	0,748	0,751	0,753
NSGA2	0,743	0,78	0,796	0,806	0,201	0,215	0,225	0,233

Tabla 1: Hipervolumen obtenido al resolver la Instancias con 50 ciudades para 2 y 4 objetivos

Objetivos	8				10			
Generaciones	2.500	5.000	7.500	10.000	2.500	5.000	7.500	10.000
M3AS	0,289	0,293	0,295	0,296	0,178	0,181	0,182	0,183
MAS	0,284	0,288	0,292	0,294	0,181	0,186	0,188	0,19
MOACS	0,292	0,295	0,297	0,298	0,178	0,182	0,184	0,186
MOACS_base_p	0,411	0,421	0,426	0,431	0,279	0,289	0,294	0,298
NSGA2	0,009	0,011	0,012	0,013	0,002	0,002	0,002	0,003

Tabla 2: Hipervolumen obtenido al resolver la Instancias con 50 ciudades para 8 y 10 objetivos

7 Conclusiones y trabajos futuros

Este trabajo presentó por primera vez en la literatura un estudio del comportamiento de los algoritmos MOACO aplicados a problemas *many-objective*. Se introdujo una nueva y simple variante del MOACS denominada MOACS base-p, que fue empíricamente comparada contra los algoritmos MOACO de mejor rendimiento al resolver el TSP (MOACS, M3AS, MAS) [4] y el algoritmo evolutivo de referencia para resolver problemas multi-objetivos, el NSGA2 [7]. Todos estos algoritmos fueron utilizados para resolver el TSP con 2, 4, 8 y 10 objetivos; almacenando los Frentes Pareto para diferentes cantidades de generaciones, y promediando los resultados de varias ejecuciones. Estos resultados fueron comparados utilizando la reconocida métrica de Hipervolumen, ampliamente utilizada [7].

Descrita la problemática existente en la asignación del parámetro λ en los algoritmos MOACO cuando resuelven problemas *many-objective*, se propuso como solución, una nueva estrategia de asignación de λ denominada Asignación de λ base-p, en la cual se basa el MOACS base-p.

Resultados experimentales demostraron que el MOACS base-p es superior a los mejores MOACO del estado del arte así como al algoritmo de referencia NSGA2, incluso con solo 2 objetivos aunque claramente su ventaja aumenta en la medida que crece el número de objetivos, conforme se verificó experimentalmente hasta 10 objetivos, quedando como trabajo futuro considerar un mayor número de objetivos para lo cual se deberá reconsiderar el uso de la métrica de rendimiento dada la complejidad del cálculo del Hipevolumen que crece ex-

ponencialmente con el número de objetivos. Otras extensiones del presente trabajo serían: implementar el esquema de Asignación de λ base-p en otras implementaciones de MOACO y compararlas con otra metaheurística como el *Particle Swarm Optimization* (PSO), extendiendo el análisis comparativo a otros casos de estudios como son el *Quadratic Assignment Problem* (QAP)[5], el *Vehicle Routing Problem with Time Windows* (VRPTW)[2]; u otros problemas multiobjetivos que puedan ser extendidos para el caso *many-objective*; por último, se propone analizar otros parámetros de los algoritmos MOACO, como por ejemplo la utilización de grupos de tablas de feromonas donde cada tabla de feromonas corresponderá a un grupo de objetivos.

Referencias

1. Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G., Shmoys, D.B.: The travelling salesman problem. Wiley, New York (1985)
2. Barán, B., Schaerer, M.: A multiobjective Ant Colony System for Vehicle Routing Problems with Time Windows. Proc. Twenty first IASTED International Conference on Applied Informatics, pp. 97–102. Insbruck, Austria (2003)
3. Pinto, D., Barán, B.: Solving Multiobjective Multicast Routing Problem with a new Ant Colony Optimization approach. In Proceedings of the 3rd international IFIP/ACM Latin American conference on Networking, pp. 11-19, Cali, Colombia (2005)
4. Paciello, J., Martínez, H., Lezcano, C., Barán, B.: Algoritmos de Optimizacin multiobjetivos basados en colonias de hormigas. XXXII Conferencia Latinoamericana de Informática - CLEI'2006. Santiago de Chile (2006)
5. Martínez, H., Paciello, J., Barán, B.: Equipo distribuido de algoritmos ACO multiobjetivos. VIII Argentine Symposium on Artificial Intelligence - ASAI, en el marco de las 35 Jornadas Argentinas de Informática e Investigación Operativa - 35 JAIIO. Mendoza, Argentina (2006)
6. Paciello, J., Martínez, H., Barán, B.: Aplicación de un equipo de algoritmos ACO al VRPTW bi-objetivo. XIII Conferencia Latino-IberoAmericana de Investigación Operativa - CLAIO'2006. Montevideo, Uruguay (2006)
7. von Lücken, C., Barn, B., Brizuela, C.: A survey on multy-objective evolucionay algorithms for many-objective problems. Computational Optimization and Applications 58, no. 3, pp. 707–756, (2014)
8. Ishibuchi, H., Tsukamoto, N., Nojima, Y.: Evolutionary many-objective optimization: A short review. In IEEE Congress on Evolutionary Computation CEC'2008, pp. 2419-2426 (2008)
9. Zitzler, E.: Evolutionary algorithms for multiobjective optimization: Methods and applications. Vol. 63. Ithaca: Shaker, (1999)
10. While, L., Hingston, P., Barone, L., Huband, S.: A faster algorithm for calculating hypervolume. Evolutionary Computation, IEEE Transactions on, 10 no. 1, pp. 29–38 (2006)
11. Optymalizacji, H. M., Kot, K., Pietraszko, A.: Heurystyczne metody wielokryterialnej optymalizacji kombinatorycznej. Akademia Górniczo-Hutnicza Thesis (2009)
12. Barán, B., Gómez, O.: Reasons of ACO's Success in TSP, ANTS'2004 - Fourth International Workshop on Colony Optimization and Swarm Intelligence. Bruselas, Bélgica. (2004)

A Preliminary Framework for Reasoning with Inconsistent Possibilistic Description Logics Ontologies with Disjunctive Assertions

Sergio Alejandro Gómez

Artificial Intelligence Research and Development Laboratory (LIDIA)
Department of Computer Science and Engineering
Universidad Nacional del Sur
Av. Alem 1253, (8000) Bahía Blanca, ARGENTINA
Email: sag@cs.uns.edu.ar

Abstract. We present a preliminary framework for reasoning with *possibilistic description logics ontologies with disjunctive assertions* (PDLDA ontologies for short). PDLDA ontologies are composed of a terminology as well as an assertional box that allows to declare three kinds of assertional statements: an individual is a member of one concept, two individuals are related through a role, *an individual is a member of the union of two or more concepts* or *two individuals are related through the union of two or more roles*. Each axiom in the ontologies has a certainty degree as is usual in possibilistic logics. For reasoning with PDLDA ontologies, we interpret them in terms of a adaptation of Bodanza's Suppositional Argumentation System. Our framework allows to reason with modus ponens and constructive dilemmas. We use it for determining the membership of individuals to concepts when there is doubt to exactly which one of the concepts in the union the individual belongs. We think that our approach will be of use for implementers of reasoning systems in the Semantic Web where uncertainty of membership of individuals to concepts or roles is present.

Keywords: Supppositional argumentation, ontology reasoning, inconsistency handling, Description Logics.

1 Introduction

Reasoning with Description Logics ontologies [1] is an important topic for the implementation of the Semantic Web [2], where data resources are described in terms of ontologies expressed in the Web Ontology Language (OWL-DL). OWL-DL underlying semantics are expressed in Description Logics. Traditionally, an ontology is a set of axioms that defines intensionally (by means of a Tbox or terminological box) a set of concepts and roles and extensionally a set individuals belonging to some of the concepts and/or relating to each other through roles (by means of an Abox or assertional box).

Incoherence and inconsistency are two anomalies that may affect an ontology. An incoherent ontology has definitions that render some concepts empty.

Inconsistent ontologies imply that an individual is a member of both a concept and its complement. Because DL ontologies are based on decidable fragments of first-order logic, inconsistency makes anything derivable from such an ontology.

Inconsistent ontologies pose a problem for this because they have to be debugged by the knowledge engineer prior to deployment of applications. Many times it is not possible to do this because the domain being modeled is intrinsically contradictory or the programmer does not have the authority to edit the ontology's contents. Many approaches have been proposed for dealing with this situation is solved in two main ways. The first kind of approaches consists of eliminating part of the ontology to make it consistent again (e.g. approaches based on Belief Revision theory [3, 4] or on paraconsistent logics [5]). The second kind consists of accepting inconsistency and using some kind of non-standard reasoning mechanism to get some meaningful answers from an inconsistent knowledge base (e.g. argumentation [6–9]). We will rely on the latter approach in this paper.

Defeasible argumentation is an approach to non-monotonic reasoning that can be used when several diverging opinions may exist (each opinion is supported by an argument instead of a proof), so differences can be resolved by considering all pros and cons of a given claim through a dialectical analysis [10–12]. Given a claim, this process usually takes into account defeaters (other claims that are against the original claim and seem to have greater importance) and defeaters of those defeaters (thus maybe reinstating the original claim) as part of a recursive process that ends when no defeater can be found. Defeasible Logic Programming (DeLP) [13] is a rule-based implementation of defeasible argumentation based on the Prolog programming language. Possibilistic DeLP (or just PDeLP) [14] is another argumentative approach to reasoning that derives from DeLP. Unlike DeLP that uses specificity for comparing arguments during the dialectical analysis, PDeLP uses possibilistic degrees associated to each rule to qualify arguments indicating its relative weight or priority, thus imposing a rule priority principle for comparing arguments based on the idea that an argument is as credible as the least credible of the rules supporting it.

The hypothesis underlying this work is that defeasible argumentation is a reliable tool for dealing with the problem of reasoning with possibly inconsistent ontologies having disjunctive assertions. Therefore, in this paper, we will extend PDeLP for it to be able to handle facts with disjunctions. We call the extension presented here *Suppositional Possibilistic Defeasible Logic Programming* (SPDeLP). SPDeLP is an adaptation of Bodanza's Suppositional Argumentation System [15] for making it suitable for its computational implementation.

We then extend traditional DL ontologies with disjunctive assertional statements of the form *"an individual is a member of the union of two or more classes"*. Each axiom in the ontology is coupled with a weight or certainty degree in order to decide between contradicting axioms about the membership of individuals to concepts, a reasoning task known as instance checking. We chose to call this new kind of ontologies *PDLDA ontologies*. We then interpret PDLDA ontologies as SPDeLP programs in order to answer queries of membership of individual to classes in the presence of inconsistency. We think that this

approach could extend the range of applications in which ontologies are used, such as improving the development of applications that need dealing with uncertain knowledge to perform decision-making. We present a running example that shows how SPDeLP allows to reason on a given PDLDA ontology.

The rest of the article is structured as follows. In Sect. 2, we present the fundamentals of PDLDA ontologies. In Sect. 3, we describe the argumentation approach to reasoning with SPDeLP and how it is used to perform instance checking in PDLDA ontologies. Finally, Sect. 4 concludes.

2 Possibilistic Description Logic Ontologies with Disjunctive Assertions

We introduce here possibilistic description logic ontologies with disjunctive assertions. In brief, they are ontologies with numeric degrees attached to axioms and that feature disjunctive assertions of membership of individuals to concepts and roles. First we briefly recall reasoning in description logics, second the variation proposed for possibilistic description logics, to finally present ontologies with disjunctive assertions.

2.1 A Brief Recall of Description Logics

Description Logics (DL) are a well-known family of knowledge representation formalisms [1]. They are based on the notions of *concepts* (unary predicates, classes) and *roles* (binary relations), and are mainly characterized by the constructors that allow complex concepts and roles to be built from atomic ones. Let C and D stand for concepts and R for a role name. Concept descriptions are built from concept names using the constructors conjunction ($C \sqcap D$), disjunction ($C \sqcup D$), negation ($\neg C$), existential restriction ($\exists R.C$), and value restriction ($\forall R.C$). To define the semantics of concept descriptions, concepts are interpreted as subsets of a domain of interest, and roles as binary relations over this domain. Other constructors include inverse R^- and transitive R^+ roles.

A DL ontology consists of two finite and mutually disjoint sets: a *Tbox* which introduces the *terminology* and an *Abox* (assertional box) which contains facts about particular objects in the application domain. A Tbox contains inclusion axioms $C \sqsubseteq D$, where C and D are (possibly complex) concept descriptions, meaning that every individual of C is also a D, and equality axioms $C \equiv D$ meaning that C and D are equivalent concepts (i.e. every individual in C is an individual in D and vice versa). Objects in the Abox are referred to by a finite number of *individual names* and these names may be used in assertional statements: *concept assertions* of two types: $a : C$ (meaning the individual a is a member of concept C), and *role assertions* of the type $\langle a, b\rangle : R$ (meaning that a is related to b through the role R).

Many reasoning *Abox reasoning* tasks are defined in DL, but in this work we are only interested in *instance checking* that refers to determining if an individual is a member of a certain class.

One form of assigning semantics to a DL ontology is based on the fact that DL is isomorphic with first-order logic restricted to two variables. Then, for example, the inclusion axiom $C \sqsubseteq D$ can be interpreted as the first-order logic formula $(\forall x)(c(x) \rightarrow d(x))$ and an assertion $a : C$ as $c(a)$. Description Logic Programming approaches [16] take advantage of this to interpret such axioms as the Prolog rules "d(X) :- c(X)." and "c(a).", resp. We will apply this later on to redefine instance checking in terms of defeasible argumentation allowing to infer that a is a member of the concept D by finding a proof for the goal ":- d(a)" (see [7] for details).

2.2 Fundamentals of Possibilistic Description Logics

We now recall the fundamentals of possibilistic description logic ontologies. Our presentation is based on [17] and [18]. Let $\mathcal{L}_{DL}$ be a DL description language, a *possibilistic DL ontology* is a set of possibilistic axioms of the form $(\varphi, W(\varphi))$ where φ is an axiom expressed in $\mathcal{L}_{DL}$ and $W(\varphi) \in [0, 1]$ is the degree of certainty (or priority) of φ. Namely, a possibilistic DL ontology Σ is such that $\Sigma = \{(\varphi_i, W(\varphi_i)) : i = 1, \ldots, n\}$. Only somewhat certain information is explicitly represented in a possibilistic ontology. That is, axioms with a null weight $(W(\varphi) = 0)$ are not explicitly represented in the knowledge base. The weighted axiom $(\varphi, W(\varphi))$ means that the certainty degree of φ is at least equal to $W(\varphi)$. A possibilistic DL ontology Σ will also be represented by a pair $\Sigma = (T, A)$ where elements in both T and A may be uncertain. Note that if we consider all $W(\varphi_i) = 1$, then we find a classical DL ontology $\Sigma^* = \{\varphi_i : (\varphi_i, W(\varphi_i)) \in \Sigma\}$. We say that Σ is consistent if the classical ontology obtained from Σ by ignoring the weights associated with axioms is consistent, and inconsistent otherwise.

2.3 Introducing Possibilistic Description Logic Ontologies with Disjunctive Assertions

We now introduce the concept of Possibilistic Description Logic ontology with disjunctive assertions. These new kind of ontologies are the possibilistic DL ontologies introduced above in Sect. 2.2 but with axioms formed in a particular way allowing to translate them into equivalent logic program rules and with assertional statements that encode uncertainty about the membership of an individual to a disjunction of concepts. Gómez et al. [7, 8, 17] exploited the Description Logic Programming approach for translating DL ontologies into logic programming rules to reason on inconsistent DL ontologies in DeLP and PDeLP. These previous works, although they handle a big number of cases, are not able to handle the kind of ontologies we are going to propose here. Handling these kind of ontologies requires a new reasoning framework that we will introduce in Sect. 3.

To understand how the translation from DL to logic programming works, we recall some concepts presented in [7]. In the presentation, we will refer to $\mathcal{L}_b$-classes, $\mathcal{L}_h$-classes and $\mathcal{L}_{hb}$-classes. Remember that our aim is to translate inclusion axioms $C \sqsubseteq D$ into logic programming rules $d(X) \leftarrow c(X)$. Because

of the restrictive form of logic programming rules, $\mathcal{L}_b$-classes C are classes that can appear in the body of logic programming rules, that is in the left-hand side of DL inclusion axioms $C \sqsubseteq D$; $\mathcal{L}_h$-classes D can appear in the head of logic programming rules (e.g. they cannot be existentially quantified), so they appear in the right-hand side of DL inclusion axioms $C \sqsubseteq D$; finally, as DL equality axioms $C \equiv D$ are more restrictive, $\mathcal{L}_{hb}$-classes C and D are such that they have to be able to appear in the head and the body of rules.

Definition 1. *Let $C, C_1, \ldots, C_n$ be an $\mathcal{L}_b$-class, D an $\mathcal{L}_h$-class, A, B $\mathcal{L}_{hb}$-classes, $P, P_1, \ldots, P_n, Q$ properties, a, b individuals. Let T be a set of pairs $(\varphi, W(\varphi))$ whose first component is an inclusion or an equality axiom φ in $\mathcal{L}_{DL}$ coupled with a second component $W(\varphi)$ which is a possibilistic degree between 0 and 1. The axioms φ are of the form $C \sqsubseteq D$, $A \equiv B$, $\top \sqsubseteq \forall P.D$, $\top \sqsubseteq \forall P^-.D$, $P \sqsubseteq Q$, $P \equiv Q$, $P \equiv Q^-$, or $P^+ \sqsubseteq P$. Let A be a set of concept and role assertions $(\varphi, W(\varphi))$ disjoint with T of the form $a : (C_1 \sqcup \ldots \sqcup C_n)$ or $\langle a, b \rangle : (P_1 \sqcup \ldots \sqcup P_n)$. A* suppositional description logic ontology *Σ is a pair (T, A). The set T is called the* possibilistic terminology *(or just possibilistic Tbox), and A the* suppositional possibilistic assertional box *(or just supossitional possibilistic Abox).*

We will present now a paradigmatic example that is based on [15]. We will use this as a running example to show how the approach proposed in this work handles the instance checking reasoning task in the presence of disjunction in assertional boxes of possibly inconsistent ontologies.

Example 1. Consider the ontology $\Sigma_1 = (T, A)$ that expresses that ambulances and school buses are vehicles, regular vehicles are not allowed to park except ambulances and school buses; also it is known that a is an ambulance or an school bus, and also a vehicle:

$$T = \left\{ \begin{array}{l} (\mathsf{Ambulance} \sqcup \mathsf{SchoolBus} \sqsubseteq \mathsf{Vehicle}, 1) \\ (\mathsf{Vehicle} \sqsubseteq \neg \mathsf{Parking}, 0.6) \\ (\mathsf{Ambulance} \sqcup \mathsf{SchoolBus} \sqsubseteq \mathsf{Parking}, 0.9) \end{array} \right\}$$

$$A = \left\{ (\mathsf{a} : (\mathsf{Ambulance} \sqcup \mathsf{SchoolBus}), 1), (\mathsf{a} : \mathsf{Vehicle}, 1) \right\}$$

We now explain intuitively why Σ_1 is inconsistent considering Σ_1^*. Suppose that a is an ambulance, then a is a member of $\mathsf{Parking}$. But as ambulances are vehicles, then a is also a vehicle. Therefore a is a member also of $\neg\mathsf{Parking}$. Then a is a member of $\mathsf{Parking} \sqcap \neg\mathsf{Parking}$, that is a is a member of $\bot$. Absurd. The same reasoning applies when we suppose that a is an school bus. We will present next a framework that will allow to deal with the situations introduced here in an elegant, natural, simple way.

3 Reasoning with Ontologies with Disjunctive Assertions in Suppositional Argumentation Systems

In this section, we deal with the problem of reasoning with possibilistic description logic ontologies with disjunctive assertions (PDLDA ontologies for short).

For this, we first introduce suppositional possibilistic defeasible logic programming (SPDeLP) that is a reasoning framework that we created; it allows to to deal with uncertain information codified as suppositions and statements with numerical degrees of certainty. This framework is adaptation of Bodanza's suppositional argument system [15]. Second, we show PDLDA ontologies are expressed as SPDeLP programs and how reasoning task in ontologies are interpreted in this framework. Finally, we discuss differences and similarities of our interpretation of suppositional argumentation with the original framework proposed by Bodanza.

3.1 SPDeLP: Suppositional Possibilistic Defeasible Logic Programming

Suppositional argumention systems (SAS) introduced by Bodanza [15] provide a foundation for dealing intuitively with disjunctive information in a defeasible reasoning framework. Bodanza's view is that suppositional reasoning is present in defeasible arguments involving disjunctions, just as reasoning by cases appears in classical logic. Disjunctive information can express different plausible alternatives, and SAS study in what extent an argument assuming such possible alternatives can be considered relevant on the basis of its explicative power in comparison with other explanations. In this work we will present a adaptation of Bodanza's SAS to put emphasis in its implementability, thus providing a simpler presentation. We will adapt the language of Possibilistic Defeasible Logic Programming (PDeLP) [14] for allowing to represent facts having disjunctive literals.

A *suppositional possibilistic defeasible logic (SPDeLP) program* $\mathcal{P}$ is a set of rules $(P \leftarrow Q_1, \ldots, Q_n, \alpha)$ where $0 < \alpha \leq 1$ is a possibilistic degree. When $n = 0$, we will note them simply as (P, α). Facts have the form $(P_1 \text{ OR } \ldots \text{ OR } P_k, \alpha)$, where the P_i are either unary or binary predicates, meaning that at least one $P_i, i = 1, \ldots, k$ holds with degree α (when $k = 1$ we have traditional PDeLP facts). P, Q_1, $\ldots$, Q_n are called literals and can be positive or negative atoms (that is classically negated with $\sim$). A set of rules is contradictory iff it allows to derive a pair of complementary literals L and $\sim L$ with some degree α_1 and α_2, resp. The tentative conclusions of the system are called, as usual, *arguments* and are built by the rules of derivation presented below.

Definition 2. *Let $\mathcal{P}$ be a SPDeLP program and α be a real number such that $0 < \alpha \leq 1$. An* argument *is a structure $\langle \mathcal{D}, \mathcal{S}, H, \alpha \rangle$, where $\mathcal{D}$ is a finite set of ground instances of rules in $\mathcal{P}$ (called the argument's* defeasible support*), $\mathcal{S}$ is a set of ground literals called* suppositions*, and H is a ground literal called the argument* conclusion *such that $\mathcal{D} \cup \mathcal{S}$ allows to derive H with strength α derived minimally according to the following deductive rules:*

- ***Fact (Fact):*** *If $(H, \alpha) \in \mathcal{P}$, then it holds that $\langle \emptyset, \emptyset, H, \alpha \rangle$ is an argument.*
- ***Supposition (Sup):*** *If $(H_1 \text{ OR } \ldots \text{ OR } H_n, \alpha) \in \mathcal{P}$, then it holds that $\langle \emptyset, \{H_i\}, H_i, \alpha \rangle$ are arguments for $i = 1, \ldots, n$.*

- ***Generalized Modus Ponens (GMP):*** *If* $\langle \mathcal{D}_1, \mathcal{S}_1, B_1, \alpha_1\rangle$, ..., $\langle \mathcal{D}_n, \mathcal{S}_n, B_n, \alpha_n\rangle$, $(H \leftarrow B_1, \ldots, B_n, \alpha) \in \mathcal{P}$ *and* $(\bigcup_{i=1}^{n} \mathcal{D}_i) \cup (\bigcup_{i=1}^{n} \mathcal{S}_i) \cup \{H \leftarrow B_1, \ldots, B_n\}$ *is not contradictory, then* $\langle \bigcup_{i=1}^{n} \mathcal{D}_i \cup \{H \leftarrow B_1, \ldots, B_n\}, \bigcup_{i=1}^{n} \mathcal{S}_i, H, min(\alpha_1, \ldots, \alpha_n, \alpha)\rangle$ *is an argument.*
- ***Constructive Dilemma (CD):*** *If* $\langle \mathcal{D}_i, \{H_i\} \cup \mathcal{S}_i, B, \alpha_i\rangle$ *for* $i = 1, \ldots, n$ *and* $(H_1 \text{ OR } \ldots \text{ OR } H_n, \alpha) \in \mathcal{P}$, *and* $(\bigcup_{i=1}^{n} \mathcal{D}_i) \cup (\bigcup_{i=1}^{n} \mathcal{S}_i)$ *is not contradictory, then* $\langle \bigcup_{i=1}^{n} \mathcal{D}_i, \bigcup_{i=1}^{n} \mathcal{S}_i, B, min(\alpha_1, \ldots, \alpha_n, \alpha)\rangle$ *is an argument.*

Notice that the CD rule is proposed based on the intuition that in propositional logics r can be inferred from $p \vee q$, $p \rightarrow r$ and $q \rightarrow r$. Also, we will say that an argument $\langle \mathcal{D}, \mathcal{S}, H, P\rangle$ is *self-defeating* iff $\mathcal{D} \cup \mathcal{S}$ allows to derive a pair of contradictory literals.

Property 1. There are no self-defeating arguments in SPDeLP.

We refine the usual notion of disagreement: a set of arguments S is in disagreement if there are at least two arguments for some L and $\sim L$ in S that *have the same set of suppositions*. If S is not in disagreement, it is said to be in agreement. The conclusions of the system are obtained in the same way as in traditional PDeLP—through a dialectical process that considers all the arguments for a query, then all of its defeaters and defeaters for those defeaters and so on. Given two (contradicting) arguments, if the attacking argument is strictly preferred over the attacked one (that is its weight is greater), then it is called a *defeater*. Given a SPDeLP program $\mathcal{P}$ and a query H, the final answer to H w.r.t. $\mathcal{P}$ is based on such dialectical analysis. The answer to a query can be: *Yes* (when there exists a warranted argument $\langle \mathcal{D}, \mathcal{S}, H, \alpha\rangle$), *No* (when there exists a warranted argument $\langle \mathcal{D}, \mathcal{S}, \sim H, \alpha\rangle$), *Undecided* (when neither $\langle \mathcal{D}, \mathcal{S}, H, \alpha\rangle$ nor $\langle \mathcal{D}, \mathcal{S}, \sim H, \alpha\rangle$ are warranted), or *Unknown* (when H does not belong to $\mathcal{P}$).

We now introduce how the ontology presented in Ex. 1 is treated in SPDeLP. In Sect. 3.2, we will discuss how the translation from DL to SPDeLP is achieved.

(1) $(vehicle(X) \leftarrow ambulance(X), 1)$
(2) $(vehicle(X) \leftarrow schoolbus(X), 1)$
(3) $(\sim parking(X) \leftarrow vehicle(X), 0.6)$
(4) $(parking(X) \leftarrow ambulance(X), 0.9)$
(5) $(parking(X) \leftarrow schoolbus(X), 0.9)$
(6) $(ambulance(a)$ OR $schoolbus(a), 1)$
(7) $(vehicle(a), 1)$

Fig. 1. Program $\mathcal{P}_1$ from Example 2

Example 2. (Continues Ex. 1) The ontology Σ_1 is interpreted as the program $\mathcal{P}_1$ in Fig. 1. Some of the arguments we can build from this program are:

(8) $\mathcal{A}_1 = \langle \emptyset, \emptyset, vehicle(a), 1\rangle$ (by Fact from (7))
(9) $\mathcal{A}_2 = \langle \emptyset, \{ambulance(a)\}, ambulance(a), 1\rangle$ (by Sup from (6))
(10) $\mathcal{A}_3 = \langle \emptyset, \{schoolbus(a)\}, schoolbus(a), 1\rangle$ (by Sup from (6))
(11) $\mathcal{A}_4 = \langle \{\sim parking(a) \leftarrow vehicle(a)\}, \emptyset, \sim parking(a), 0.6\rangle$ (by GMP from (3) and (8))

(12) $\mathcal{A}_5 = \langle\{(parking(a) \leftarrow ambulance(a)\}, \{ambulance(a)\}, parking(a), 0.9\rangle$ (by GMP from (4) and (9))
(13) $\mathcal{A}_6 = \langle\{parking(a) \leftarrow schoolbus(a)\}, \{schoolbus(a)\}, parking(a), 0.9\rangle$ (by GMP from (5) and (10))
(14) $\mathcal{A}_7 = \langle\{parking(a) \leftarrow ambulance(a), parking(a) \leftarrow schoolbus(a)\}, \emptyset, parking(a), 0.9\rangle$ (by CD from (6), (12) and (13)).

We see that $\mathcal{A}_5$, $\mathcal{A}_6$ and $\mathcal{A}_7$ are in agreement and disagree with $\mathcal{A}_4$. Argument $\mathcal{A}_7$ defeats argument $\mathcal{A}_4$, therefore a can park because $\mathcal{A}_4$ is warranted. Notice that $\mathcal{A}_5$ and $\mathcal{A}_6$ cannot attack $\mathcal{A}_4$ because, even entailing opposing conclusions, the set of suppositions for each argument are different.

3.2 Interpreting Suppositional Possibilistic DL Ontologies in SPDeLP

For assigning semantics to a description logics ontology, we define a translation function $\mathcal{T}(\cdot)$ from DL to SPDeLP based on the work of [16] (for details, see [7] to study the case in which DL ontologies are translated into DeLP). First, axioms are considered to be in negation-normal form, meaning that negations are pushed inward class expressions. Informally, an axiom of the form $\mathsf{C} \sqsubseteq \mathsf{D}$ will be translated as $d(X) \leftarrow c(X)$. Abox assertions of the form $\mathsf{a} : (\mathsf{C_1} \sqcup \ldots \sqcup \mathsf{C_n})$ are translated as facts $c_1(a)$ OR $\ldots$ OR $c_n(a)$ and $\langle \mathsf{a}, \mathsf{b}\rangle : (\mathsf{r_1} \sqcup \ldots \sqcup \mathsf{r_n})$ as $r_1(a,b)$ OR $\ldots$ OR $r_n(a,b)$. Moreover, a formula of the form $\exists \mathsf{r}.\mathsf{C} \sqsubseteq \mathsf{D}$ is translated as $d(X) \leftarrow r(X,Y), c(Y)$, and one of the form $C \sqcup D \sqsubseteq E$ as two axioms $C \sqsubseteq E$ and $D \sqsubseteq E$. See Fig. 2 for a formal account of the translation function. The interpretation of Σ is a SPDeLP program $\mathcal{P} = \mathcal{T}(T) \cup \mathcal{T}(A)$.

Instance checking is redefined to handle possible inconsistencies while retaining classical DL functionality: If C is a class,a an individual and α a real number between 0 and 1, then (i) a is a potential member of C iff there exists an argument $\langle \mathcal{A}, \emptyset, C(a), \alpha\rangle$ w.r.t. $\mathcal{P}$, and, (ii) a is a justified member of C iff there exists a warranted argument $\langle \mathcal{A}, \emptyset, C(a), \alpha\rangle$ w.r.t. $\mathcal{P}$.

Example 3. (Continues Ex. 2) Consider again Σ_1. The individual a is a justified member of the concept $\mathsf{Parking}$.

3.3 Discussion

Here we briefly discuss how our proposal for suppositional argumentation differs from the one presented by Bodanza in [15]. Bodanza includes a *deduction* rule that is too powerful to be implemented because it relies in deduction in first-order classical logic (see [15, p. 27]). We instead chose to include a simpler rule such as CD. We also chose to drop the *conditionalization* rule of Bodanza; this makes our system less expressive in the sense that we cannot discharge suppositions (except by using CD) and our system does not allow reasoning by contrapositive reasoning. Bodanza's approach does not use weights for rules but instead uses generalized specificity to compare arguments. Our inclusion of numerical weights obscures the knowledge representation but gives more control to the knowledge engineer and simplifies the implementation of the reasoning system.

$$
\begin{array}{rl}
\mathcal{T}(\{C \sqsubseteq D\}) =_{df} & \{\, T_h(D,X) \leftarrow T_b(C,X) \,\}, \text{if } C \text{ is an } \mathcal{L}_b\text{-class and } D \text{ an } \mathcal{L}_h\text{-class} \\
\mathcal{T}(\{C \equiv D\}) =_{df} & \mathcal{T}(\{C \sqsubseteq D\}) \cup \mathcal{T}(\{D \sqsubseteq C\}), \text{if } C \text{ and } D \text{ are } \mathcal{L}_{hb}\text{-classes} \\
\mathcal{T}(\{\top \sqsubseteq \forall P.D\}) =_{df} & \{\, T_h(D,Y) \leftarrow P(X,Y) \,\}, \text{if } D \text{ is an } \mathcal{L}_h\text{-class} \\
\mathcal{T}(\{\top \sqsubseteq \forall P^-.D\}) =_{df} & \{\, T_h(D,X) \leftarrow P(X,Y) \,\}, \text{if } D \text{ is an } \mathcal{L}_h\text{-class} \\
\mathcal{T}(\{a : (D_1 \sqcup \ldots \sqcup D_n)\}) =_{df} & \{\, T_h(D_1,a) \text{ OR } \ldots \text{ OR } T_h(D_n,a) \,\}, \text{if } D_i \text{ are } \mathcal{L}_h\text{-classes} \\
\mathcal{T}(\{\langle a,b\rangle : (P_1 \sqcup \ldots \sqcup P_n)\}) =_{df} & \{\, P_1(a,b) \text{ OR } \ldots \text{ OR } P_n(a,b) \,\} \\
\mathcal{T}(\{P \sqsubseteq Q\}) =_{df} & \{\, Q(X,Y) \leftarrow P(X,Y) \,\} \\
\mathcal{T}(\{P \equiv Q\}) =_{df} & \left\{ \begin{array}{l} Q(X,Y) \leftarrow P(X,Y) \\ P(X,Y) \leftarrow Q(X,Y) \end{array} \right\} \\
\mathcal{T}(\{P \equiv Q^-\}) =_{df} & \left\{ \begin{array}{l} Q(X,Y) \leftarrow P(Y,X) \\ P(Y,X) \leftarrow Q(X,Y) \end{array} \right\} \\
\mathcal{T}(\{P^+ \sqsubseteq P\}) =_{df} & \{\, P(X,Z) \leftarrow P(X,Y) \wedge P(Y,Z) \,\} \\
\mathcal{T}(\{s_1,\ldots,s_n\}) =_{df} & \bigcup_{i=1}^{n} \mathcal{T}(\{s_i\}), \text{ if } n > 1 \\
\textbf{where:} & \\
T_h(A,X) =_{df} & A(X) \\
T_h((C \sqcap D),X) =_{df} & T_h(C,X) \wedge T_h(D,X) \\
T_h((\forall R.C),X) =_{df} & T_h(C,Y) \leftarrow R(X,Y) \\
T_b(A,X) =_{df} & A(X) \\
T_b((C \sqcap D),X) =_{df} & T_b(C,X) \wedge T_b(D,X) \\
T_b((C \sqcup D),X) =_{df} & T_b(C,X) \vee T_b(D,X) \\
T_b((\exists R.C),X) =_{df} & R(X,Y) \wedge T_b(C,Y)
\end{array}
$$

Fig. 2. Mapping $\mathcal{T}$ from DL ontologies with assertions to SPDeLP rules

4 Conclusions and Future Work

This paper proposed a kind of description logic ontologies that allow to represent disjunctive assertions of membership of individuals to concepts and roles where the ontology axioms have been assigned possibilistic degrees of certainty, and also proposed a variation of suppositional argumentation to handle the reasoning task of instance checking in such ontologies. The idea is to translate the ontologies into a possibilistic suppositional argumentation system in order to perform the reasoning on the ontologies. For every disjunctive assertion, if supposing each disjunct allows the system to reach a conclusion, then that conclusion can be taken for granted. As the ontology can be potentially inconsistent, the reasoning framework takes this possibility into account to compute defeaters and defeaters for these defeaters to perform a dialectical reasoning in order to warrant conclusions. The possibilistic degrees of certainty are used to compute the relative weight of each possible conclusion, which, in turn, are used to decide which arguments prevail. Much work remains to be done such us proposing larger case studies. With respect to knowledge representation aspects, we would like to adapt the system to have disjunctions in the head of rules. It also appears to be interesting to consider different argumentation semantics for the definition of the dialectical process.

Acknowledgments: This research is funded by Secretaría General de Ciencia y Técnica, Universidad Nacional del Sur, Argentina. The author would like to thank the anonymous reviewers for their comments.

References

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P., eds.: The Description Logic Handbook – Theory, Implementation and Applications. Cambridge University Press (2003)
2. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American (2001)
3. Moguillansky, M.O., Falappa, M.A.: A non-monotonic Description Logics model for merging terminologies. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial **11**(35) (2007) 77–88
4. Moguillansky, M.O., Rotstein, N.D., Falappa, M.A.: Generalized Abstract Argumentation: A First-order Machinery towards Ontology Debugging. Inteligencia Artificial **46** (2010) 17–33
5. Huang, Z., van Harmelen, F., ten Teije, A.: Reasoning with Inconsistent Ontologies. In Kaelbling, L.P., Saffiotti, A., eds.: Proc. 19th International Joint Conference on Artificial Intelligence (IJCAI'05), Edinburgh, Scotland (August 2005) 454–459
6. Gómez, S.A., Simari, G.R.: Merging of ontologies using belief revision and defeasible logic programming. Inteligencia Artificial **16**(52) (2013) 16–28
7. Gómez, S.A., Chesñevar, C.I., Simari, G.R.: Reasoning with Inconsistent Ontologies Through Argumentation. Applied Artificial Intelligence **1**(24) (2010) 102–148
8. Gómez, S.A., Chesñevar, C.I., Simari, G.R.: ONTOarg: A Decision Support Framework for Ontology Integration based on Argumentation. Expert Systems with Applications **40** (2013) 1858–1870
9. Antoniou, G., Bikakis, A.: DR-Prolog: A System for Defeasible Reasoning with Rules and Ontologies on the Semantic Web. IEEE Transactions on Knowledge and Data Engineering **19**(2) (2007) 233–245
10. Chesñevar, C.I., Maguitman, A., Loui, R.: Logical Models of Argument. ACM Computing Surveys **32**(4) (December 2000) 337–383
11. Bench-Capon, T.J.M., Dunne, P.E.: Argumentation in artificial intelligence. Artificial Intelligence **171**(10-15) (2007) 619–641
12. Rahwan, I., Simari, G.R.: Argumentation in Artificial Intelligence. Springer (2009)
13. García, A., Simari, G.: Defeasible Logic Programming an Argumentative Approach. Theory and Practice of Logic Programming **4**(1) (2004) 95–138
14. Alsinet, T., Chesñevar, C.I., Godo, L.: A level-based approach to computing warranted arguments in possibilistic defeasible logic programming. In Besnard, P., Doutre, S., Hunter, A., eds.: COMMA. Volume 172 of Frontiers in Artificial Intelligence and Applications., IOS Press (2008) 1–12
15. Bodanza, G.: Disjunctions and Specificity in Suppositional Defeasible Argumentation. Logic Journal of the Interest Group in Pure and Applied Logics **10**(1) (2002) 23–49
16. Grosof, B.N., Horrocks, I., Volz, R., Decker, S.: Description Logic Programs: Combining Logic Programs with Description Logics. WWW2003, May 20-24, Budapest, Hungary (2003)
17. Gómez, S.A., Chesñevar, C.I., Simari, G.R.: Using Possibilistic Defeasible Logic Programming for Reasoning with Inconsistent Ontologies. In Giusti, A.D., Diaz, J., eds.: Computer Science & Technology Series. XVII Argentine Congress of Computer Science Selected Papers. (2012) 19–29
18. Benferhat, S., Bouraoui, Z., Lagrue, S., Rossit, J.: Merging Inconmensurable Possibilistic DL-Lite Assertional Bases. In Papini, O., Benferhat, S., Garcia, L., Mugnier, M.L., eds.: Proceedings of the IJCAI Workshop 13 Ontologies and Logic Programming for Query Answering. (2015) 90–95

Imperialist Competitive Algorithm for the Flowshop Problem

Gabriela Minetti[1] and Carolina Salto[1,2]

[1] Facultad de Ingeniera, Universidad Nacional de La Pampa
Calle 110 N390, General Pico, La Pampa, Argentina
[2] CONICET
{minettig,saltoc}@ing.unlpam.edu.ar

Abstract. This article presents a new optimization techniques based on Imperialistic Competitive Algorithm to solve the flowshop scheduling problems, which objective is to minimize the makespan. Furthermore, this approach is enhanced by a local search procedure in order to improve the best solutions. To show the efficiency of the proposed methods, we consider many instances of increasing complexity for the flowshop problem. Computational tests are presented and comparisons between the two approaches are made. The analysis of the results suggests that the enhanced approach is capable to find the best solutions of the problem at hand.

Keywords: Imperialist Competitive Algorithm, optimization, flowshop problem

1 Introduction

In many manufacturing and assembly facilities a number of operations have to be completed on every job. Often, these operations must be done on all jobs in the same order, which implies that the jobs have to follow the same route through a set of machines. This problem is known as the Flowshop Sequencing Problem (FSSP) [10], which objective is to determine the sequence of the n jobs so that a certain performance measure will be optimized. The most commonly studied performance measure is the minimization of makespan (c_{max}).

Since it has been proven that the FSSP with makespan minimization is NP-complete when the number of machines is greater or equal than three [8], there have been proposed exact, (constructive and improvement) heuristic, and (trajectory and population-based) metaheuristic methods. The exact methods need exponential computation time in most cases, as a result they are feasible for only small size problems, as shown in [12, 13, 21]. On the one hand, many constructive heuristic methods have been introduce since 1954 in [6, 10, 11, 15, 17]. On the other hand, the improvement heuristics enhance an existing initial FSSP solution using a local search method [5, 22]. Concerning the metaheuristics for FSSP, some research are based on trajectory methods, as we can see in [7, 16,

20]. Contrary to trajectory methods, population-based metaheuristics have been proposed to solve FSSP in [4, 18, 19, 23].

The Imperialist Competitive Algorithm (ICA) [1] is a new population-based metaheuristic that is inspired by the imperialistic competition, where the countries (solutions to the problem at hand) are divided into two types: imperialist and colonies. The base of the algorithm is the imperialistic competition, where the colonies move toward the empires, empires compete among them and the weak ones collapse. ICA has been successfully used to solve several NP-Hard optimization problems, e.g. non-convex dynamic economic power dispatch [3], constrained economic dispatch [14], complicated image matching [9], among others. The objective of this work is to research the potential of this metaheuristic to solve the FSSP in an efficient way. By doing this, we propose the ICA_{FSSP} algorithm to solve problems with solution represented by permutations. Consequently, we analyse their performance using different operators to produce the movements of the colonies toward the imperialistic countries. Also, we present an improved ICA_{FSSP} called ImpICA_{FSSP}, which implements an improvement procedure in order to simulate the growth of the imperialistic countries. We analyze the behavior of both ICAs over a wide range of complex FSSP instances.

The rest of this article is organized as follows. Section 2 describes in detail the flowshop problem. In Section 3, we describe the metaheuristic ICA to solve the FSSP (ICA_{FSSP}). In Section 4, we outline an improved version of the ICA_{FSSP}, called ImpICA_{FSSP}. Section 5 explains the experimental design. Then, we study and analyze the results obtained by ICA_{FSSP} and ImpICA_{FSSP} for FSSP in Section 6. Finally, we present our principal conclusions and future lines of research in Section 7.

2 FSSP Description

The flowshop scheduling problem is generally described as follows: each job from the job set $J = \{1, 2...n\}$, for $n > 1$, has to be processed on m machines in the order given by the indexing of the machines $(1, 2..., m)$. Thus job j, $j \in J$, consists of a sequence of m operations; each of them corresponds to the processing of job j on machine i for an uninterrupted processing time $t_{ji} \geq 0$. It is assumed that a zero processing time on a machine corresponds to a job performed by that machine in an infinitesimal time. Machine i, $i = 1, 2...m$, can execute at most one job at a time, and it is assumed that each machine processes the jobs in the same order. The objective is to find a sequence of jobs that minimizes the maximum flow time, which is called *makespan* (c_{max}).

If we let $c(j_j, i)$ denote the completion time of job j_j on machine i and let $\{j_1, j_2, ..., j_n\}$ denote a job permutation of J, then we can calculate the completion times for a n jobs and m machines flowshop problem as shown in Equation 1, where the final makespan is $c_{max} = c(j_n, m)$.

$$\begin{array}{ll} c(j_1, 1) = t_{j_1 1}, & \\ c(j_1, i) = c(j_1, i-1) + t_{j_1 i}, & i = \{2, ..., m\} \\ c(j_j, 1) = c(j_{j-1}, 1) + t_{j_j 1}, & j = \{2, ..., n\} \\ c(j_j, i) = \max\{c(j_{j-1}, i), c(j_j, i-1)\} + t_{j_j i}, & j = \{2, ..., n\}, i = \{2, ..., m\} \end{array} \quad (1)$$

As we can see, this kind of problem is essentially a permutation schedule problem, and the permutation of jobs can be naturally represented by a sequence of positive integers. For example, $S = [2, 3, 1, 4]$ means that the corresponding job sequence is j_2, j_3, j_1, and j_4.

3 Imperialist Competitive Algorithm for the FSSP

The ICA was first proposed in [1], inspired by the imperialistic competition. It is a population-based metaheuristic for solving optimization problems. It starts with an initial population of countries (solutions), which are categorized into two groups: imperialist (the best solutions) and colonies (the remaining solutions). Then, the algorithm iterates over a set of steps to improve the population, producing imperialistic competitions among these empires. Powerful imperialists try to increase in the power and absorb colonies to their empire from weak imperialists, whose power gradually decreases. When an empire loses all of its colonies, this empire collapses. The competition ends when only one empire exists, and the best solution of the problem is represented by its imperialistic country.

In this article, we propose an ICA to solve a permutation-based problem like the flowshop problem, denominated ICA_{FSSP}. The flowchart of this algorithm is presented in Figure 1. The steps of the ICA_{FSSP} are described as follows:

Step 1: Initialize all countries. An initial size set, N_C, of countries is created by a random procedure.

Step 2: Determine the imperialistic countries. After calculating the fitness function to each country $c_i = 1/c_{max}$, the best N_{imp} of them are selected as imperialists. The rest $N_{col} = N_C - N_{imp}$ countries are the colonies.

Step 3: Assign the colonies to each imperialist according to ist power. The colonies are divided among imperialists based on their power, in order to form the initial empires. To calculate the power p_j of each imperialistic country, first a normalized power of each one is calculated according to Equation 2.

$$p_j = \frac{c_j}{\sum_{i=1}^{N_{imp}} c_i} \tag{2}$$

The number of colonies of each imperialistic country (N_{col_j}) is proportional to its power, and it is determined by Equation 3 [3]. In order to assign colonies to an imperialist j, N_{col_j} colonies are randomly selected between the remaining ones.

$$N_{col_j} = p_j \times N_{col} \tag{3}$$

Step 4: Move the colonies toward their relevant imperialist. The colonies in each of the empires start moving toward their relevant imperialistic country. In this work, as the solutions are represented by permutations, those movements are accomplished by crossover and mutation operators. The crossover operator is applied in such a way that a colony and its imperialist country are considered as parents. The most popular and standard

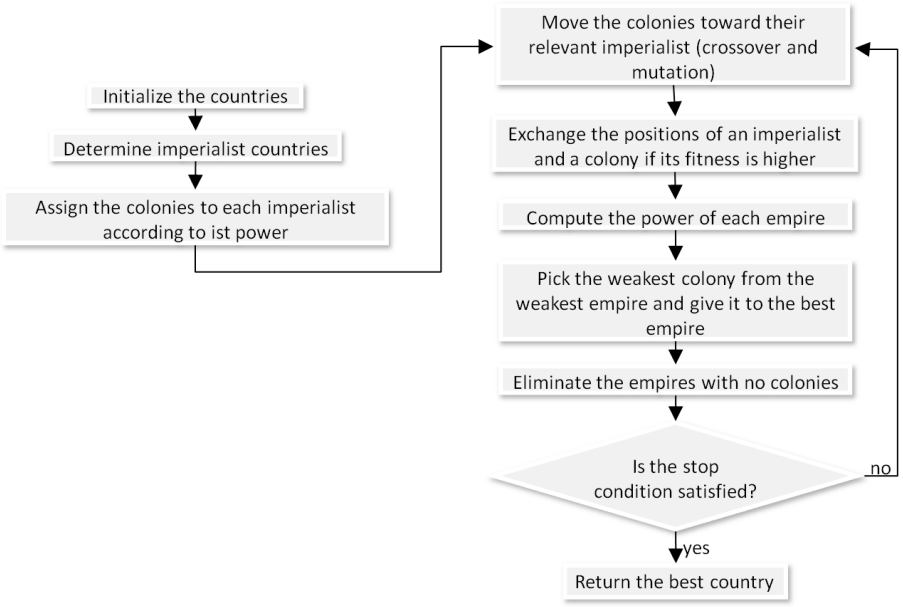


Fig. 1: The flowchart of the ICA_{FSSP}.

permutation-based crossover operators are the PMX and the OX. Regarding mutation operators, they are applied to each colony in order to simulate a randomly deviated direction, as the original proposal of ICA. In particular, we can use insert, pairwise interchange and inversion operator for permutation-based representation.

Step 5: Exchange the positions between the imperialist and colony if the colony fitness is higher than imperialistic one. When moving toward the imperialistic country, a colony might reach a position with higher fitness than of its related imperialist country. In this case, the imperialist and the colony should change their positions. If there are several colonies better than the imperialistic country, then the best colony will replace the imperialist.

Step 6: Compute the total power of each empire. The total power of each empire (tp_j) is computed including the power of the imperialistic country and the power of its colonies, according to the Equation 4 [1]. In this equation appears the parameter $\sigma \in [0, 1]$, which causes that the total power of the empire might be determined by just the imperialistic country ($\sigma = 0$) or by the colonies increasing the σ value. We adopt the value of $\sigma = 0.1$ in our implementation as suggested in [1].

$$tp_j = c_j + \sigma \times \sum_{i=1}^{N_{col_j}} c_i \tag{4}$$

Step 7: Pick the weakest colony from the weakest empire and give it to the best empire. Select the empire with the highest tp_j as the best one and increase its number of colonies by one ($N_{col_j} + 1$). The colony of the weakest empire, k, with the lowest c_i is considered as the weakest one. Decrease the number of colonies of the weakest empire by one ($N_{col_k} - 1$).

Step 8: Eliminate the empires with no colonies. The number of empires decreases by one ($N_{imp} - 1$) if the weakest empire j has $N_{col_k} = 0$.

Step 9: Stop condition. If more than one empire remained ($N_{imp} > 1$) and the $time < time_{max}$, then go to Step 4.

Step 10: End. The best imperialistic country is returned as the best solution to the optimization problem.

Finally, we remark that ICA algorithms only need two parameters: the population size (N_C) and the empire number (N_{imp}). As a consequence, the time incurred in fine-tunning the parametric values is drastically reduced.

4 Improvements to the ICA$_{FSSP}$

The ICA$_{FSSP}$ for the FSSP, discussed above, is very simple and easy to implement. However, it can become inefficient because no improvements are applied to the imperialistic country, i.e. the imperialist solution, does not undergo to any improvement procedure, while their colonies move to it (Step 4) and no one reaches a better position (Step 5).

Consequently, we suggest a modification to the ICA$_{FSSP}$ to overcome the problem mentioned in the previous paragraph. The resultant algorithm is named ImpICA$_{FSSP}$, which incorporates a local search procedure to improve the imperialistic country. Introducing this improvement step (after Step 4) the imperialists try to become more powerful and continue with their dominance. The local search procedure applies repeatedly the interchange operator on the imperialistic country and works as follows: the first job in the first position of the imperialist solution x, denominated x_1, is interchanged for a job in a different randomly selected position. If this new solution x' has a higher fitness value, the current imperialistic solution is replaced by the new x' and the local search stops. If no improvement is found, the procedure continues with the job in the next position x_2. The search iterates through all the jobs x_i, without repetition. The basis of this procedure is to make a very quick local search.

5 Experimental Design

In this section, we describe the experimental design used in this approach and the execution environment. In order to evaluate the proposed algorithms ICA$_{FSSP}$ and ImpICA$_{FSSP}$ to solve the FSSP, we have selected a wide range of instances used in the literature taking into a account the complexity of them. This complexity is given by the number of jobs and machines. The Taillard's instances with 20, 50, 100, 200, and 500 jobs and 10 and 20 machines were chosen, whose

best known values are available in the OR Library [2]. In this benchmark set, 10 instances for each problem size are provided totalizing 90 instances.

Due to the stochastic nature of the algorithms, we performed 30 independent runs of each trial to gather meaningful experimental data and to apply statistical confidence metrics to validate our results and conclusions. Consequently, a total of 2700 executions (90 instances $\times$ 30 independent runs) were carried out by each algorithm. The hardware involved in the experimentation was an INTEL I7 3770K, 8 GB RAM, and the Slackware Linux with 2.6.27 kernel version.

To evaluate our algorithms, we use as performance measure the relative difference (*gap*) between the cost of the best solution found by the algorithm ($BestCost$) in each trial and the best known cost ($BestKnownCost$) for each instance . The *gap* is calculated as shown in Equation 5.

$$gap = \frac{Best\,Cost - Best\,Known\,Cost}{Best\,Known\,Cost} \tag{5}$$

The parameters that our ICA algorithms use are the population size, the number of empires, and the complementary stop condition. The size of the population (N_C) of a metaheuristic is crucial to the performance of the algorithm and as the ICA is not the exception, two different values of N_C are considered: 50 and 100 (these values are widely used in the literature). After an experimentation on all instances, the *gap* values found by ICA$_{FSSP}$ with N_C=100 are less than the ones of case of N_C=50. The Kruskal-Wallis test confirms this situation, indicating that there are statistically significant differences between groups of ranks (p-values higher than 0.01, the significance level). For that reason, N_C=100 is the country size used in the remaining experimentation. The value of N_{imp} is set to the 10% of N_C because it is a typical set in the literature. The following criteria are used to determine the stop condition: the existence of an only one empire or to reach the maximum time $time_{max}$ of execution fixed in $n \times n/2 \times 30ms$ [19].

6 Result Analyisis

An analysis of the results obtained by ICA_{FSSP} and ImpICA_{FSSP} is presented in this section. First, we compare the ICA_{FSSP}'s behavior considering different combinations of crossover and mutation operators. Finally, we study the performance of ICA_{FSSP} and ImpICA_{FSSP} algorithms.

6.1 Analysis of ICA_{FSSP} behavior

In this section, we investigate the impact of different combinations of crossover and mutation operators on the ICA_{FSSP} search. Recall that these operators are applied in the process of movement of the colonies toward their imperialistic country. Two well known permutation-based crossover operators, such as OX and PMX, and three mutation operators, insertion (ins), interchange (int), and

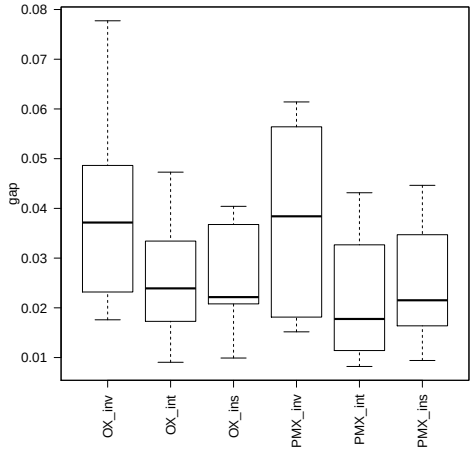


Fig. 2: Boxplot of the *gap* values for ICA_{FSSP} and each combination of operators

Table 1: Duncan test

Groups	Combinations
a	OX_inv
a	PMX_inv
b	OX_int
b	PMX_ins
c	OX_ins
c	PMX_int

inversion (inv) are considered in this analysis. Consequently, a total of six different combinations of these operators are obtained (for example, OX_inv means the use of OX as crossover operator and inversion as mutation operator). ICA_{FSSP} with each combination of operators was run for the whole set of instances.

The first considered quality indicator is the *gap* value. Figure 2 provides a revealing summary of this indicator by means of a boxplot considering the gaps values obtained for the total of the instances. In general, the combinations with interchange mutation give the best *gap* values, followed by insertion. The boxes with the inversion mutation are much higher than the remaining ones, suggesting differences between groups. Furthermore, we observe that the best combination is the PMX_int, whose gap values are between 0.013 and 0.032. Moreover, the minimum *gap* value (0.008) is obtained for ICA_{FSSP} with the PMX_int combination, followed by the OX_ins (0.009 *gap* value).

Given that the p-value of the Friedman test is less than the significance level ($\alpha = 0.01$), we can conclude that significant differences arise between the observed results. Attending to this result, a post-hoc statistical analysis help us to detect concrete differences between algorithms. For this purpose, we use the Duncan's Multiple Range Test and present the Table 1 for the duncan's output, where combinations with the same letter are not significantly different. In average, the worst results are obtained by the two combinations with the inversion operator (group a). In contrast, the minimum average gap values are found by the OX_ins and PMX_int combinations belonging to the group c. Consequently, the best combination of operators is given by PMX and interchange, as crossover and mutation operator, respectively.

Another interesting measure for the ICA_{FSSP} with the different operator combinations is the runtime to find the best solution, as shown in Table 2. In

Table 2: Average of the ICA_{FSSP} runtime values to find the best solution considering different operators and for each problem size.

Instance	OX			PMX		
	Inv.	**Int.**	**Ins.**	**Inv.**	**Int.**	**Ins.**
20×10	0.07	0.08	**0.05**	0.07	0.09	**0.05**
20×20	0.16	0.12	**0.08**	0.15	0.11	0.11
50×10	1.29	0.78	0.65	1.01	**0.64**	1.03
50×20	**1.03**	1.83	1.43	1.55	1.34	1.87
100×10	5.74	5.81	**1.93**	7.45	5.31	6.15
100×20	13.52	12.28	**8.07**	11.39	10.29	14.54
200×10	37.26	30.29	31.56	37.17	**19.85**	21.30
200×20	76.78	78.04	75.94	78.62	**51.28**	89.96
500×20	907.84	900.65	565.02	801.05	**540.60**	735.88
Average	115.96	114.43	76.08	104.27	**69.95**	96.76

this case, the mean runtime in seconds is displayed. In average, ICA_{FSSP} with PMX_int is the fastest approach.

Summarizing, we can conclude that ICA_{FSSP} with PMX_int is the best approach because the best *gap* values, meaning best schedules, are obtained in the minimum execution time. Therefore, PMX operator and interchange mutation are the operators selected to be used in the following experimentation.

6.2 Comparison between ICA$_{FSSP}$ and ImpICA$_{FSSP}$

In this section, we compare the behaviour of ICA$_{FSSP}$ and its improved version ImpICA$_{FSSP}$ on the FSSP, see Table 3. Figures in the two first columns stand for the average of the minimum best *gap* values (column Gap$_{Best}$) and the mean best *gap* values (column Gap$_{Mean}$). While, figures in the last column present the average number runtime (in seconds) to reach the best value (Time$_{Best}$). The minimum values of Gap$_{Mean}$ column are bolded and the symbol "$*$" indicates that the respective p-values for t-test are less than the 0.01 significance level.

From the results of Table 3, we can observe that the mean best *gap* obtained by ImpICA$_{FSSP}$ is less than the found by ICA$_{FSSP}$, for the all groups of instances. This indicates that the solution quality given by ImpICA$_{FSSP}$ is higher than the obtained ones by ICA$_{FSSP}$, suggesting the superiority of our improved ICA. This shows nicely how the effectiveness of adding an improving phase to the imperialistic countries contributes positively to the search process. Furthermore, we can notice that statistical significant differences exist between these options (p-values $< 4.2e^{-09}$). Considering the spent time to find the best solution, ImpICA$_{FSSP}$ is faster than ICA$_{FSSP}$ when the most complex instances are solved. Therefore, we can conclude that ImpICA$_{FSSP}$ produces more optimal schedules than the ICA$_{FSSP}$ and is more efficient when the complexity grows.

7 Conclusions

In this paper, we presented ICA$_{FSSP}$ and ImpICA$_{FSSP}$, two new metaheuristic approaches based in the Imperialist Competitive Algorithm for solving the flow-

Table 3: Best and average *gap* values and runtime to find the best solution for ImpICA$_{FSSP}$ and ICA$_{FSSP}$.

Instance	**ImpICA$_{FSSP}$**			**ICA$_{FSSP}$**		
	Gap_{Best}	Gap_{Mean}	$Time_{Best}$	Gap_{Best}	Gap_{Mean}	$Time_{Best}$
20×10	0.010	**0.041***	0.39	0.011	0.052	0.09
20×20	0.010	**0.035***	0.78	0.014	0.044	0.11
50×10	0.017	**0.054***	5.33	0.042	0.083	0.64
50×20	0.038	**0.069***	6.04	0.043	0.100	1.34
100×10	0.011	**0.037***	18.66	0.010	0.055	5.31
100×20	0.032	**0.067***	15.72	0.033	0.097	10.29
200×10	0.008	**0.027***	55.83	0.008	0.049	19.85
200×20	0.026	**0.068***	39.10	0.026	0.098	51.28
500×20	0.016	**0.054***	193.12	0.018	0.072	540.60
Average	0.019	**0.050**	37.29	0.023	0.072	69.95

shop sequencing problem. ICA$_{FSSP}$ is the result of adapting the original ICA to solve a permutation-based problem like FSSP, which uses crossover and mutation operators based on this codification to move the colonies toward the imperialistic countries. The results indicates that ICA$_{FSSP}$ with the combination of PMX and interchange operators obtains the best quality solutions.

Our second proposal is ImpICA$_{FSSP}$, an improved version of ICA$_{FSSP}$ considering the application of a local search procedure to the imperialistic countries in order to increase its power. This improvement allows to obtain better schedules than the found by ICA$_{FSSP}$ in all tested instances.

As further research lines, we would first attempt to fine-tune the algorithmic design with the aim of improving the current results for the FSSP. As a second research line, we plan to adapt this metaheuristic to solve related problems.

Acknowledgements

The authors are supported by the Universidad Nacional de La Pampa, and the ANPCYT under contract PICTO 2011-0278 and the Incentive Program from MINCyT. The second author is also supported by CONICET.

References

1. E. Atashpaz-Gargari and C. Lucas. Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition. In *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*, pages 4661–4667, Sept 2007.
2. J. E. Beasley. OR-Library: distributing test problems by electronic mail. *Journal of the Operational Research Society*, 41(11):1069–1072, 1990.
3. M. Behnam, R. Abbas, S. Alireza, and Mehdi E. Imperialist competitive algorithm for solving non-convex dynamic economic power dispatch. *Energy*, 44(1):228 – 240, 2012.
4. P. Chang, W. Huang, J. Wu, and T.C.E. Cheng. A block mining and recombination enhanced genetic algorithm for the permutation flowshop scheduling problem. *International Journal of Production Economics*, 141(1):45–55, 2013.

5. J. P. Fan and G. K. Winley. A heuristic search algorithm for flow-shop scheduling. *Informatica: journal of computing and informatics*, 32(4):453–464, 2008.
6. J. M. Framinan and R. Leisten. An efficient constructive heuristic for flowtime minimisation in permutation flowshops. *Omega*, 31(4):311317, 2003.
7. J. Gao, R. Chen, and W. Deng. An efficient tabu search algorithm for the distributed permutation flowshop scheduling problem. *International Journal of Production Research*, 51(3):641–651, 2013.
8. M. R. Garey, D. S. Johnson, and R. Sethi. The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research*, 1:117–129, 1976.
9. L. Huang, H. Duan, and Y. Wang. Hybrid bio-inspired lateral inhibition and imperialist competitive algorithm for complicated image matching. *Optik - International Journal for Light and Electron Optics*, 125(1):414 – 418, 2014.
10. S. M. Johnson. Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1:61–68, 1954.
11. C. Koulamas. A new constructive heuristic for the flowshop scheduling problem. *European Journal of Operational Research*, 105(1):66–71, 1998.
12. Z. A. Lomnicki. A "branch-and-bound" algorithm for the exact solution of the three-machine scheduling problem. *OR*, 16(1):pp. 89–100, 1965.
13. G. B. McMahon and P. G. Burton. Flowshop scheduling with the branch and bound method. *Operations Research*, 15(3):473481, 1967.
14. G. Mokhtari, A. J. Ghanizadeh, and Ebrahimi E. Application of imperialist competitive algorithm to solve constrained economic dispatch. *International Journal on Electrical Engineering and Informatics*, 4(4):553–562, 2012.
15. M. Nawaz, E. Enscore Jr, and I. Ham. A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Omega*, 11(1):91 – 95, 1983.
16. V. B. Nouri, P. Fattahi, and R. Ramezanian. Hybrid firefly-simulated annealing algorithm for the flow shop problem with learning effects and flexible maintenance activities. *International Journal of Production Research*, 51(12):3501–3515, 2013.
17. D. Palmer. Sequencing jobs through a multi-stage process in the minimum total time-A quick method of obtaining a near optimum. *Operational Research*, 16(1):101–107, 1965.
18. C.R. Reeves. A genetic algorithm for flowshop sequencing. *Computing Operation Research*, 22:5–13, 1995.
19. R. Ruiz, C. Maroto, and J. Alcaraz. Two new robust genetic algorithms for the flowshop scheduling problem. *Omega*, 34(5):461–476, 2006.
20. R. Ruiz and T. Stützle. A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. *European Journal of Operational Research*, 177(3):2033–2049, 2007.
21. E. F. Stafford. On the development of a mixed-integer linear programming model for the flowshop sequencing problem. *The Journal of the Operational Research Society*, 39(12):pp. 1163–1174, 1988.
22. S. M. A. Suliman. A two-phase heuristic approach to the permutation flow-shop scheduling problem. *International Journal of Production Economics*, 64(1-3):143–152, 2000.
23. X. Wang and L. Tang. A discrete particle swarm optimization algorithm with self-adaptive diversity control for the permutation flowshop problem with blocking. *Applied Soft Computing*, 12(2):652 – 662, 2012.

Integración de agentes inteligentes heterogéneos en el entorno T-World

Sergio Burdisso, Guillermo Aguirre, and Marcelo Errecalde

Universidad Nacional de San Luis, Departamento de Informática
Ejército de los Andes 950, San Luis.
sergio.burdisso@gmail.com,gaguirre@unsl.edu.ar,merreca@unsl.edu.ar

Resumen A fin de facilitar la tarea de crear y experimentar con agentes computacionales inteligentes se ha creado una plataforma que funciona como campo de experimentación portable. Cuenta con una simulación gráfica atractiva y numerosas cualidades que la hacen ideal para docencia. Uno de los desafíos más importantes es manejar la heterogeneidad de los agentes, lo que se consigue mediante un programa intermediario.

1. Introducción

Una ejercitación típica en los cursos de IA consiste en plantear distintos tipos de situaciones para las cuales los alumnos deben encontrar una solución, codificar esa solución en algún lenguaje de programación y finalmente evaluar cómo se desempeña el agente creado. Existen pocos escenarios controlados [3] donde los estudiantes puedan desarrollar este tipo de ejercitación. Las limitaciones provienen por las restricciones del lenguaje de programación empleado y por la escasez de variantes en los parámetros de configuración del entorno de prueba. Uno de los pocos recursos de este tipo que ha sido bastante reconocido es Tileworld [5], pero este mundo simulado presenta numerosas dificultades que han impedido su uso de manera masiva. Muchos de estos impedimentos se describen en un trabajo previo [2].

Actualmente se encuentra en funcionamiento una plataforma de experimentación denominada T-World (tworld-ai.com) que permite configurar escenarios similares a los de Tileworld, crear agentes para resolver los problemas planteados y verificar el funcionamiento de los mismos mediante una simulación gráfica. Para facilitar la accesibilidad y simplicidad de utilización, se optó por utilizar el navegador web como plataforma de implementación. Esto aumenta la portabilidad del sistema debido a que todos los dispositivos actuales tales como smartphones, tablets y computadoras personales disponen de un navegador web. Además permite que el sistema pueda ser accedido vía Internet (o localmente) y ejecutado directamente en el dispositivo del usuario sin necesidad de instalar o configurar software adicional.

El campo de pruebas construido es una extensión de Tileworld, consiste en una grilla bidimensional en la cual existen agentes, huecos, baldosas y obstáculos. El robot puede moverse en las cuatro direcciones del plano y si está junto a una

baldosa la puede empujar. Los obstáculos no pueden ser atravesados por los robots y los huecos deben ser tapados con baldosas, el objetivo es llenar tantos huecos como sea posible. Este entorno es dinámico ya que los huecos aparecen y desaparecen aleatoriamente lo que representa todo un desafío a la hora de implementar algoritmos que resuelvan las pruebas.

La interfaz gráfica de usuario cuenta con una gran colección de elementos visuales tales como menús, íconos, botones e imágenes junto al entorno en tres dimensiones para las simulaciones. Desde el comienzo del diseño del T-World quedó claro que la comunicación con los agentes tendría que ser bidireccional, ya que tanto el ambiente como los agentes pueden recibir respectivamente acciones y percepciones, sin seguir ningún patrón preestablecido, e.d. ambos pueden hacerlo en cualquier momento y en cualquier orden, incluso en simultáneo. Sin embargo, este requerimiento se contrapone a la idea original con la cual fue concebida la Web, donde el usuario especifica una URL para recuperar un recurso y como respuesta el servidor entrega una instancia del mismo —por ejemplo, un documento HTML, un archivo de imagen o de cualquir otro tipo. Afortunadamente, con la reciente aparición de WebSocket de HTML5, ahora sí es posible tener un modelo de comunicación full-duplex para el navegador web. El protocolo *WebSocket* proporciona canales de comunicación *full-duplex* sobre una única conexión TCP.

En este artículo se describe un componente que es clave para permitir que en T-World se ejecuten agentes codificados en diferentes lenguajes de programación, este componente es denominado *proxy*. Mediante el proxy se vinculan, de una manera transparente, los robots que corren en la simulación con los agentes creados por los alumnos. Las percepciones del entorno y las acciones sobre el mismo son realizadas por los agentes respetando las características particulares del lenguaje de programación escogido. Es el proxy quien se encarga de hacer la traducción para cumplir con el formato de protocolo requerido para interactuar con T-World.

2. Entorno T-World

El diseño e implementación de *T-World* se hizo para tener una versión de Tileworld que los estudiantes pudieran usar inmediatamente. Desde el plan inicial se pretendió que fuese *portable* y *flexible*, y que permitiera incorporar razonamiento a los robots mediante *programas* escritos en cualquier lenguaje. También se consideró la aplicación de diversos parámetros al mismo tiempo y que la simulación fuese *visual*, para lograr experiencias educativas atractivas.
En la Figura 1 se observa que la experiencia es similar a un experimento real en el contexto de un juego, ya que por ejemplo se puede ver el efecto del robot al empujar las baldosas. También se puede observar la reacción del robot cuando choca con un obstáculo.

T-World además de incorporar un editor de código para programar los robots, brinda una API con facilidades para desarrollarlos.
El diseño de la plataforma se basó en las ideas y conceptos presentados en el libro

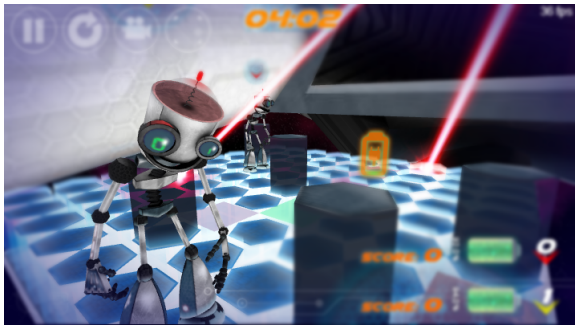


Figura 1: Captura de pantalla de T-World

Artificial Intelligence: A modern approach [6], en esta sección se abordan algunos de esos fundamentos con la intención de explicar cómo se relacionan entre sí y cómo fueron tenidos en cuenta durante la implementación de T-World.

2.1. Descripción del agente

Un agente es todo aquello que se pueda ver como una entidad que percibe su ambiente a través de *sensores* y actúa sobre ese ambiente a través de *actuadores* [6]. Ver figura 2. El término percepción se refiere a la entrada recibida por el agente en un determinado momento. Una secuencia de percepciones de un agente es la historia completa de todo lo que el agente ha percibido. En general la elección de la acción que hace el agente en un instante dado, depende sólo de la secuencia de percepciones observadas y no de lo que no ha percibido.
En el contexto de T-World los agentes se corresponden con los *robots* de la simulación 3D, ver figura 1.

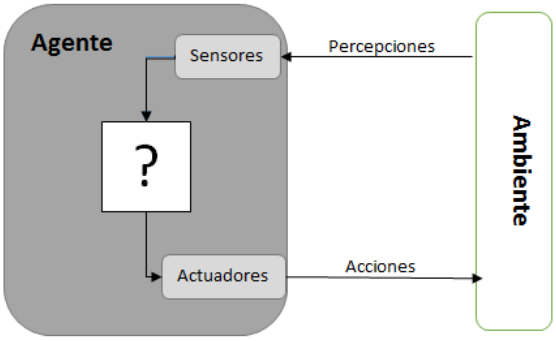


Figura 2: Interacción con el ambiente a través de sensores y actuadores

2.2. La percepción

Los agentes perciben el ambiente en el que se encuentran por medio de *sensores*, un agente humano percibe mediante sus ojos, sus oídos, etc., un robot lo hace por medio de cámaras y sensores infrarrojo. En T-World un programa agente recibe el estado del ambiente cuando es creado y posteriormente por requerimientos explícitos invocando a la función *$perceive()*. Aquí, una percepción brinda la información que necesita el agente para tomar sus decisiones, por ejemplo, su próxima acción. En T-World la percepción que es enviada hacia el agente está compuesta por dos partes: el *encabezado* y los *datos* propiamente dichos. En el encabezado, el ambiente especifica si se trata de un objeto con información sobre el estado del ambiente o si se trata de un evento especial como un mensaje o la finalización de la simulación. La parte de datos se distribuye en tres compartimentos según el origen que tenga: Información *externa* al agente (e.g. disposición de elementos físicos y de otros agentes en el mundo, tiempo transcurrido, etc.). Información *relativa al agente* receptor de la percepción (niveles de energía, puntuación, posición actual, etc). Y el tercer tipo de información corresponde al *conocimiento incorporado*, son los valores constantes establecidos por el usuario para ese ambiente en particular. Por ejemplo dimensiones del escenario, los valores y las reglas que definen cómo evoluciona el ambiente.

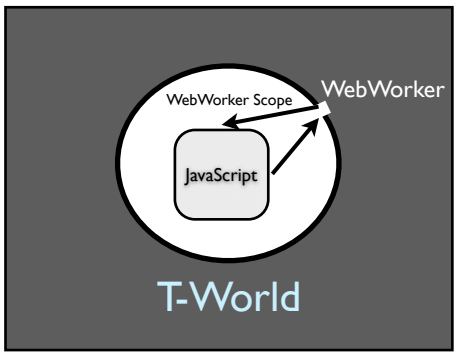
2.3. Las acciones

La descripción del comportamiento de un agente se completa con la especificación de sus acciones. Los agentes actúan sobre el ambiente por medio de sus *efectores*, lo que para un agente humano serían sus manos, piernas y cuerdas vocales, para un agente de software pueden ser, hacerse visible en la pantalla, escribir en un archivo y enviar un paquete por la red. Los agentes que están inmersos en T-World actúan por medio de las siguientes acciones: *desplazamiento* en la grilla, *restauración* de la batería y *envío* de mensajes a otros agentes. El desplazamiento consiste en actualizar la ubicación del agente a una nueva fila o columna adyacente a la actual, siempre que sea posible hacerlo. La restauración de la batería se puede hacer cuando el agente se ha quedado sin carga; antes de realizar esta acción el agente debe considerar su costo, ya que a cambio de la energía deberá ceder la mitad de su puntaje actual. Un agente puede enviar un mensaje a un agente específico o difundirlo a todos los integrantes del equipo. El contenido de un mensaje es una secuencia de caracteres que idealmente deberá estar convenientemente estructurada, por ejemplo, siguiendo los estándares de comunicación entre agentes como la establecida por FIPA [1]. Existe una acción especial que es hacer *nada*, empleada por ejemplo para indicarle a T-World que el agente necesita más tiempo para decidir la próxima acción o cuando no es posible realizar ninguna otra acción.

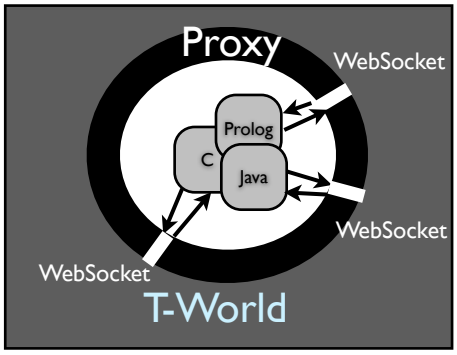
2.4. Programas Agente

El comportamiento de un agente se describe por una *función agente*, que a cada secuencia de percepciones le hace corresponder una acción. Se podría

pensar en crear una tabla equivalente a la función agente para describir cualquier agente. Para eso sería necesario tabular todas las secuencias posibles de percepciones y para cada una de ellas registrar la acción escogida por el agente. La tabla es claramente una caracterización externa del agente. Internamente, la función agente será implementada por un *programa agente*. El programa agente se corresponde con el signo de interrogación en la figura 2.
En T-World, dependiendo del lenguaje usado para codificar el programa agente se consideran dos situaciones diferentes: cuando se usa JavaScript y cuando se usa algún otro lenguaje que no sea JavaScript, como ser Java, C o Prolog.
En T-World la creación de un programa agente consiste de los siguientes pasos:
1) Asignar un nombre al programa agente, esto es necesario entre otras cosas, para que el usuario identifique a su programa agente dentro del sistema.
2) Determinar cómo se controlará el robot, las posibilidades son que sea el usuario humano mediante el teclado o algún algoritmo escrito en un lenguaje de programación. El T-World procede de manera distinta cuando el lenguaje usado para el agente es JavaScript o cuando es algún otro lenguaje, este último caso se describe en la próxima subsección.
3) Ingresar una descripción del programa agente, útil para ayudar a reconocer las características del programa agente como el algoritmo escogido para razonar.



3 a) Lenguaje JavaScript



3 b) Distintos lenguajes

Si se ha elegido escribir el programa agente en JavaScript, en el segundo paso se abre automáticamente un editor con un esqueleto básico para comenzar a programar. Dentro del editor se presentan cuatro secciones de código: la primera, "Agent Program", es para el algoritmo principal del programa agente. La segunda, "Global Scope", se usa para definir funciones y variables de alcance global. La tercera sección, "Start Event", es para manejar el evento de "inicio de simulación" que se dispara al comenzar la simulación. Esto es útil para inicializar variables o estructuras de datos que dependan de la primer percepción y que luego usará el agente. Generalmente se usa para obtener la configuración inicial

del ambiente: condiciones de finalización, tamaño de la grilla, etc. La última sección, "Message Received Event", se debe implementar solamente cuando el agente requiere recibir mensajes, o sea, cuando integra un sistema multi-agente.

Después de crear el programa agente ya puede comenzar a interactuar con T-World. Se debe asociar una configuración del ambiente de tarea con el programa agente creado, esto se puede hacer directamente desde el ambiente del editor.

El programa agente se ejecuta como un Web worker (ver figura 3a), es decir como un script en el background independiente del script principal que atiende la simulación. La invocación al constructor Worker() crea un worker y retorna un objeto Worker que representa al programa agente; mediante el objeto creado, T-world se comunica con el programa agente.

Uso de un lenguaje distinto a JavaScript La diferencia está en el segundo paso de la creación del programa, ya que no se edita el programa dentro de T-World como se hizo en el caso de JavaScript. Como el usuario utilizará un lenguaje de programación externo al navegador web, se le debe indicar a T-World cómo enlazar el programa del usuario con su correspondiente programa agente. Para establecer el enlace, se ingresa la dirección IP y el puerto donde correrá el Proxy, y finalmente, un string que identifique al programa agente [1]. Este string, llamado *Magic String*, es utilizado por el Servidor Proxy para vincular el programa agente del usuario con su contraparte en T-World. En la próxima sección se explica con más detalle el uso del *Magic String*.
Notar que los programas agente escritos en lenguajes externos al navegador, tales como C o Java, presentan una diferencia significativa con los desarrollados en JavaScript en la manera en que se comunican con T-World, estos requieren del Servidor Proxy para comunicarse con T-World (ver figura 3b), por lo que finalmente el usuario deberá conectarse al proxy desde su programa agente, mediante un socket para recibir la percepción y devolver la respectiva acción a realizar por el agente. De este modo, el cuerpo del programa agente típicamente va a corresponderse con el siguiente algoritmo:

0) Crear socket
1) Conectar socket a la dirección IP y puerto del proxy
2) Enviar cadena de conexión
3) Recibir percepción
4) Según la percepción determinar la próxima acción.
5) Envia la acción
6) Si Percepción.Header distinto de END saltar a 3

3. El servidor Proxy de T-World

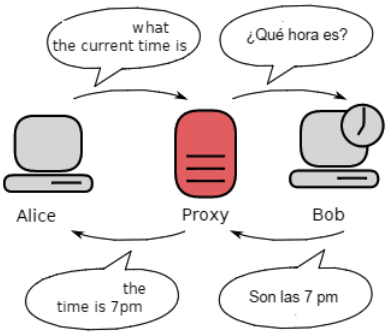
En esta sección se describen con más detalle dos aspectos importantes de la plataforma que, si bien se mencionaron antes, requieren una explicación más

[1] por defecto se le asigna el valor del nombre del programa agente

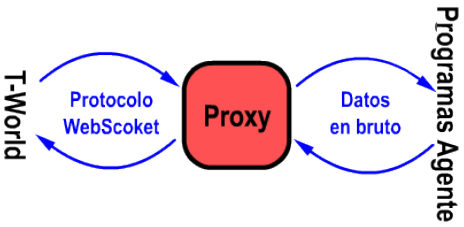
detallada: (a) el protocolo WebSocket que permite una comunicación bidireccional entre los programas agente con su correspondiente robot y (b) el formato estándar elegido para las percepciones: JSON (o XML), capaz de ser interpretado por programas escritos en distintos lenguajes.

Si bien tanto el protocolo como el formato elegidos son muy conocidos, requieren que el estudiante deba aprender a implementar WebSocket para comunicar el programa agente con el ambiente y conocer JSON para interpretar lo que sucede en el entorno. Esto complica la tarea del alumno al incorporar su programa agente, recibir las percepciones, y enviar las acciones a T-World. Esa complicación se solucionó con la creación de un *proxy*.

Un *servidor proxy* es un programa que rompe la conexión directa entre un emisor y un receptor, funcionando como un intermediario. Ambos, emisor y receptor piensan que se están comunicando entre sí, pero en realidad, lo hacen sólo con el proxy. Por eso su nombre, ya que proxy en inglés significa "actuar en nombre de otro". Este concepto se ilustra en la Figura 4a, en donde se ve



4 a)Idea general de un proxy



4 b)Proxy de T-World

la comunicación entre dos computadoras (Alice y Bob) conectadas a través de una tercera computadora que actúa como un Proxy. Notar que Bob desconoce que Alice hable Inglés y Alice que Bob habla Español, pero aún así pueden comunicarse normalmente ya que el Proxy se encarga de hacerlo transparente para ellos.

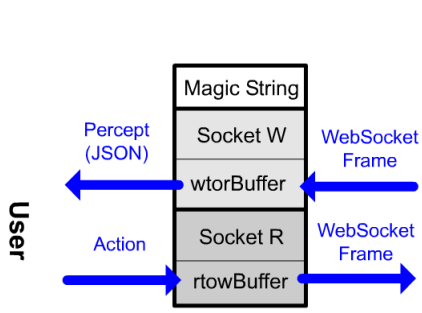
El proxy de la plataforma sigue el esquema planteado en la Figura 4a, donde Bob se corresponde con el programa agente, Alice con T-World, el idioma Inglés con WebSocket y el Español con las percepciones y las acciones.

El proxy se encarga de enlazar cada agente (robot) de la simulación 3D con su respectivo programa agente. Dado que la simulación transcurre dentro del navegador web, T-World asigna un WebSocket a cada robot y conecta cada WebSocket a su respectivo proxy. Por eso el diálogo entre T-World y el proxy sigue el protocolo WebSocket. A su vez, el programa del usuario sólo debe conectarse al proxy, recibir las percepciones JSON desde T-World y enviar las correspondientes acciones. Este proceso se ilustra en la figura 4b. El servidor

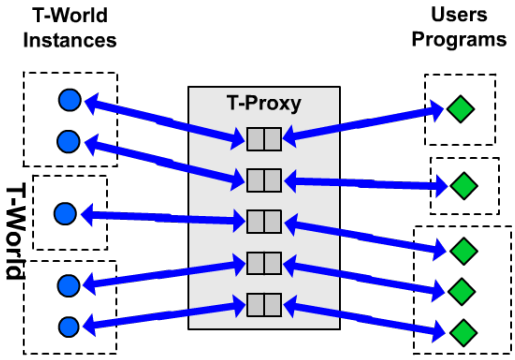
proxy ha sido codificado en lenguaje C, haciendo uso sólo de socket estándares (POSIX). Esto asegura la portabilidad a todas las plataformas y sistemas operativos. Para poder realizar su labor, el proxy dispone de un socket principal cuya tarea es la de escuchar nuevos pedidos de conexión, cada vez que uno de éstos arriba al servidor, se crea una nueva conexión. Esto último se lleva a cabo creando un nuevo socket por cada nueva conexión. Cuando uno de estos sockets reciben el mensaje correspondiente al handshake del protocolo WebSocket (ver sección 1.3 Opening Handshake de [4]), el proxy identifica a este socket como un WebSocket proveniente de T-World. Luego ambas partes, tanto los programas de usuario como los WebSockets envían un mensaje especial, llamado CONNECT, para que el proxy pueda identificarlos y aparearlos de forma apropiada. El mensaje CONNECT se forma por la cadena "CONNECT:" concatenada con la Magic String que el usuario le asignó al programa agente durante la creación del mismo en T-World. La Magic String es una cadena dada por el usuario que sirve como el identificador del programa agente, de forma tal que éste se enlace sólo con el robot correcto –aquel que tenga asignado la misma Magic String que su programa agente. Este Magic String se utiliza para armar pares de conexiones (WebSocket, Socket) bajo un mismo valor de Magic String. Estos pares se arman de la siguiente manera: cuando un mensaje CONNECT arriba desde un socket, a este socket se le asigna una etiqueta con el valor de la Magic String recibida. Posteriormente cuando un WebSocket envía el mensaje CONNECT se busca un socket libre[2] con la misma Magic String, de encontrarse uno se arma el par (WebSocket, Socket) con ellos. Este proceso también puede ocurrir a la inversa, que el WebScoket sea el primero en enviar el mensaje CONNECT y que posteriormente un socket lo haga. En este caso se busca un socket libre para este WebSocket. Para poder llevar a cabo el proceso antes descrito, el proxy genera una *estructura* por cada par de conexión. Esta estructura se ilustra en la figura 5a. Como se observa en la figura, esta estructura tiene un campo para almacenar la Magic String bajo la cual están apareados los sockets. También hay dos campos para guardar el socket de usuario (Socket R) y el WebSocket (Socket W) del par[3]. Los campos wtorBuffer y rtowBuffer son buffers de 16KB utilizados para el reenvío de mensajes desde el WebSocket hacia el usuario y viceversa, proceso que se describe en las siguientes secciones.

La estructura descripta le brinda al proxy la capacidad de soportar múltiples pares de conexiones, todos conectados de manera concurrente al proxy. Como se observa en la figura 5b, cada estructura se encarga de mantener un par de conexión entre múltiples instancias de T-World y múltiples instancias de programas de usuarios. Los círculos representan WebSockets, y el cuadrado donde éstos se encuentran instancias de T-World. Los rombos representan sockets de usuarios, cada cuadrado contenedor representa una instancia de programa de usuario.

[2] por libre se refiere a que no pertenezca ya a un par (WebSocket, Socket)
[3] Socket R y Socket W guardan el descriptor de archivo de cada socket



5 a) Interacción Proxy-T-World



5 b) Interacción Agente-Proxy

3.1. Interacción entre Proxy y T-World

Como se mencionó anteriormente la comunicación entre el proxy y T-World se hace por medio del protocolo WebSocket. Esto implicó implementar manualmente dicho protocolo durante el desarrollo del proxy, siguiendo las especificaciones detalladas en su respectivo RFC [4]. Los mensajes que se envían utilizando el protocolo WebSocket tienen un formato específico el cual se denomina como "WebSocket Frame" o en este contexto simplemente Frame. Los datos que viajan dentro de Frames que se envían desde T-World hacia el proxy lo hacen enmascarados –esto es uno de los requerimientos del protocolo. Por esta razón cuando se recibe un Frame desde T-World el proxy debe desenmascarar el dato enviado (la percepción) y almacenarlo en el buffer wtorBuffer para ser posteriormente reenviado hacia su correspondiente programa agente (ver figura 5a). Posteriormente cuando el programa de usuario responde con la acción a realizar por el robot, esta se guarda en el buffer rtowBuffer. El proxy obtiene dicha acción desde este buffer, la encapsula dentro de un WebSocket Frame y finalmente envía este Frame al WebSocket correspondiente dentro de T-World.

3.2. Interacción entre programa agente y Proxy

La interacción en este caso es mucho más simple, ya que no se realiza utilizando ningún protocolo en particular. La percepción almacenada en el buffer wtorBuffer se envía hacia el programa agente sin modificarla. Esto es, la secuencia de caracteres que se encuentran en el buffer se envían hacia el programa agente. Como esta secuencia de caracteres tiene un formato especial, JSON, puede ser interpretada por el programa agente receptor. Una vez interpretada, el programa agente típicamente realizará algún cómputo sobre la misma y posteriormente devolverá una acción. Las acciones son también secuencias de caracteres. Una vez recibida la acción, el proxy la encapsula dentro de un WebSocket Frame y escribe los bytes del Frame en el buffer rtowBuffer. Acto seguido este Frame se envía hacia el correspondiente WebSocket –esto provoca que el robot correspondiente actúe dentro la simulación 3D.

4. Conclusiones y Consideraciones finales

Se describe un entorno para evaluar el desempeño de agentes y particularmente se detalla la implementación del servidor proxy creado especialmente para integrar agentes escritos en diferentes lenguajes. En procura de crear un entorno portable, se eligió como lenguaje de implementación JavaScript y así poder ejecutar en el ámbito de un navegador web. Los usuarios que también elijan Java Script para programar sus agentes, tendrán una API y una librería de agentes a su disposición. Otra cualidad destacada es la flexibilidad porque los agentes se pueden programar en distintos lenguajes como C, Java o Prolog. Mediante un servidor Proxy, distintos agentes heterogéneos se integran a T-World. El proxy usa WebSocket para hablar con el robot de T-World y transmitir información en formato JSON a los programas agente que controlan al robot. La conexión entre el robot y su controlador se mantiene en el proxy mediante una estructura de datos que contiene dos buffers para la comunicación bidireccional.

Los objetivos que se plantearon desde el inicio del diseño de T-World (tworld-ai.com), fueron variados, entre otros que fuera accesible desde cualquier parte del mundo, de manera que simplemente ingresando a un sitio web se tenga acceso a una herramienta completa e integrada de forma nativa al navegador web. Que su funcionamiento fuese independiente de un sistema operativo en particular o de cualquier otra pieza de software –como intérpretes, runtime frameworks o máquinas virtuales. Para lograr todos estos objetivos, entre otras cosas, fue requerido codificar más de 18 mil líneas de código –12 mil en JavaScript, aproximadamente mil en Lenguaje C y el resto en HTML y PHP. También fue nacesaria la utilización de diversas tecnologías tales como AngularJS de Google como framework para la aplicación web y la librería CopperLitch como una abstracción de WebGL para la generación de gráficos 3D.

Referencias

1. Fabio Luigi Bellifemine, Giovanni Caire, and Dominic Greenwood. *Developing Multi-Agent Systems with JADE*. John Wiley & Sons, NJ, April 2007.
2. Sergio Burdisso, Guillermo Aguirre, and Marcelo Errecalde. T-world: un entorno gráfico, flexible y portable para la enseñanza e investigación de agentes inteligentes. In *CACiC 2014*. Universidad Nacional de La Matanza, 2014.
3. S. Hanks, M. Pollack, and P. Cohen. Benchmarks, test beds, controlled experimentation and the design of agent architectures. *AIMagazine*, 14(4):17–42, 1993.
4. I. Fette and A. Melnikov. The WebSocket Protocol. Technical report, RFC Editor, December 2011.
5. Martha E. Pollack and Marc Ringuette. Introducing the tileworld: Experimentally evaluating agent architectures. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 183–189, 1990.
6. Stuart J. Russell and Peter Norvig. *Artificial Intelligence - A Modern Approach (3. internat. ed.)*. Pearson Education, 2010.

A Desiderata for Modeling and Reasoning with Social Knowledge

Fabio R. Gallo[1,3], Natalia Abad Santos[2],
Gerardo I. Simari[1,3], and Marcelo A. Falappa[1,3]
{fabio.gallo,gis,mfalappa}@cs.uns.edu.ar,nasantos@uns.edu.ar

[1] Institute for Computer Science and Engineering UNS-CONICET, Argentina
[2] Dept. of Mathematics, Universidad Nacional del Sur, Bahía Blanca, Argentina
[3] Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina

Abstract. The ongoing surge in the amount of users that engage in online activities, as well as the expansion of the type of such activities, has recently made it clear that there is a widening gap between current knowledge representation and reasoning tools and the type of knowledge that is essentially up for grabs for whoever is willing (and has the tools) to extract it from social media sites. In this position paper, we propose the concept of Social Knowledge Base (SKB, for short) as an extension of traditional KBs with representation and reasoning capabilities that arise from the singular combination of characteristics that define this setting: (i) ontological knowledge, (ii) user preferences, (iii) reasoning under uncertainty, (iv) stream reasoning, and (v) representation of complex social networks. We propose a list of desirable properties—a desiderata—that next-generation KR formalisms for modeling and reasoning with SKBs should enjoy. The treatment is non-technical, focusing on building a road map of the formidable list of problems that must be solved in this complex setting rather than proposing a concrete solution, which would be impossible in a single article. We conclude by proposing some first steps towards achieving this goal.

Keywords: Social Web, Complex Networks, Reasoning Under Uncertainty, Preferences, Ontology Languages

1 Introduction and Motivation

Recent times have seen a veritable explosion in the amount and kind of information that is available to anyone with a connection to the Internet. This explosion has its roots in the so-called World Wide Web [1], which revolutionized internet applications by allowing users to link resources with one another and easily organize the material that they wish to publish. The second revolution came with the advent of Web applications in which users produced their own material, such as in blogs or forums where users share information ranging from plain text to photos, videos, and audio—this "new version" is often referred to as *Web 2.0* to highlight that a step was taken since the implementation of the original idea.

Finally, in the last few years, the Web has once again taken an evolutionary step: in its current form, which many refer to as *Social Web* (or the *Web 3.0*), users and the relationships among them are the central participants. Another revolutionary aspect that appeared in Web 3.0 is that data is now also produced automatically by computers; examples of this are data output by the host of sensors now carried by most smart phones, or by the smart homes that are slowly becoming more and more present.

Unfortunately, research in Knowledge Representation and Reasoning formalisms has lagged behind this rapid evolution in how data is created and disseminated. The goal of this position paper is thus to explore a desiderata—a list of desirable characteristics—for the development of what we will call *social knowledge bases* (SKBs, for short). The idea behind this line of research is to derive a framework and methodology akin to Ontology Based Data Access (OBDA) [23] that is specialized for the unique social aspects discussed above. Our work is influenced by recent proposals in the complex networks literature [21, 20], which also establishes a set of criteria that is desirable for modeling cascades, a specific phenomenon that—as we will see—also plays an important role in our setting. Our desiderata are therefore inspired in this work, but necessarily go above and beyond their scope given the greater generality of the problems that need to be solved.

We now describe two settings that we will use as running examples to motivate our discussion. The first setting is an online matchmaking service.

Example 1 (Friendship/Dating site). Consider a web site where people register and complete a profile with the objective of meeting new people—the goal might be to establish a romantic relationship or simply make new friends. As a way to simplify the creation of profiles, the site offers the option to log in with the users' favorite social media site (like Facebook, Twitter, Google Plus, etc.), and optionally also link the profile with multiple such sites.

The main aim of the site is to match people who have compatible personalities; in order to have tools that can be leveraged to solve this (very difficult) problem, the site allows users to explicitly specify their preferences in different domains, such as music, literature, movies, and even relationships—these inputs are complemented by the information that is extracted (with the users' permission) from their linked social media profiles. ■

The second example setting is a comprehensive trip organization service.

Example 2 (Travel site). As a second example, suppose we have a web site (similar to TripAdvisor[1]) that is designed to help people choose a place to spend their next vacation; it includes information on destinations, transportation, hotels, tours, restaurants, best season to go, etc. Members publish reviews, including numeric scores for several different categories as well as free text where they can go into detail regarding their experience. There are rich social features available, such as tagging in posts or reviews, suggestions, and private messages.

[1] `http://www.tripadvisor.com/`

As before, we assume that users are able to sign in with their social media accounts, which gives the system the possibility to extract relevant information—for instance, to suggest a destination, the system might use the fact that a user participates in Facebook groups for learning the German language to infer that they probably would like to travel to Germany. ■

In the following, we describe a list of desiderata to achieve the goal of designing a formalism to model and reason with social knowledge bases. As we will argue below, the problem essentially requires the combination of knowledge representation machinery for areas that have up to now largely been considered in relative isolation: (i) ontology languages, (ii) preference models, (iii) reasoning under uncertainty, (iv) stream reasoning, and (v) complex social networks.

2 Desiderata for Building and Querying Social KBs

In the previous section, we argued that it is necessary to develop novel KR tools to reason with social data; we will now offer further support for this argument by proposing a series of characteristics and capabilities that SKBs should have—developing such a desiderata has the additional value of acting as a road map for guiding future research efforts in this direction. For further discussion of literature related to each point, see Section 4,

(1) Model complex networks. In social knowledge bases, it should be possible for entities to be of different types: people, products, companies, books, movies, etc. Furthermore, it should be possible for there to be different kinds of relationships among them. It is thus necessary to be able to represent networks with different kinds of nodes, as well as multiple attributes and relationships for each one—such models are often referred to as *complex networks* [2].

Consider the setting from Example 1; in this case, it is clear that it would be useful for connections between users to contain additional information about the relationship they have. For instance, kind of tie (relative, classmate, work partner, etc.), how long they have known each other, how many social media sites they frequent, etc. Another important observation is that connections do not always need to be symmetric—in the dating example, person A can consider person B to be a good match, but B may not agree. Having rich information about entities and how they are related can thus be useful to improve users' experiences.

(2) Model atomic actions. A specific set of actions (by agents or exogenous factors) that can occur in the domain need to be identified.

Considering social media sites like Facebook or Google Plus, such actions could include posting, commenting, liking/+1 a post, friending/unfriending, messaging, etc. It is these actions that will be the building blocks for inferences about preferences or regarding reasoning under uncertainty, as we will discuss below. Also in connection with a point discussed in the following is the fact that an adequate selection of atomic actions to be modeled will have an impact on computational tractability. User data in social media suffers constant

change, and a regular user could produce a large amount of data per day; depending on the way in which the SKB will be used, it may not be necessary to incorporate all of this data into the model. For instance, regarding the setting in Example 2, it may be a good idea to incorporate comments as actions since users may give information regarding preferences in their mode of travel (for example, that they are afraid to fly); on the other hand, this might be less relevant for Example 1. Hence, it is essential to characterize and prioritize atomic actions so that resources are not wasted by processing and storing unnecessary data.

(3) Model quantitative and qualitative preferences. Quantitative preferences are often useful when automatically learning from data, or in simple domains; on the other hand, qualitative preferences (defining strict partial orders) are often more naturally elicited from human beings but more difficult to extract automatically.

To illustrate this point, consider the travel setting from Example 2. Quantitative preferences could be obtained from users' explicit rankings of favorite cities, countries, museums, beaches, etc. On the other hand, reviews or polls could also provide less structured preferences, such as the fact that the user prefers beach destinations to mountain ones, or that hotels near the city center are preferred over those that are not.

(4) Reason about groups. Social knowledge is inherently related to groups of entities (where entities are not necessarily all people); groups sometimes function as higher-level entities with their own preferences, relationships, etc.

There are many ways in which groups can be important when leveraging social knowledge. In Example 1, a group may be defined with respect to people's age group and interests, and the general preferences of such groups can be used in order to supplement the preferences of the individual. On the other hand, in Example 2 one can take the users' closest friends as a source of suggestions for travel destinations or activities—in this case, the group of friends is used as the basis of a kind of crowdsourcing. Challenges in this respect involve identifying the best possible composition of groups (for instance, determining who the users' closest friends are by considering how long they have known each other, share interests, etc.), and what to do about group members with conflicting preferences. The latter has been recently addressed in [16].

(5) Reason about cascading processes. One of the main characteristics of social networks is that information "flows" through them—this kind of dynamic is often referred to as a "cascading process" [11].

A clear example of this kind of process can occur in the travel domain (Example 2), where a user might travel to a new destination and post a series of pictures with very positive comments about their experience. This might cause several of the user's connections to "like" that destination and even plan trips there—the process can of course continue, with the new converts' activities causing some of their connections to do the same. It is thus important to model how influence propagates; there is extensive work in this area, and the logic programming proposal of [20] is perhaps the closest in spirit to the general approach that is required for SKBs.

(6) Flexible characterization of consistency/inconsistency. Classical conceptions of consistency are not adequate for modeling the kind of information that occurs in social settings—a more flexible approach is required for handling conflicts.

In our example settings, simple inconsistency cases might occur, for instance, when users have accounts in several social media sites but focus more on one than the others. Since data is usually not shared between accounts, it can occur that a user who lives in city C_1 later moves to city C_2 and only updates their profile for one of the accounts. An SKB taking information from these profiles would thus encounter an inconsistency. A more challenging case of inconsistency, much more difficult to characterize, is the case of a user of the system in Example 2 who strongly prefers beach destinations but suddenly starts paying attention to mountain-related places and activities (such as with +1s, posts, comments). The classical way to deal with the above situations is to try to modify the information contained in the knowledge base as little as possible in order to reach a consistent state without losing unnecessary information [7]; this is closely related to the following point.

(7) Social network-based belief revision operators. In close connection to the previous point, belief revision operations need to be applied in response to different kinds of events that signal changes in the SKB. The difference with respect to the classical setting is in relation to other points on this list—in particular, consistency, cascades, and uncertainty.

Among these, the relationship between cascades and belief revision operators is, to the best of our knowledge, never been studied. As an example, consider our travel setting and suppose an influential individual changes their opinion with respect to a certain destination (for instance, they start to express negative opinions about it and "unlike" the relevant pages), causing others to follow suit; how should this cascading belief revision process evolve?

(8) Reason about uncertainty. Conflicting information and inherently uncertain data makes it necessary to have an explicit representation of uncertainty.

There are many examples of the need to reason with uncertain knowledge. In our example dating application, some user information is private, and so cannot be directly used and perhaps only approximations can be obtained. For instance, user location can be approximated by content-based methods leveraging features of posts, such as mentions of place names and use of local dialect—since these are prone to error, a measure of probability must be assigned that depends on the kind and amount of information that supports each inference. Approaches to reasoning with ontological knowledge and user preferences have recently been proposed in [17].

(9) Rich query answering. Social knowledge is rich, and access to such knowledge often requires queries that combine the basic relational database-style queries with the graph-based queries often used in linked data [4].

Consider the travel setting from Example 2; queries to an SKB in this case might involve complex requests such as "*hotels with free wi-fi connection that have been positively reviewed by people who share my views and that at least one*

connection recommends, in order of preference". This involves reasoning under uncertainty (it is not always possible to determine if free wi-fi is available), reasoning about groups, and network structure, and preferences. Formalizing novel types of queries for SKBs, and obtaining effective algorithms to answer them, is therefore one of the main challenges ahead. Recent work [10] that can be leveraged towards this goal has proposed efficient algorithms for social networks under uncertainty.

(10) Time and space constraints: scalability and stream reasoning. Successful SKB formalisms must be able to cope with very large knowledge bases that are updated often with information that must be processed on the fly (or nearly so).

Micro-blogging is a clear example of how often new data is created: Twitter has about 100M active users who post over 230M tweets a day [3]. Processing such a high volume of data—much of which may not even be valuable [13] and that has a short life span—is a formidable challenge. An even greater challenge is to make the tools and processes that we propose in the previous points work adequately in such a setting. Considering the travel application from Example 2, a site with many active users must deal with a large volume of new comments, reviews, multimedia posts, and connections between users; an SKB that models even a portion of this activity must therefore be able to keep up with updates that, as we have seen, involve complex reasoning tasks.

3 Outlining a Framework for Social Knowledge Bases

Using the list of features discussed in Section 2 as a guide, we now briefly outline what a framework that integrates all of them might look like. A social knowledge base can be modeled as a 5-tuple of the form $SKB = (O, N, P, M, B)$, where:

- O is an *ontology* modeling the general knowledge about the domain. For instance, in the travel domain O would contain the database of hotels, flight routes, etc., as well as intensional knowledge such as *hostels are a kind of lodging*, or *wi-fi is a kind of internet connection*. This component could be modeled with the Datalog+/– family of ontology languages [5], which contains many different fragments focused on tractable query answering that generalize other well-known ontology languages such as the DL-Lite family of description logics.
- N is a model of the underlying *social network structure*. Since this is a kind of ontological knowledge, it could also be modeled using Datalog+/–; however, we propose to model them as separate components so that other approaches that are more specific can be used, such as the MANCaLog language [20].
- P is a *preference model* over the consequences of ontology O. This kind of integration has already been proposed in [14] and later extended to preferences under uncertainty [17] and preferences over groups [16].

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
O – Ontology	×	×	×	×	×	×	×	×	×	×
N – Network	×			×	×	×	×	×	×	×
P – Preferences			×	×		×		×	×	×
M – Probabilistic model	×				×	×		×	×	×
B – BR operators		×					×	×		×

Fig. 1. Assuming an SKB of the form (O, N, P, M, B), this table shows an example of the involvement of each component in satisfying the desiderata from Section 2. Different application settings may require different setups.

- M is a *probabilistic model* for ontology O. There are different ways in which probabilistic uncertainty can be integrated into ontological knowledge. For instance, in [8] annotations are added to both extensional and intensional knowledge, and the probabilistic model provides a probabilistic distribution over the annotations—this is an elegant way to allow for a separation of interests between the two models. Of course, other possibilities may be more appropriate depending on the domain of application.
- B is a set of *belief revision operators*. As was motivated before, revision operators that are informed by all the other components are needed in order to modify the knowledge base when new information needs to be incorporated. One approach in the logic-based probabilistic belief revision literature is the recent work of [22], which studies quantitative approaches to belief revision in a probabilistic structured argumentation language.

These components are coupled differently depending on the modeling or reasoning task that they are required to perform—the table in Figure 1 shows how each component might be typically involved in addressing each desideratum proposed above. For instance, desideratum D6 regarding consistency might involve components O, N, P, and M, since determining consistency may require ontological knowledge, access to network connections, user preferences, and probabilities. Of course, different applications may require different setups; returning to our example, in this case perhaps social connections are not considered for assessing consistency.

4 Discussion: Related Work and Challenges

Extracting, representing, and reasoning about the kind of information described above is a complex problem; although the examples may look simple, many issues arise when trying to combine all available data. There are some recent developments in the literature on ontological languages that are related to our present efforts in that they have already begun to investigate how some subsets of these areas can be adequately combined. The Datalog+/– family of ontology languages [5] has recently received a lot of attention given its flexibility and variety of available fragments that ensure tractable query answering. In [14],

the authors explore an extension of Datalog+/– with preference models that allows to rank the answers to queries with respect to users' preferences; a further extension to this approach was proposed in [16], where group preferences are considered as well. A related approach, considering the problem from the somehow dual perspective of extending the general model of CP-theories for preference representation with ontological constraints, was recently proposed in [19]. Another recent approach is the Prob-$\mathcal{EL}$ formalism [9], which extends the $\mathcal{EL}$ description logic with probabilistic uncertainty over both assertional and terminological knowledge.

In a separate but closely related vein, Datalog+/– was also extended with probabilistic models in [8], where the authors study both algorithms for ranking answers with respect to their associated probabilities and query answering under inconsistency. These two lines were considered together in [17], where the authors explore the problem of ranking answers to queries with respect to both probabilistic uncertainty and user preferences. Several other ontology languages have been extended with probabilistic uncertainty—see [18] for a survey of earlier approaches. Also related to this line of research is the study of probabilistic databases [12], where the ontological aspect is missing but the focus is rather on computational tractability. Another quite recent formalism for expressing preferences under uncertainty—also not ontology-based—was introduced in [15].

Stream reasoning [6] refers to the problem of processing information that continuously becomes available and cannot all be stored (a fixed window is generally assumed). From the point of view of making sense of data in social media, the recent work of [3] analyzes key research questions for mining data with semantic content from social media streams. Their work is perhaps the closest in spirit to our goal, though the main difference is that they are focused primarily on extracting information while we are focusing on the problems of adequately organizing and accessing the information that is already extracted.

Towards a general framework

We have thus far proposed a set of desirable properties and sketched the organization of a framework for modeling and reasoning with SKBs; however, there are many challenges towards materializing the general vision. Figure 2 shows a high-level outline of this vision—SKBs are populated by three general sets of sources: social media and general Web-based resources, users themselves, and users' interactions with others. A mix between learning, scraping, and elicitation techniques, as well as knowledge engineering in general, will help obtain not only the information necessary for the individual components of the SKB but also the relationships between them. These components will be built by leveraging as much as possible existing tools (such as Bayesian networks, Datalog+/–, etc.). Even if we assume that all necessary information is available to populate these components, there are many challenges associated with bringing them together: scalability issues arising from the combination of individually tractable components, semantic issues arising from the combination of open-world and closed-world assumptions, alignment issues arising from different schemas used

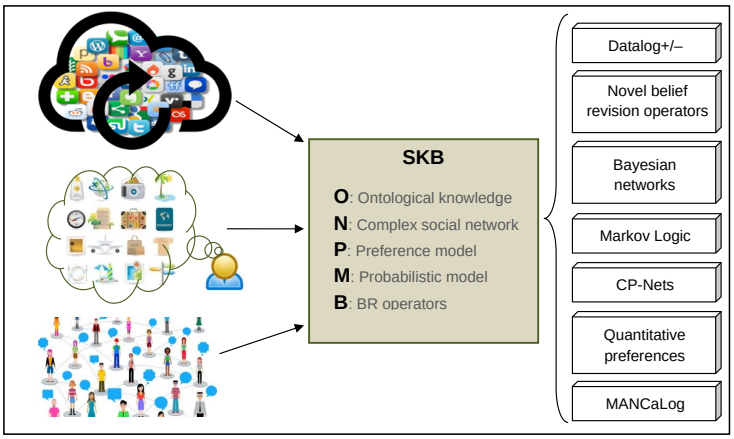


Fig. 2. A high-level overview of the proposed process of modeling and reasoning with social knowledge. SKBs are built with information from the Web, individual users interacting with online services, and social media. Individual components of the SKB are modeled using different kinds of formalisms proposed in the literature for solving more specific subproblems.

in different components, normalization issues arising from combining different quantitative preferences, and so on.

5 Conclusions

In this position paper, we have discussed the need to develop novel knowledge representation and reasoning tools and techniques that are adequate for tackling the challenges that come with modeling social knowledge. We proposed a set of desiderata to guide the development of such formalisms, and briefly outlined how a unifying model can be built by leveraging existing research and novel developments. The main contribution of such a discussion is the proposal of a road map to guide research efforts towards this goal, as well as the novel proposal of combining several research lines that up to now have been considered largely in isolation: ontologies, preferences, uncertainty, stream reasoning, and complex social networks.

Acknowledgments. This work was supported by funds provided by CONICET and Universidad Nacional del Sur, Argentina. Some of the authors of this work were also supported by the U.S. Department of the Navy, Office of Naval Research, grant N00014-15-1-2742. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Office of Naval Research.

References

1. Berners-Lee, T., Hendler, J., Lassila, O., et al.: The semantic web. Scientific american 284(5), 28–37 (2001)
2. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U.: Complex networks: Structure and dynamics. Physics Reports 424(4), 175–308 (2006)
3. Bontcheva, K., Rout, D.: Making sense of social media streams through semantics: a survey. Semantic Web 1, 1–31 (2012)
4. Broecheler, M., Pugliese, A., Subrahmanian, V.: DOGMA: A disk-oriented graph matching algorithm for RDF databases. In: Proc. of ISWC 2009, LNCS, vol. 5823, pp. 97–113. Springer (2009)
5. Calì, A., Gottlob, G., Lukasiewicz, T.: A general Datalog-based framework for tractable query answering over ontologies. J. Web Sem. 14, 57–83 (2012)
6. Della Valle, E., Ceri, S., Van Harmelen, F., Fensel, D.: It's a streaming world! reasoning upon rapidly changing information. IEEE Intell. Sys. (6), 83–89 (2009)
7. Fermé, E., Hansson, S.O.: Agm 25 years. J. of Philos. Log. 40(2), 295–331 (2011)
8. Gottlob, G., Lukasiewicz, T., Martinez, M.V., Simari, G.I.: Query answering under probabilistic uncertainty in Datalog+/– ontologies. AMAI 69(1), 37–72 (2013)
9. Gutiérrez-Basulto, V., Jung, J.C., Lutz, C., Schröder, L.: A closer look at the probabilistic description logic Prob-EL. In: Proc. of AAAI (2011)
10. Kang, C., Pugliese, A., Grant, J., Subrahmanian, V.: STUN: querying spatio-temporal uncertain (social) networks. Soc. Netw. Anal. Min. 4(1), 1–19 (2014)
11. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network. In: Proc. of KDD '03. pp. 137–146. ACM (2003)
12. Koch, C., Olteanu, D., Re, C., Suciu, D.: Probabilistic Databases. Morgan-Claypool (2011)
13. Liu, Y., Kliman-Silver, C., Mislove, A.: The tweets they are a-changin: Evolution of twitter users and behavior. In: Proc. of ICWSM. vol. 13, p. 55 (2014)
14. Lukasiewicz, T., Martinez, M.V., Simari, G.I.: Preference-based query answering in Datalog+/– ontologies. In: Proc. of IJCAI. pp. 1017–1023. IJCAI/AAAI (2013)
15. Lukasiewicz, T., Martinez, M.V., Simari, G.I.: Probabilistic preference logic networks. In: Proc. of ECAI 2014. pp. 561–566 (2014)
16. Lukasiewicz, T., Martinez, M.V., Simari, G.I., Tifrea-Marciuska, O.: Ontology-based query answering with group preferences. ACM Trans. Internet Techn. 14(4), 25:1–25:24 (2014)
17. Lukasiewicz, T., Martinez, M.V., Simari, G.I., Tifrea-Marciuska, O.: Preference-based query answering in probabilistic Datalog+/– ontologies. J. Data Semantics 4(2), 81–101 (2015)
18. Lukasiewicz, T., Straccia, U.: Managing uncertainty and vagueness in description logics for the Semantic Web. J. Web Sem. 6(4), 291–308 (2008)
19. Noia, T.D., Lukasiewicz, T., Martínez, M.V., Simari, G.I., Tifrea-Marciuska, O.: Combining existential rules with the power of CP-theories. In: Proc. of IJCAI 2015. pp. 2918–2925 (2015)
20. Shakarian, P., Simari, G.I., Callahan, D.: Reasoning about complex networks: A logic programming approach. TPLP 13(4-5-Online-Supplement) (2013)
21. Shakarian, P., Simari, G.I., Schroeder, R.: MANCaLog: A logic for multi-attribute network cascades. In: Proc. of AAMAS-2013 (2013)
22. Simari, G.I., Shakarian, P., Falappa, M.A.: A quantitative approach to belief revision in structured probabilistic argumentation. AMAI (2015, *In Press*)
23. Spanos, D.E., Stavrou, P., Mitrou, N.: Bringing relational databases into the semantic web: A survey. Semantic Web 3(2), 169–209 (2012)

Algoritmos de Inteligencia de Enjambres Orientados a Map Reduce

S. Molina y G. Leguizamón

Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)
Universidad Nacional de San Luis,
Ejército de los Andes 950, (5700) San Luis, Argentina
{smolina,legui}@unsl.edu.ar

Resumen La *Inteligencia de Enjambres* involucra acciones de grupos de individuos descentralizados y auto-organizados, las cuales pueden realizarse en paralelo, particularmente, utilizando *Map-Reduce*, un modelo de programación paralela que permite con facilidad conseguir algoritmos escalables. En este trabajo se propone una breve revisión de algoritmos de *Inteligencia de Enjambres* orientadas a *Map Reduce*, observando especialmente su escalabilidad. Se revisan publicaciones de metaheurísticas clásicas como *Optimización de Colonias de Hormigas* y *Optimización de Enjambre de Partículas*, además de metaheurísticas más recientes como *Búsqueda Cuco* y *Optimización de Enjambre de Luciérnagas*.

1. Introducción

La *Inteligencia de Enjambres* (*Swarm Intelligence, SI*) estudia el comportamiento colectivo de sistemas compuestos por muchos individuos (el *swarm*) interactuando localmente y con su entorno. Ejemplos de *SI* son: *Optimización de Colonias de Hormigas* (*Ant Colony Optimization*, ACO) (Dorigo and Blum [9]); *Optimización de Enjambre de Partículas* (*Particle Swarm Optimization*, PSO) (Eberhart and Kennedy [10]); *Búsqueda Cuco* (*Cuckoo Search*, CS) (Yang et al. [24]) y la *Optimización de Enjambre de Luciérnagas* (*Glowworm Swarm Optimization*, GSO)(Krishnanand and Ghose [13]).

Varias razones hacen a los *SI* aptos para aplicar técnicas paralelas en su solución computacional, por un lado el comportamiento de cualquier *swarm* implica acciones independientes de sus individuos. Por otro lado, tienen mejor desempeño cuando se requiere mucho tiempo de procesamiento, como: cuando el *swarm* es de gran tamaño o cuando los problemas a abordar son de gran escala (por ejemplo, *BigData*) en donde se manipulan espacios de búsqueda grandes y se evalúa la calidad de soluciones de alta dimensionalidad.

Map-Reduce, es un modelo de programación paralela, diseñado originalmente por *Google* para simplificar el procesamiento de datos en paralelo sobre grandes clusters, actualmente también es utilizado para el procesamiento de *BigData* [15]. En este modelo, un usuario especifica la computación a través de dos funciones, *Map* y *Reduce*. La librería *Map-Reduce* paraleliza la computación,

maneja la distribución de datos, el balance de carga, la tolerancia a fallas y la asignación de recursos.

El objetivo de este trabajo es realizar un cuidadoso análisis del estado del arte de los algoritmos *SI*: *ACO*, *PSO*, *CS* y *GSO*. Nos focalizamos en algoritmos propuestos desde el año 2012, poniendo especial interés en cómo cada una de las partes algorítmicas de los *SI* son incluidas en las funciones *Map* y *Reduce*, de manera tal de conseguir los beneficios de este modelo, particularmente la escalabilidad.

El trabajo está organizado de la siguiente manera, en la sección 2 se describe brevemente la arquitectura *Map-Reduce* destacando sus principales características. En la sección 3 se realiza una introducción a las metaheurísticas *SI*, para luego a partir de la sección 4 dedicarnos al análisis de los algoritmos *SI* orientados a *Map-Reduce*. Finalmente en la sección 5 se presentan las conclusiones.

2. Architectura Map-Reduce

Map-Reduce es un modelo de programación paralelo (McNabb et al.[18]). La infraestructura provista por su implementación, maneja los detalles de comunicación, balance de carga, tolerancia de fallas, asignación de recursos, inicialización de tareas y distribución de archivos.

Un programa basado en este modelo consiste de una función *Map* y una función *Reduce* las cuales procesan entradas de la forma <*clave,valor*>.

Las operaciones *Map-Reduce* se dividen en dos etapas. En la primera etapa la función *Map* es invocada para cada una de las entradas, en cada invocación puede producir cualquier número de salidas. En la segunda etapa estas salidas son ordenadas y agrupadas según el valor *clave* y una función *Reduce* es invocada por cada *clave*. La función *Reduce* produce una lista de valores de salidas asociados a una *clave* determinada.

Todas las operaciones *Map* pueden realizarse en paralelo. La operación *Reduce* puede comenzar cuando todas las operaciones *Map* se completan. Las operaciones *Reduce* también pueden ser realizadas en paralelo.

Un programa más complejo puede consistir de múltiples etapas *Map-Reduce*, conformando un programa *Map-Reduce Iterativo*. En este caso, la salida de una función *Reduce* es la entrada de una función *Map* de la próxima iteración.

Existen varias implementaciones *Map-Reduce* (Lee et al. [15]) tales como: la original de *Google*, su contraparte *Open Source Hadoop* (Apache [7]) con *HaLoop* para *Map-Reduce Iterativo*); *Mars* (Bingsheng et al. [12]) para procesadores gráficos, Phoenix (Ranger et al. [20]) para sistemas de memoria compartida y *Disco*(Aljarah and Ludwig [3]) para el manejo de grandes vólumenes de datos y cálculos científicos.

3. Inteligencia de Enjambres

La *Inteligencia de Enjambres* estudia el comportamiento colectivo de sistemas compuestos de muchos individuos (*swarm*) interactuando localmente y con

su entorno. Los *swarms* inherentemente usan formas de control descentralizadas y auto-organización para alcanzar sus objetivos (Martens et al. [17]). En este trabajo revisamos los algoritmos orientados a *Map-Reduce* de las metaheurísticas *ACO*, *PSO*, *CS* y *GSO*. A continuación mostramos sus características:

- *ACO* (Dorigo and Blum [9])
  Se inspira en el comportamiento de una especie de hormigas en busca de comida. Éstas, inicialmente exploran aleatoriamente el área cercana al nido, luego se guían por los niveles de $feromona$ que depositan en el entorno, cuando llevan comida al nido.
  Un algoritmo *ACO* básicamente consta de tres procedimientos: *Construcción de Soluciones* (simula una colonia de hormigas, las cuales concurrente y asincrónicamente construyen una solución mediante movimientos a través del grafo de construcción, representante del espacio de búsqueda de la solución), *Modificación de Feromona* (deposita o evapora feromona, modifica la matriz de feromonas τ) y *Acciones Demonios* (son acciones centralizadas que no pueden ser realizadas por una única hormiga).
- *PSO* (Eberhart and Kennedy [10])
  El *PSO* imita la interacción social entre individuos (o *partículas*), tales como la interacción de los pájaros durante el vuelo en búsqueda de comida. En el mismo, un número de *partículas* son colocadas en el espacio de búsqueda. Cada *partícula* mantiene su mejor posición (*Local Best*) según el valor de fitness de una función objetivo dada. Luego, cada *partícula* determina su movimiento (o *velocidad*), teniendo en cuenta la historia de su posición actual y la mejor posición alcanzada por el *swarm* y otras constantes. A su vez, el *swarm* se mueve a lo largo de las iteraciones del algoritmo cerca de la *Mejor Solución Global* (*Global Best*).
- *CS* (Yang et al. [24])
  Se basa en el comportamiento parásito de algunas especies de pájaros cucos al depositar sus huevos en nidos ajenos. Si el dueño del nido descubre que el huevo no es suyo, lo tira o abandona el nido construyendo un nido en otro lugar. Si el huevo del cuco eclosiona el ciclo se repite.
  El algoritmo *CS* comienza con un conjunto de nidos con un huevo cada uno. Los mejores huevos pasarán a la próxima generación. Los peores huevos tendrán una cierta probabilidad de ser descubiertos y no sobrevivirán. Los nidos cuyos huevos han sido descubiertos se sustituirán por nuevos nidos.
- *Optimización de Enjambre de Luciérnagas* (Krishnanand and Ghose [13])
  Se basa en el comportamiento de los enjambres de luciérnagas. Cada luciérnaga tiene una posición dentro del espacio de búsqueda, un *Nivel de luminosidad* asociado con el valor objetivo de su posición y un rango de decisión local. La luciérnaga que emite más luminosidad está más cerca de una posición de interés.
  A lo largo del proceso, las luciérnagas se mueven hacia otras con mayor luminosidad dentro de su rango de decisión local. Finalmente, muchas de ellas quedarán en los picos del espacio de búsqueda.

4. Algoritmos SI orientados a Map-Reduce

La propuesta de este trabajo, surge a partir de la observación de que se han realizado varios trabajos abocados al estudio del estado del arte de metaheurísticas paralelas en general y sus tendencias (Alba et al. [2]), otros más abocados a *SI* específicas (Krömer et al. [14]) y para un tipo de arquitectura particular (Molina et al. [19] para $GPUs$), pero en ninguno de ellos hacen referencia al uso de *Map-Reduce*.

Recientemente, Gong et al. [11] han presentado un estado del arte de modelos y algoritmos evolutivos distribuidos, además de los *SI*: ACO y PSO, en donde mencionan que implementaciones *Cloud*, *Map-Reduce* y $GPUs$ son temas de interés actual para investigar.

En el caso de *Map-Reduce*, no existen trabajos con la idea que proponemos por lo que creemos que es una interesante aportación.

4.1. Revisión de Publicaciones Relacionadas a *SI* sobre *Map-Reduce*

Los algoritmos *SI* sobre *Map-Reduce* buscan mejorar su eficiencia afectada por, incluir técnicas que mejoran aspectos de éstos pero hacen que aumente su tiempo de ejecución (t_e), o para abordar problemas con espacios de búsquedas muy grandes.

Como mencionamos anteriormente, aquí ponemos especial interés en cómo las partes algorítmicas de los *SI* son incluidas en las funciones Map y *Reduce*, para lograr algoritmos escalables.

Distinguimos tres tipos de *Escalabilidad*: A, B y C. La escalabilidad tipo A y B son incluidas en Gong et al. [11] como *Escalabilidad de Tamaño* y *Escalabilidad de Tarea* respectivamente. La escalabilidad tipo C o *Escalabilidad de parámetros* es incluida por nosotros para agregar claridad a nuestra revisión.

La escalabilidad tipo A muestra la habilidad de los algoritmos de mejorar su desempeño en proporción al aumento del número de procesadores. La escalabilidad tipo B muestra la habilidad de los algoritmos para adaptarse a los cambios en la escala del problema, manteniendo su eficiencia cuando se incrementan las dimensiones del mismo. La escalabilidad tipo C muestra la habilidad de los algoritmos de mantener su eficiencia al aumentar la cantidad de procesamiento por un aumento en el valor de algún parámetro del mismo.

A continuación describimos cada una de las publicaciones revisadas. En la mayoría de los casos, se utiliza el framework $Hadoop$. En los casos excepcionales se hacen los comentarios pertinentes. El cuadro 1 resume las características de los trabajos revisados, su nomenclatura se describe a continuación. m_1: *Un individuo construye una solución*, m_2: *Una colonia construye una solución*, m_3: *Un individuo construye una solución parcial*, m_4: *Cálc. de centroides y distancia entre éstos y los datos*, m_5: *Definición de un subespacio de búsqueda*, m_6: *Gen. de huevos*, m_7: *Cálc. de centroides y distancia entre éstos y un conjunto parcial de datos*, r_1: *Sel. del Global Best y actualización de* τ, r_2: *Unificación de matrices* τ *parciales y ejecución de* ACO, r_3: *Cálc. de valores de fitness*, r_4: *CS*, r_5: *Sel. de huevos*, r_6: *Cálc. de centroides y distancia entre éstos y el conjunto*

SI	Modelo	Algoritmo	Problema	Map	Reduce	Año	Ref.
ACO	*Estándar*	*[6]-ACO*	TSP	m_1	r_1	2014	[6]
	Multicolonia	*[8]-ACO*	MST	m_2	r_1	2014	[8]
	Isla Dinámico	*DIIMR-ACO*	TSP	m_1	r_1	2013	[22]
	ACO Independientes	*[21]-Indep*	TSP, PM	m_1	r_1	2012	[21]
	Espacio Particionado	*[21]-Part*	TSP, PM	m_3	r_2	2012	[21]
PSO	*Estándar*	*IDS-MRCPSO*	*Clustering*	m_4	r_3	2013	[5]
	Estándar	*MR-CPCO*	*Clustering*	m_4	r_3	2012	[3]
CS	*Espacio Particionado*	*[23]-CS*	*Benchmark*	m_5	r_4	2014	[23]
	Estándar	*MRMCS*	*Benchmark*, *Ingeniería*	m_6	r_5	2013	[16]
GSO	*Espacio Particionado*	*MRCGSO*	*Clustering*	m_7	r_6	2014	[1]
	Estándar	*[4]-GSO*	*Benchmark*	m_4	r_7	2013	[4]

Cuadro 1. Características de SI revisados.

completo de datos, r_7: *Cálc. de luminosidad*, MST: *Minimum Spanning Tree*, TSP: *Traveling Salesman Problem* y PM: *Problema de la Mochila*.

Metaheurística ACO sobre Map-Reduce

Se han propuesto algoritmos ACO basados en modelos: con una única colonia de hormigas y varias colonias de hormigas independientes; con varias colonias de hormigas que interactúan entre sí como el modelo *Multicolonia de Hormigas* y el modelo *Isla Dinámico* en donde los niveles de colaboración y de competencia entre colonias se gradúa a lo largo del tiempo.

Una única publicación, propone particionar el espacio de búsqueda para distribuir la carga de trabajo entre las hormigas (Wu et al. [21]).

La mayoría de las publicaciones revisadas proponen algoritmos para el TSP (Wu et al. [21], Anuraj and Remya [6] y Cheng and Xiao [22]), Wu et al. [21] también estudian el *Problema de la Mochila* y Bhavani and Sudha [8] estudian un problema de clustering de genes.

- En Anuraj and Remya [6] proponen un *ACO Map-Reduce Iterativo*. Cada Map calcula una solución, la función *Reduce* realiza la modificación de τ y calcula la mejor ruta de cada iteración. La etapa *Reduce* final retorna la mejor solución global. Para mejorar la versión paralela, múltiples hormigas se implementan en un mismo Map.
  El algoritmo es escalable. Se visualiza la escalabilidad tipo A, B y C a través de la observación del t_e. Para la escalabilidad tipo B, se observa que el algoritmo es eficiente para un número grande de ciudades. Para la escalabilidad tipo A, se prueba el algoritmo variando el número de nodos de distintos clusters. Para la escalabilidad tipo C, se evaluan los t_e variando el número de: hormigas por Map, de Map por etapas *Map-Reduce* y etapas *Map-Reduce*; en este caso el incremento del número de hormigas no afecta al t_e pero sí lo hace el incremento del número de Map y etapas *Map-Reduce*.
- Bhavani and Sudha [8] proponen un algoritmo *Multicolonia de Hormigas*. El mismo construye el árbol de expansión mínimo (MST, *Minimum Spanning*

Tree) desde el dato de expresión del gen, utilizando *Map-Reduce Iterativo* y luego realiza el proceso de clustering, utilizando el procedimiento *K-medias*. Cada *Map* es una colonia que construye un MST. En cada generación, cada colonia intercambia información referente a su solución. La función *Reduce* encuentra la mejor solución hasta un momento dado y actualiza τ. Finalmente, se calcula el valor de *Threshold* utilizando los pesos de los arcos del MST mínimo, los arcos que son mayores a este valor son quitados obteniendo de esta manera el clustering.
Se estudia la escalabilidad tipo A y B para diferentes tamaños de archivos de genes y números de procesadores. Utilizan hasta 4 procesadores y archivos de entrada de hasta 17.8 MB. El algorimo tiene buena escalabilidad.

- Cheng and Xiao en [22] proponen el algoritmo *DIIMR-ACO*, basado en un modelo *Isla* dinámico e iterativo. Para disminuir el overhead causado por incluir una técnica de feedback dinámico, utilizan un *Map-Reduce Iterativo* con la plataforma *HaLoop*.
  Las hormigas son divididas en varias colonias, cada hormiga elige el próximo paso según los valores de feromona de las colonias *Inter* e *Inner*.
  La función Map construye un tour, calcula los niveles de feromona para la matriz de la colonia *Inter* e *Inher*, calcula la probabilidad de transición y elige el próximo arco. Finalmente realiza el depósito de feromona según el tour construido. La función *Reduce* modifica los valores de feromona respecto a una hormiga de una determinada colonia y modifica la τ global.
  Estudian la escalabilidad tipo B para diferentes instancias. Para las instancias más grandes (en este caso se utilizan instancias de 198, 318, 442 y 532 ciudades), el algoritmo *DIIMR-ACO* tiene mejor desempeño que un algoritmo *ACO* secuencial y un algoritmo $MMAS$ paralelo.
- Wu et al. [21] implementan dos versiones *ACO*.
  En la primera versión, cada *Map* ejecuta un algoritmo ACO independiente que calcula una única solución. Cada *Reduce*, selecciona el valor óptimo.
  En la segunda versión, el espacio de búsqueda es particionado en el número de tareas Map. Para el problema de la mochila, los items y τ se particionan y se envían a cada tarea Map. Cada Map calcula las soluciones parciales y realiza las modificaciones parciales de niveles de feromonas. En las tareas *Reduce* las matrices parciales son unificadas y el algoritmo *ACO* es ejecutado sobre todos los ítems considerando la τ optimizada. Para el TSP se replican las entradas para cada Map y se le asignan diferentes ramas del árbol de búsqueda, la tarea *Reduce* produce la ruta óptima.
  Los algoritmos son escalables alcanzando valores cercanos al *Speedup Lineal*.
  Se estudia la escalabilidad tipo A utilizando hasta 16 nodos.

Metaheurística PSO sobre Map-Reduce

- Aljarah and Ludwig [3], proponen el algoritmo *MR-CPSO* para clustering, el cual en cada generación aplica dos tareas *Map-Reduce*. En la primera tarea, se modifican las partículas centroides del *swarm* y en la segunda tarea se evalúan sus fitness. Luego, se actualizan los valores *Local Best* y *Global Best*.

En la primera función *Map* se modifican los centroides, la función *Reduce* los ordena y los combina en un archivo de salida.
La segunda función *Map* calcula la distancia entre un registro (dato) y los centroides, luego emite a la función *Reduce* una clave compuesta armada con el identificador del centroide con distancia mínima y dicha distancia. La función *Reduce* calcula el promedio de los valores con igual clave y lo asigna como el nuevo valor de fitness de cada centroide en cada partícula. Luego, todos los valores de fitness son guardados en el *Sistema de Archivo Distribuido*.
Se estudia la escalabilidad tipo A y B, utilizan las métricas *Scaleup* y *Speedup*. Para el *Scaleup* varía el tamaño del conjunto de datos y el número de nodos. Para el *Speedup* se fija el conjunto de datos y varía el número de nodos. El algoritmo escala muy bien alcanza valores de *speedup* cercanos al *Speedup Lineal*.

- Aljarah and Ludwig [5] presentan el algoritmo *IDS-MRCPSO*, un *Sistema de Detección de Intrusión*. Este algoritmo, utiliza el algoritmo *MR-CPSO* descripto en el punto anterior para generar centroides óptimos para los datos de entrenamiento.
  Se evalúa la escalabilidad tipo A utilizando la métrica *Speedup* y el t_e. El algoritmo muestra una escalabilidad razonable para distintos número de nodos (en este caso desde 2 hasta 16). Para el conjunto de datos más grande se observan valores de *speedup* cercanos al *Speedup Lineal*.

Metaheurística CS sobre Map-Reduce

- Xu et al. [23] proponen un algoritmo que combina la técnica *Dividir* y *Conquistar* y *Map-Reduce* con el CS secuencial, para funciones benchmarks.
  Primero el dominio de búsqueda es transformado en una región de 2 dimensiones y es dividido en n^2 subrectángulos no sobrelapados. Luego, un procedimiento Map para cada subrectángulo, arma un par $<$*clave,valor*$>$, con el identificador del subrectángulo como *clave* y como *valor* la posición de su esquina inferior izquierda. Cada par se emite R veces a una etapa *Reduce*. La etapa *Reduce*, ejecuta R veces un algoritmo CS estándar para un mismo subrectángulo, evitando así caer en soluciones óptimas locales. De esta manera, el tiempo de comunicación de la red disminuye al aprovechar la localidad del dato. El mejor resultado es la salida. Finalmente, se comparan los mejores resultados para cada subrectángulo y la salida será la mejor solución global.
  Se compara un algoritmo CS secuencial respecto al CS *Map-Reduce* ejecutado sobre un cluster con dos nodos de 24 cores cada uno. El estudio no está focalizado en el estudio de escalabilidad, aunque se visualiza la existencia de una escalabilidad tipo A, ya que consiguen un menor t_e usando *Map-Reduce* y un mayor número de nodos. Destacan que esto se debe principalmente a que $Hadoop$ propaga todas las tareas de computación a través de los cores de CPU y las reordena dinámicamente.

- Lin et al. [16] proponen el algoritmo *MRMCS* sobre un *Map-Reduce Iterativo* y se aplica a dos problemas de ingeniería y funciones benchmarks.
  Las funciones *Map* se encargan de la generación de un nuevo huevo y la función *Reduce* realiza la selección de los mejores huevos para la próxima generación.
  Estudian la escalabilidad tipo A. El t_e disminuye a medida que aumenta el número de procesadores (en este caso de 1 a 8) mostrando así su eficiencia.

Metaheurística GSO sobre Map-Reduce

- Al-Madi et al. [1] presentan el agoritmo *MRCGSO* para tareas de *clustering*, para encontrar múltiples centroides. Este algoritmo se basa en el algoritmo *CGSO*, en el mismo, cada luciérnaga compite por ser un centroide y trata de cubrir la mayor cantidad de registros de datos. El *MRCGSO* primero a cada luciérnaga le asigna un valor de posición inicial.
  El conjunto de datos es particionado según el número de tareas Map a utilizar. La función Map calcula para cada luciérnaga el número de registro de datos que son cubiertos por ella y la suma total de las distancias entre ella y cada dato cubierto. La función *Reduce* realiza las mismas operaciones con los resultados de todas las tareas Map.
  Se estudia la escalabilidad tipo B. Los resultados revelan que el algoritmo escala muy bien y alcanzan valores de *speedup* cercanos al *Speedup Lineal*.
- Aljarah and Ludwig [4] presentan el algoritmo *MR-GSO* aplicado a funciones multimodales de gran escala con diferentes dimensiones. El *MR-GSO* consiste de las fases: *Inicialización* y *Map-Reduce Iterativo*.
  En la fase *Inicialización* se crea un grupo de luciérnagas con su posición; para cada una se evalúa la función objetivo y se calcula el nivel de luminosidad. Luego, todo el grupo es guardado en un archivo.
  En la fase *Map-Reduce Iterativo*. Cada tarea *Map-Reduce* representa una iteración del algoritmo estándar *GSO*. En cada tarea el algoritmo se focaliza sobre las etapas que consumen tiempo: modificación del nivel de luminosidad, modificación de la posición de las luciérnagas, búsqueda del grupo de luciérnagas vecinas.
  En la tarea *Map* se realiza la búsqueda del grupo de luciérnagas vecinas con mayor luminosidad para cada luciérnaga del *swarm*. En la función *Reduce* se modifican los niveles de luminosidad.
  Estudian la escalabilidad tipo A y C. La métrica utilizada es el *Speedup Paralelo*. El algoritmo es escalable cuando son optimizadas funciones polimodales dificultosas y se utilizan *swarms* grandes.

5. Conclusiones

En este trabajo se logra una breve pero detallada revisión de algoritmos *SI* orientados a *Map-Reduce*, considerando su *escalabilidad*.

El estudio revela que, los *SI* abordados son escalables y en muchos casos alcanzan valores cercanos al *Speedup Lineal*. Inclusive, en los casos que se utiliza

Map-Reduce Iterativo, el cual ha mostrado tener menor desempeño que el *Map-Reduce* de una única iteración, según se menciona en la literatura. Aunque, con el uso de $HaLoop$ se espera que los algoritmos tengan mejor desempeño por las técnicas que el mismo utiliza para el manejo de las iteraciones.

Los algoritmos muestran un mejor desempeño para instancias más grandes de problemas y con pocas comunicaciones entre sus nodos. Esto se debe a que se logra superponer las computaciones con las comunicaciones disminuyeno así el overhead causado por las mismas.

En general, las funciones Map realizan las acciones independientes de los SI, como las realizadas por lo individuos del *swarm*, las funciones *Reduce* se encargan de las acciones centralizadas.

Si bien, el desarrollador no se esfuerza en los aspectos mencionados en la sección 2, debe distribuir cuidadosamente las acciones de un SI en las funciones *Map* y *Reduce* para evitar el exceso de escrituras en archivos además de un incremento del overhead de comunicaciones.

Finalmente, se vislumbra la posibilidad de extender la aplicabilidad de las SI estudiadas a *Big Data*, a problemas con espacios de búsqueda más grandes. En esta dirección de estudio es interesante además, considerar la experimentación en plataformas *Cloud Computing* lo que mostraría la versatilidad de las SI. Modelos SI multicolonias con diferentes técnicas de particionado del espacio de búsqueda, muestran ser alternativas de estudio prometedoras.

Referencias

1. N. Al-Madi, I. Aljarah, and S.A. Ludwig. Parallel glowworm swarm optimization clustering algorithm based on mapreduce. In *Swarm Intelligence (SIS), 2014 IEEE Symposium on*, pages 1–8, Dec 2014.
2. E. Alba, G. Luque, and S. Nesmachnow. Parallel Metaheuristics: Recent Advances and New Trends. *International Transactions in Operational Research*, 20(1):1–48, 2013.
3. I. Aljarah and S.A. Ludwig. Parallel Particle Swarm Optimization Clustering Algorithm Based on MapReduce Methodology. In *Nature and Biologically Inspired Computing (NaBIC), 2012 Fourth World Congress on*, pages 104–111, Nov 2012.
4. I. Aljarah and S.A. Ludwig. A MapReduce Based Glowworm Swarm Optimization Approach for Multimodal Functions. In *Swarm Intelligence (SIS), 2013 IEEE Symposium on*, pages 22–31, April 2013.
5. I. Aljarah and S.A. Ludwig. MapReduce Intrusion Detection System Based on a Particle Swarm Optimization Clustering Algorithm. In *Evolutionary Computation (CEC), 2013 IEEE Congress on*, pages 955–962, June 2013.
6. M. Anuraj and Remya G. Article: A Parallel Implementation of Ant Colony Optimization for TSP Based on MapReduce Framework. *International Journal of Computer Applications*, 88(8):9–12, February 2014. Published by Foundation of Computer Science, New York, USA.
7. Apache. Documentación de Hadoop de Apache, 2014.
8. R. Bhavani and S. G. Sudha. A Novel Ant Based Clustering of Gene Expression Data Using MapReduce Framework. In *International Journal on Recent and Innovation Trends in Computing and Communication*, volume 2, pages 398–402.

International Journal on Recent and Innovation Trends in Computing and Communication, 2014.
9. M. Dorigo and C. Blum. Ant colony Optimization Theory: A Survey. *Theoretical Computer Science*, 344(2–3):243–278, 2005.
10. R. Eberhart and J. Kennedy. A New Optimizer Using Particle Swarm Theory. In *Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium on*, pages 39–43, Oct 1995.
11. Y. Gong, W. Chen, Z. Zhan, J. Zhang, Y. Li, Q. Zhang, and J. Li. Distributed Evolutionary Algorithms and their Models: A Survey of the State-of-the-art. *Applied Soft Computing*, 34:286–300, September 2015.
12. B. He, W. Fang, Q. Luo, N. K. Govindaraju, and T. Wang. Mars: A MapReduce Framework on Graphics Processors. In *Proceedings of the 17th International Conference on Parallel Architectures and Compilation Techniques*, PACT '08, pages 260–269, New York, NY, USA, 2008. ACM.
13. k. N. Krishnanand and D. Ghose. Detection of Multiple Source Locations Using a Glowworm Metaphor with Applications to Collective Robotics. In *Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE*, pages 84–91, June 2005.
14. P. Krömer, J. Platoš, and V. Snášel. Nature-Inspired Meta-Heuristics on Modern GPUs: State of the Art and Brief Survey of Selected Algorithms. *International Journal of Parallel Programming*, 42(5):681–709, 2014.
15. K. Lee, Y. Lee, H. Choi, Y. D. Chung, and B. Moon. Parallel Data Processing with MapReduce: A Survey. *SIGMOD Rec.*, 40(4):11–20, jan 2012.
16. C. Lin, Y. Pai, K. Tsai, C. H. P. Wen, , and L. Wang. Parallelizing Modified Cuckoo Search on MapReduce Architecture. *Journal of Electronic Science and Technology*, 11(2):115–123, June 2013.
17. D. Martens, T. Fawcett, and B. Baesens. Editorial Survey: Swarm Intelligence for Data Mining. *Machine Learning*, 82(1):1–42, January 2011.
18. A. W. McNabb, J. Lund, and K. D. Seppi. Mrs: MapReduce for Scientific Computing in Python. In *SC Companion*, pages 600–608. IEEE Computer Society, 2012.
19. S. Molina, F. Piccoli, and G. Leguizamón. Algoritmos de inteligencia de enjambres sobre gpu:una revisión exhaustiva. In *Congreso Argentino en Ciencias de la Computación*, 2014.
20. C. Ranger, R. Raghuraman, A. Penmetsa, G. Bradski, and C. Kozyrakis. Evaluating MapReduce for Multi-core and Multiprocessor Systems. In *Proceedings of the 2007 IEEE 13th International Symposium on High Performance Computer Architecture*, HPCA '07, pages 13–24, Washington, DC, USA, 2007. IEEE Computer Society.
21. B. Wu, G. Wu, and M. Yang. A MapReduce Based Ant Colony Optimization Approach to Combinatorial Optimization Problems. In *Natural Computation (ICNC), 2012 Eighth International Conference on*, pages 728–732, May 2012.
22. Ch. Xingguo and X. Nanfeng. Parallel Dynamic Island ACO Based on Iterative Map-Reduce Model. *JCIS: Journal of Communications and Information Sciences*, 3(3):139–147, 2013.
23. X. Xu, Z. Ji, F. Yuan, and X. Liu. A Novel Parallel Approach of Cuckoo Search Using MapReduce. *International Conference on Computer, Communications and Information Technology (CCIT 2014)*, pages 114–117, 2014.
24. X. Yang and S. Deb. Cuckoo Search Via Lévy Flights. In *Nature Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*, pages 210–214, Dec 2009.

Evaluación de dos nuevos algoritmos en el diseño de granjas eólicas

Fabricio Loor, Guillermo Leguizamón y Javier Apolloni

Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)
Departamento de Informática - FCFMN
Universidad Nacional de San Luis, Argentina
loorfabricio@gmail.com,{legui,javierma}@unsl.edu.ar

Resumen En los últimos años el crecimiento en el consumo de energía eléctrica ha sido exorbitante, lo cual ha generado la necesidad de utilizar un recurso prometedor como el viento para extraer dicha energía. La distribución de turbinas de viento dentro de una granja eólica, con el objeto de optimizar la energía capturada, es un problema complejo de resolver. En este artículo se intenta solucionar este problema abordándolo de dos formas distintas: una es la adaptación del algoritmo GWO para vectores booleanos y la otra, DonQuijote, es un método nuevo que incluye el uso de la Evolución Diferencial y surge del análisis del problema. Para mostrar la eficiencia de los métodos se comparan con un Algoritmo Genético, tan estudiado en el área. La mejor propuesta participó en la competencia WFLO de la GECCO 2015.

Palabras claves: energía eólica, diseño de granjas eólicas, optimización, evolución diferencial, metaheurísticas

1. Introducción

El diseño de granjas eólicas es un tema muy estudiado [1] y aún sigue en auge principalmente por las siguientes razones:

- Las energías renovables han sido un recurso ampliamente requerido y con un futuro prometedor.
- Son fuentes seguras y ecológicas para satisfacer la demanda energética actual.
- Encontrar la disposición óptima de las turbinas en una zona reservada para una granja eólica es un problema muy complejo. El número desmesuradamente grande de posibles lugares dentro de la granja para colocarlas lo convierte en un problema intratable. Claramente, el gran tamaño del espacio de búsqueda hace imposible utilizar un algoritmo de optimización mediante una búsqueda exhaustiva. Por este motivo, es necesaria otra clase de métodos de optimización. En este sentido, existe un gran repertorio de métodos y es importante analizar cuál es la mejor posibilidad.

De acuerdo a dónde y cuántas turbinas eólicas son ubicadas se podrá obtener grandes beneficios económicos. Una mala distribución puede disminuir significativamente la energía que las turbinas, en su conjunto, producen. Por esto resulta muy conveniente aplicar métodos estocásticos que permitan encontrar una solución provechosa en tiempos de ejecución aceptables, sacrificando la exactitud de la solución.
Como base para probar los métodos, se ha hecho uso de un Algoritmo Genético

(GA) simple, el cual fue dado como *baseline* en la competencia WFLO de GECCO 2015 [11]. El algoritmo dado en la competencia no tiene en cuenta la información del problema, por lo tanto, son admisibles muchas mejoras. En este artículo, se ha desarrollado un enfoque competitivo de un algoritmo relativamente complejo que considera las influencias entre las turbinas. Los resultados de la simulación demuestran la eficacia de la mejora.
Dentro de la literatura se pueden encontrar varios algoritmos considerados computacionalmente eficientes para resolver este problema, por ejemplo, GA[2], CMA-ES [3] y TDA [4] , DEVO-I [5], DEVO-II [6], PSO [7]. En el presente trabajo, se ha utilizado un algoritmo que incluye Evolución Diferencial (DE) para encontrar la mejor configuración de acuerdo a la zona de ubicación de las turbinas. Por lo tanto, se exhibe un nuevo enfoque a este problema. La sección 2 proporciona una visión general al problema de distribución de turbinas y los costos involucrados. En la sección 3 se describen los algoritmos propuestos. En la sección 4 se presentan los resultados experimentales y el análisis respectivo. La sección 5 incluye las conclusiones y posibles trabajos futuros.

2. Descripción del problema

Suponiendo un plano cartesiano para distribuir las turbinas y considerando que el plano admite coordenadas con valores continuos, la ubicación de cada turbina se indica mediante un par de variables que establecen una posición dentro de este plano. Los límites del plano son dados como parámetros. El plano también es conocido como escenario. Una solución al problema sería, por lo tanto, una secuencia de puntos (x, y) que satisfacen con estar dentro de los límites del escenario y, por condiciones de seguridad, cada punto se encuentra a una distancia fija de los restantes. Esta distancia fija es cuatro veces el tamaño del rotor de una turbina.
La competencia adopta como esquema para representar los escenarios el que se describe en [8]. El viento es simulado mediante un modelo homogéneo, es decir, todas las zonas del escenario reciben el mismo viento, excepto que exista una turbina que interfiera en la captación de la energía. Si esto último sucede, se utiliza una simulación del comportamiento del viento de manera estocástica mediante la distribución Weilbull (ver más detalles en [8]).
Además, las turbinas son ubicadas de forma tal que el rotor de las mismas esté orientado perpendicularmente a la dirección del viento [8].

2.1. Restricciones

Los escenarios cuentan con zonas prohibidas, llamadas obstáculos, para la ubicación de las turbinas. Los obstáculos son modelados como rectángulos, y una solución es considerada inválida si tiene una turbina dentro de algún obstáculo. En secciones posteriores se representa gráficamente un obstáculo.

2.2. Modelo del costo

Para determinar qué tan buena es una solución propuesta se sigue el estudio propuesto para la competencia WFLO 2015 [11], donde la función de fitness tiene dos variables confrontadas: la cantidad de turbinas y la potencia producida por la

configuración.

$$fitness = \frac{(c_t * n + c_s * floor(\frac{n}{m}))(\frac{2}{3} + \frac{1}{3} * e^{-0,00174 * n^2}) + c_{OM} * n}{\frac{(1-(1+r)^{-y})}{r}} * \frac{1}{8760 * P} + \frac{0,1}{n}$$

donde c_t, c_s, m , r, c_{OM} e y son constantes. Por otro lado, n y P son las variables que están en juego en las configuraciones de la granja. La constante c_t es el costo de una turbina, desde la creación de la misma hasta la instalación. En nuestro modelo de costo, el valor de c_t es de 750000 dólares. La constante c_s es el costo de la subestación, el cual es de 8000000 dólares. El significado de m es la cantidad de turbinas que entran por subestación. r (*rate*) es la tasa de interés anual; para la competencia, r tiene un valor de 0,03. La constante y (*year*) hace referencia al tiempo de vida de la granja eólica; para el problema se estima que el tiempo de vida promedio de una granja es de 20 años. Por último, la constante c_{OM} es el costo de operación anual por turbina, que, en nuestro caso, es de 20000 dólares. En cuanto a las variables, n es la cantidad de turbinas propuestas al diseñar la configuración. P es la función que devuelve la energía total de la granja a partir de la posición de cada una de las turbinas y se define mediante la siguiente ecuación:

$$P(M) = e_c/(w_f * length(M)) \tag{1}$$

En la ecuación 1, M es una matriz que describe las posiciones de cada una de las turbinas de la configuración, e_c es la energía capturada por la configuración propuesta, w_f es la energía que se pierde en la granja (*wake free energy*) y $length(M)$ corresponde con la cantidad propuesta de turbinas.

3. Un estudio comparativo

En esta sección se mostrará cómo las propuestas desarrolladas se cotejan con otro algoritmo de búsqueda estocástico. Después de la explicación de todos los métodos, damos paso para que en la siguiente sección se muestre una comparación de los resultados experimentales. En un principio, exponemos las estructuras de los algoritmos genéticos binarios, luego el GWO binario desarrollado y por último, nuestro algoritmo que se ha adaptado al problema. Cabe destacar que en todos los algoritmos descriptos las soluciones son vectores binarios que indican si en una posición (x, y) determinada se debe o no se debe colocar una turbina.

3.1. Variante binaria de un Algoritmo Genético (GA)

GA es un algoritmo metaheurístico inspirado en la evolución biológica y su base es genético-molecular. La analogía del algoritmo se encuentra planteada en que las soluciones al problema, consideradas individuos, son sometidas a acciones aleatorias que se identifican como recombinaciones (cruce), mutaciones y selecciones. El proceso esencial de un GA se muestra en el Algoritmo 1.

Algoritmo 1 Pseudocódigo del GA

```
1: generar P(0); //iniciar una población con soluciones candidatas aleatorias
2: evaluar P(0); //Evaluar cada candidato con la función de fitness
3: para e = 1 a EVALMAX hacer
4:    P'(e) = seleccionar( P(e) ); //Seleccionar padres
5:    P"(e) = aplicar operadores( P'(e) ); //Cruzar padres y generar una nueva población
6:    P(e+1) = reemplazar( P(e), P"(e) ); //Mutar nuevos individuos
7:    evaluar P(e+1);
8:    e = e + 1;
9: fin para
```

3.2. Variante binaria del algoritmo Grey Wolf Optimizer (GWO)

El algoritmo GWO simula el comportamiento natural de una manada de lobos para cazar a su presa. Dentro de los lobos se distinguen distintas categorías como $alfa, beta, delta$ y $omega$ para la simulación. El mecanismo que sigue es sencillo. El primer grupo de lobos $alfa$ son los que lideran la manada y, por lo tanto, influyen de una manera más contundente en el espacio de búsqueda. El pseudocódigo es presentado en el Algoritmo 2. Al final de la ejecución, el mejor individuo va a ser

Algoritmo 2 Pseudocódigo del GWO

```
1: generar (X) // Inicializar la población de lobos grises
2: inicializarParametros(a, A, C);
3: evaluar X(0);
4: seleccionarnuevos(Alpha, Betha, Delta, X(0));
5: para e = 1 a EVALMAX hacer
6:    para todo Lobo l en Omega hacer
7:       para i = 0 a DIM hacer
8:          actualizarPosicion(l, i); //Actualizar la posición actual
9:       fin para
10:      ajustarParametros(a, A, C); //Ajusta los parametros del algoritmo
11:      evaluar X(e+1);
12:      seleccionarnuevos(Alpha, Betha, Delta, X(e+1));
13:      e = e + 1;
14:   fin para
15: fin para
```

el que esté asociado al lobo $alfa$. El parámetro a es el factor de exploración que comienza con un valor de 2 y va decrementándose a lo largo de las evaluaciones hasta llegar a 0.
Los parámetro A y C son vectores que tienen tres números aleatorios distintos y que ayudan a las soluciones candidatas, moviéndolas en el espacio de búsqueda.
La clave del algoritmo se encuentra en la función actualizarPosición(..) (ver línea 8 del Algoritmo 2) que se define como:

$$\vec{X}_{l,i} = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{2}$$

Donde cada $\vec{X}_1$, $\vec{X}_2$ y $\vec{X}_3$ se obtiene de la siguiente manera:

$$\vec{X}_1 = |\vec{x}_\alpha - \vec{A}_1 * \vec{D}_\alpha|, \qquad \vec{X}_2 = |\vec{x}_\beta - \vec{A}_2 * \vec{D}_\beta|, \qquad \vec{X}_3 = |\vec{x}_\delta - \vec{A}_3 * \vec{D}_\delta| \quad (3)$$

Por último para determinar $\vec{D}_\alpha$, $\vec{D}_\beta$ y $\vec{D}_\delta$, usados en (3) se aplican las siguientes fórmulas:

$$\vec{D}_\alpha = |\vec{C}_1 * \vec{X}_\alpha - \vec{X}|, \qquad \vec{D}_\beta = |\vec{C}_2 * \vec{X}_\beta - \vec{X}|, \qquad \vec{D}_\delta = |\vec{C}_3 * \vec{X}_\delta - \vec{X}| \quad (4)$$

En este trabajo, se ha convertido el algoritmo inicialmente planteado para números reales en valores booleanos. Para cumplir este propósito, se ha discretizado el resultado de la fórmula (2) siguiendo el criterio que si $\vec{X}_{l,i}$ es mayor que 1, la función devuelve 1, en otro caso, devuelve 0.

3.3. Un nuevo modelo: DonQuijote

En este apartado, introducimos la idea de nuestro algoritmo de optimización predilecto. Para esto vamos a dividir el algoritmo en tres etapas: inicialización, preprocesamiento y un paso iterativo.

Inicialización:
Se generan varias poblaciones que contengan cuatro soluciones aleatorias con (i * 30) - 1 turbinas, donde i comienza desde el máximo valor posible y es decrementada a medida que se van generando nuevas poblaciones.
El hecho de que la cantidad de turbinas vaya saltando de a 30 unidades se debe a que la función de fitness tiene en su numerador la función piso ($floor$), entre una división del número de turbinas y una constante 30 (número de turbinas por subestación); esto ha resultado en una ventaja muy buena para discretizar el espacio de búsqueda.
Siguiendo con el proceso, se selecciona la población que tiene mejor promedio de fitness y/o la que tiene mejor individuo.

Pre-procesamiento:
En esta etapa se agregan varios indiviuos a la población seleccionada anteriormente para evitar centralizar las soluciones en un solo sector del espacio de búsqueda. En primer lugar, se agrega un individuo que representa una disposición de turbinas en posiciones formando líneas, intercalados en la parrilla, perpendicular al valor más alto de la matriz de salida y en la misma dirección de la más pobre de la misma matriz. Al mismo tiempo, se añaden otros nuevos individuos con un número fijo de turbinas ubicadas aleatoriamente.

Paso iterativo:
La estructura principal de esta fase se expresa en el algoritmo 3. Donde el método primeraOperacion() selecciona una turbina para que sea movida a una posición vacía indicada por la peor recepción de energía de entrada.
El método segundaOperacion() genera, para las primeras evaluaciones nuevas distribuciones aleatorias sin cambiar el número de turbinas. Cuando el número de evaluaciones es lo suficientemente grande, se selecciona la turbina que tiene el peor

Algoritmo 3 Estructura general del paso iterativo

```
1: para e= 1 a MAXEVAL  hacer
2:    para todo individuo en la población hacer
3:       switch (e %4)
4:       case 0:
5:          primeraOperacion();
6:       case 1:
7:          segundaOperacion();
8:       case 2:
9:          terceraOperacion();
10:      case 3:
11:         cuartaOperacion();
12:      end switch
13:   fin para
14: fin para
```

promedio de recepción de energía, para que desde ésta se elija una turbina que se encuentra en la mejor dirección (de las 24 posibles), y sobre esta última se realice una reubicación en la peor dirección de la primera.
El método terceraOperacion() aplica DE binaria con una probabilidad de recombinación de 0,783 y una probabilidad de mutación de 0,06.

Evolución Diferencial binaria: La DE, al igual que los algoritmos genéticos, forma parte de la categoría de la computación evolutiva; aplica pasos similares (cruzar, mutar y seleccionar) pero sus comportamientos son distintos.
La DE fue planteada inicialmente para espacios continuos. Pero en nuestro caso las soluciones son vectores booleanos, que están asociado al número de generaciones o evaluaciones (e). El vector puede ser representado como:

$$X(i, e) = (x_{ie}(1), x_{ie}(2), \ldots, x_{ie}(D)),$$

donde i representa el índice del vector en la población, o también conocido como el número del individuo, D significa la cantidad posible de lugares en donde pueden ubicarse turbinas [10].
La operación de mutación, imita sobre un vector base r_0 otros vectores dependiendo de un número aleatorio y la distancia de Hamming de otras dos soluciones distintas a la primera. El resultado de la distancia es multiplicado por un factor de mutación (0,06). Esta constante controla la velocidad y robustez de la búsqueda. Para la etapa de recombinación (crossover) se mezclan la solución actual sin mutar con la solución mutada. Cuan similar será el vector resultante al vector mutado va a estar determinado por la probabilidad de recombinación (0,783).
Por último el operador de selección elige, de forma determinística, la mejor solución entre el producto obtenido en la etapa del cruce y el miembro actual de la población.

El método cuartaOperacion() genera, para las primeras evaluaciones, distribuciones aleatorias decrementando el número de turbinas por uno. Cuando el número de evaluaciones es lo suficientemente grande, se aplica la primera operación.

4. Resultados experimentales

Para mostrar el desempeños de los distintos algoritmos se ha optado por tomar 5 escenarios, extraidos de la competencia WFLO 2015 [11]. La representación de los escenarios es detallada en la Fig. 1, donde los rectángulos azules son los obstáculos, es decir, lugares donde no es posible ubicar turbinas.

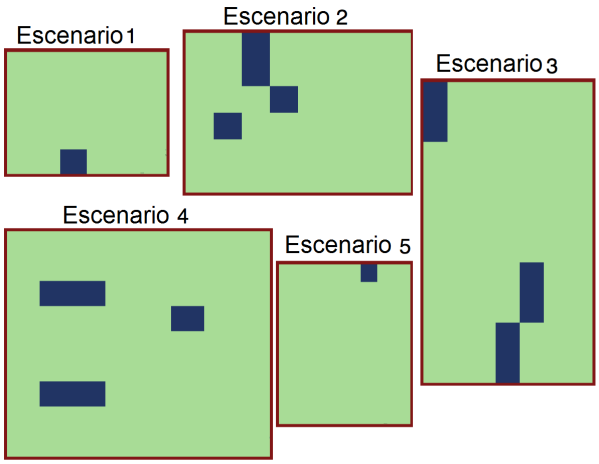


Figura 1. Tamaño de los escenarios y ubicación de los obstáculos

La Tabla 1 nos ayuda a comparar los escenarios entre sí dado que muestra el detalle de los parámetros utilizados en la prueba de cada uno de ellos.
La energía perdida a la que hace referencia la Tabla 1 es un factor que influye en la captura de energía de los aerogeneradores y que depende de las condiciones del terreno. Esta variable es definida en [8]. Mientras que, la cantidad máxima de turbinas es el número de turbinas que pueden ser ubicadas en el escenario formando hileras verticales y cumpliendo las restricciones de seguridad entre dos turbinas.

Característica	Escenario 1	Escenario 2	Escenario 3	Escenario 4	Escenario 5
Ancho	9240	6545	6930	10780	5390
Alto	6545	5005	12320	9240	6545
Energía perdida	6148.648	8674.542	12344.639	11314.82	7441.038
Cantidad máxima turbinas	607	362	843	967	390

Tabla 1. Parámetros de los escenarios usados en el estudio experimental

Para realizar las ejecuciones se hizo uso de una computadora con las siguientes características: Procesador Intel(R) Core(TM) i3-2330M CPU @2,20 GHz; Memoria física: 4044 MB; Sistema operativo Windows 7 Home Basic de 64 bits; las propuestas de este artículo fueron programadas utilizando el lenguaje JAVA.

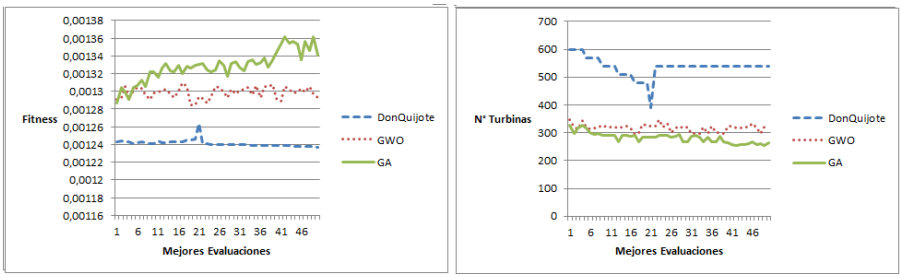
El resultado de las ejecuciones se observa en la Fig. 2. Por un lado, en las gráficas sobre la columna izquierda de la Fig. 2 se observan los resultados obtenidos por cada uno de los algoritmos en cada escenario considerando el costo de producir un kW. Por otro lado, para la misma Fig. 2, las gráficas sobre la columna de la derecha muestran la cantidad de turbinas utilizadas en cada configuración de los escenarios. De acuerdo con lo que se muestra en la Fig. 2, DonQuijote es capaz de proporcionar resultados muy competitivos y obtener, en todos los casos, el mejor valor final de fitness. Cabe destacar el comportamiento intuitivo de DonQuijote en la etapa de explotación en relación al número de turbinas (gráficos de la derecha en la Fig. 2); en esas mismas gráficas se puede observar la etapa de inicalización de DonQuijolte. El trazado inicial en forma escalonada de la gráfica se produce debido a que el algoritmo prueba, durante la inicialización, diferentes configuraciones con una cantidad fija y múltiplo de (i*30) -1 de turbinas, como se dijo previamente en la sección 3.3. Por lo tanto, estos resultados muestran que elegir el número de turbinas de esta forma provoca un rendimiento superior del algoritmo. La mejoría en el desempeño se debe a que el algoritmo explota más rápidamente un sector del espacio de búsqueda. Esta cualidad puede ser provechosa en el proceso de diseño de parques eólicos que se modelen con la función de fitness tratada aquí. En la siguiente sección vamos a discutir los beneficios y desventajas de este enfoque en el proceso de diseño de granjas eólicos.
Con respecto a los tiempos de ejecución para cada uno de los tres enfoques sobre los 5 escenarios son los siguientes:

- GA: 169 minutos con 45 segundos.
- GWO Binario: 173 minutos con 29 segundos.
- DonQuijote: 932 minutos 21 segundos.

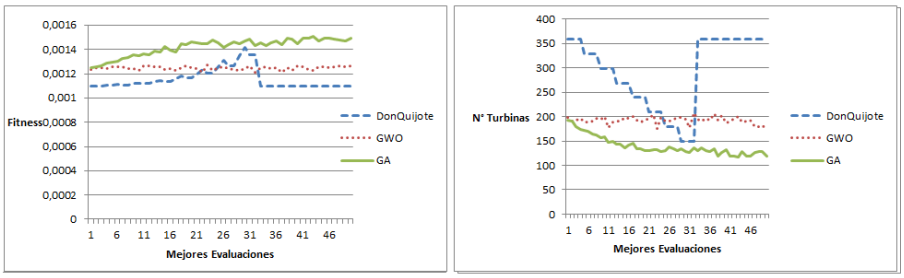
Se debe enfatizar que los tiempos no son un factor de gran énfasis en la competencia puesto que las ejecuciones son locales con respecto a cada competidor. Pero, a pesar de esto no deja de ser elemento a tener en cuenta para las posibles mejoras de DonQuijote.

5. Conclusiones y trabajos futuros

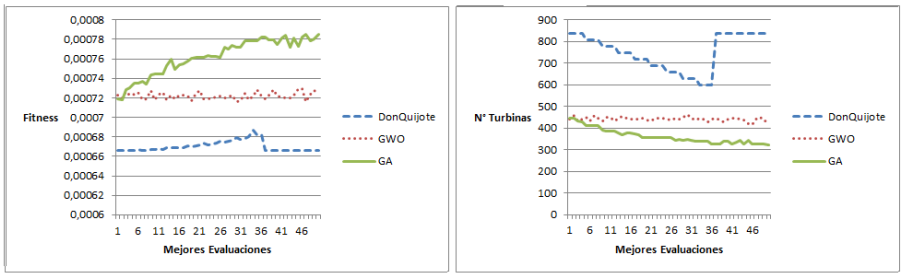
En este trabajo hemos presentado dos nuevos algoritmos para resolver el problema de configuraciones óptimas de granjas eólicas (WFLO). El mejor algoritmo combina una metaheurística con un fuerte análisis del problema, que cooperando juntos y en simultáneo son usados para optimizar el costo de producción de un kilowatt. La etapa de inicialización es utilizada para acotar en gran medida el espacio de búsqueda, la etapa de preprocesamiento sirve para explotar el sector del espacio de búsqueda seleccionado y el algoritmo evolutivo junto con otras operaciones propias del problema son utilizados para mejorar las soluciones actuales.
Los resultados preliminares muestran que el algoritmo propuesto (DonQuijote) supera al *baseline* de la competencia mediante la producción de la mejora de los resultados para el conjunto de casos probados, sin embargo posteriormente se podrían mejorar los tiempos de ejecución del mismo. Por otro lado, se ha corroborado experimentalmente los buenos resultados que arrojan los algoritmos basados en aprendizaje evolutivo con respecto a este problema.



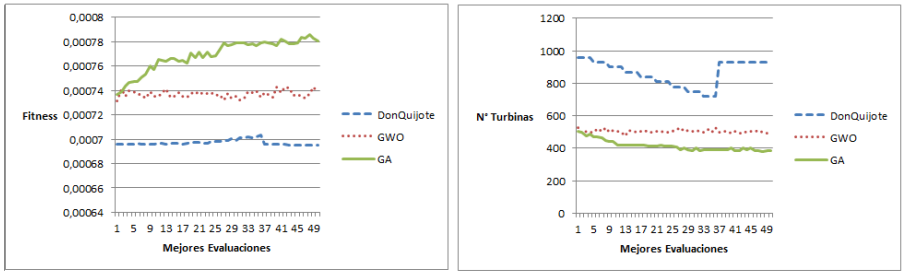
Escenario 1



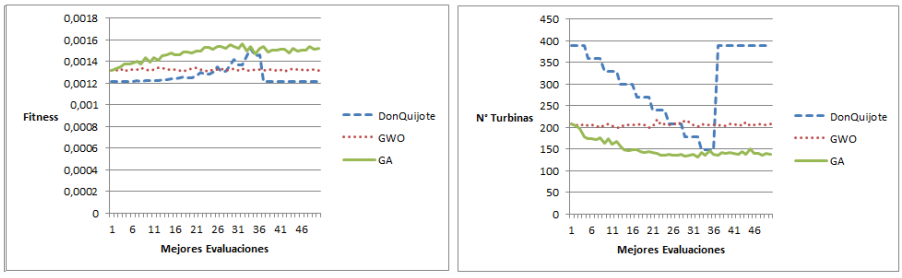
Escenario 2



Escenario 3



Escenario 4



Escenario 5

Figura 2. Comparación entre el GA y nuestros dos modelos desarrollados

A futuro se piensa probar de ubicar las turbinas en cualquier lugar, eliminando la grilla estática y guardando solo la condición de que no entren en el perímetro de seguridad de otra turbina. Otro trabajo es anticiparse a los cambios que se presenten en la competencia del año 2016 al considerar el terreno en un espacio tridimensional.

Reconocimientos

Los autores de este artículo agradecen al LIDIC (Laboratorio de Investigación y Desarrollo en Inteligencia Computacional) y al financiamiento de la Universidad Nacional de San Luis mediante el PROICO 330214.

Referencias

1. R. Wiser y M. Bolinger Annual report on U.S. wind power installation, cost, and performance trends: 2006. Available from. Golden, CO: NREL, US Department of Energy, `http://www.nrel.gov/wind/pdfs/41435.pdf`, (2007).
2. K.S. Tang, K.F. Man, S. Kwong, Q. He, Genetic algorithms and their applications, IEEE Signal Processing Magazine vol. 13 (6): 22—37, (1996).
3. M. Wagner, K. Veeramachaneni, K. Neumann, y U. O'Reilly. Optimizing the Layout of 1000 Wind Turbines.European Wind Energy Association Annual Event: 1–10, (2011).
4. M. Wagner , J. Day y F. Neumann. A Fast and Effective Local Search Algorithm for Optimizing the Placement of Wind Turbines, Renewable Energy vol. 51: 64–70, (2013)
5. D. Wilson, E. Awa, S. Cussat-Blanc, K. Veeramachaneni, y U.-M. O'Reilly. On learning to generate wind farm layouts. In Proceeding of the fteenth annual conference on Genetic and evolutionary computation conference, (2013).
6. D. Wilson , S. Cussat-Blanc, K. Veeramachaneni, , U. O'Reilly, y H. Luga. A Continuous Developmental Model for Wind Farm Layout Optimization. Proceedings of the GECCO. 745–752, (2014)
7. Veeramachaneni, K.; Wagner, M.; O'Reilly, U.-M.; Neumann, F., Optimizing energy output and layout costs for large wind farms using particle swarm optimization, Evolutionary Computation (CEC), IEEE Congress on, 1–7, (2012)
8. A. Kusiak y Z. Song. Design of wind farm layout for maximum wind energy capture. Renewable Energy, vol. 35(3):685–694, (2010).
9. S. Mirjalili, S. M. Mirjalili, A. Lewis. Grey Wolf Optimizer, accepted in Advances in Engineering Software , vol. 69: 46–61, (2013)
10. T. Gong y A. L. Tuson. Differential Evolution for Binary Encoding, Soft Computing in Industrial Applications, ASC 39, 251–262, (2009)
11. 2nd Edition of the Wind Farm Layout Optimization Competition. GECCO. `http://www.irit.fr/wind-competition/`, fuente de los escenarios: `https://github.com/d9w/WindFLO/tree/master/Wind%20Competition/2015/Scenarios`, (2015)

Clasificación de configuraciones de manos del Lenguaje de Señas Argentino con ProbSOM

Franco Ronchetti, Facundo Quiroga, César Estrebou, Laura Lanzarini

Instituto de Investigación en Informática LIDI, Facultad de informática,
Universidad Nacional de La Plata
{fronchetti,fquiroga,cesarest,laural}@lidi.unlp.edu.ar

Resumen El reconocimiento automático de lenguaje de señas es una temática actual de sumo interés dentro del reconocimiento de gestos humanos. Por un lado, su complejo campo de aplicación presenta un desafío que requiere la intervención de diferentes áreas del conocimiento como el procesamiento de video, de imágenes, los sistemas inteligentes y la lingüística. Por otro lado, la correcta clasificación de las señas podría facilitar la traducción e integración a personas con discapacidad auditiva. El presente trabajo tiene dos principales aportes: por un lado la confección de una base de datos de configuraciones de manos del Lenguaje de Señas Argentino (LSA), temática prácticamente no encontrada en el estado del arte. En segundo lugar, el procesamiento de las imágenes, extracción de descriptores y posterior clasificación de la configuración por medio de una adaptación supervisada de los mapas auto-organizativos llamada ProbSom. Dicha técnica se compara con otras del estado del arte como Máquinas de Soporte Vectorial (SVM), Random Forest, y Feedforward Neural Networks.
La base de datos desarrollada contiene 800 imágenes con 16 configuraciones de LSA lo que permite ser un paso inicial hacia la confección de una base de datos de señas argentinas completa. A su vez, la extracción de características propuestas sumadas al clasificador neuronal demostraron ser sumamente eficaces, con una tasa de acierto superior al 90 %.

Keywords: reonocimiento de configuraciones de manos, reconocimiento de formas de mano, reconocimiento de lenguajes de seña, ProbSom, mapas auto organizativos, SOM, transformada Radon, SIFT, Scale-Invariant Feature Transform

1. Introducción

El reconocimiento automático de señas es un problema multidisciplinar sumamente complejo que hoy en día sigue sin ser resuelto en forma total. Si bien en el último tiempo han habido avances en el reconocimiento de gestos, impulsados principalmente por el desarrollo de nuevas tecnologías, aún queda un largo camino por recorrer para construir aplicaciones precisas y robustas que permitan la traducción e interpretación de los gestos realizados por un intérprete [1]. La

compleja naturaleza de los gestos motivan esfuerzos de diversas áreas de investigación como interacción hombre-máquina, visión por computador, análisis de movimientos, aprendizaje automático y reconocimiento de patrones. El lenguaje de señas, y particularmente el Lenguaje de Señas Argentino (LSA), es una temática muy impulsada actualmente por gobiernos y universidades para incluir a persona hipoacúsicas en diferentes entornos. Existe poca documentación y aún menos información en formato digital.

La tarea completa de reconocer un gesto de lenguaje de señas involucra diferentes pasos: la ubicación de las manos del intérprete, el reconocimiento de las formas de las manos (configuraciones), y el seguimiento de las manos para detectar el movimiento realizado, interpretación semántica y traducción al lenguaje escrito [1]. Estas tareas pueden ser desarrolladas y evaluadas en forma separada ya que cada una tiene su complejidad particular. Existen diferentes enfoques para el seguimiento de la mano: algunos utilizando sistemas 3D como el MS Kinect y otros simplemente con una imagen 2D proveniente de una cámara RGB. Incluso existen sistemas con sensores de movimiento como guantes especiales, acelerómetros, etc.

El trabajo presentado en este documento se enfoca en el problema de clasificación de configuraciones de manos. En particular, este se centra en la extracción de características representativas de la mano y en el reconocimiento de dichas configuraciones utilizando una variante de red neuronal competitiva supervisada denominada ProbSom [3]. El trabajo tiene como finalidad generar una subunidad (*Handshape Sub-Unit*) de procesamiento para el reconocimiento automático de lenguaje de señas. En [5] se incorpora el concepto de *subunidad léxica* para modularizar el reconocimiento del gesto.

Una particularidad del lenguaje de señas es que cada región a nivel mundial tiene su propio léxico y grupo de señas que lo representan. Esto lo hace un problema diverso, y diferente de abordar en cada región, ya que nuevos gestos o configuraciones de manos involucran nuevos desafíos no contemplados con anterioridad. En particular, para el Lenguajes de Señas Argentino (LSA) practicamente no existen sistemas y bases de datos que representen los gestos que posee. En este trabajo se aborda también la confección de una base de datos de 16 configuraciones de LSA interpretados por 10 personas distintas. Las imágenes obtenidas fueron utilizadas luego para el proceso de extracción de características y posterior clasificación.

En la literatura existen numerosos trabajos desarrollados que abordan el reconocimiento automático de lenguajes de señas. No obstante, cada trabajo presenta un escenario particular, a veces difícil de replicar completamente, o con ciertas limitaciones. Por ejemplo, diferente trabajos utilizan sensores de profundidad como el MS Kinect, o similares para capturar imágenes 3D. En [8],[12] y [9] se utilizan imágenes de profundidad para clasificar configuraciones del lenguaje de señas norteamericano (ASL). Estos enfoques en general presentan dos problemas: por un lado la necesidad de contar con un equipo de similares características con el que fue probado, y por otro lado la alta tasa de error que todavía tienen estos dispositivos (al menos los de un costo bajo) para calcular

las imágenes de profundidad. Otros enfoques, como el que se presenta en este trabajo, utilizan sólo imágenes RGB. En [10] se crea un modelo probabilístico de color de piel para detectar y seguir las manos del intérprete en un video. En [2] se utiliza este modelo para segmentar las manos y aplicar un clasificador basado en Modelos de Markov. En general los sistemas basados únicamente en color de piel no son robustos a la variabilidad en el fondo o la vestimenta del intérprete, y en las oclusiones mano-mano o mano-cara. Para realizar un reconocimiento de la posición de la mano suele ser necesario adicionar información morfológica al filtrado de color. Por último, en [1] se hace una gran revisión del estado del arte en el reconocimiento de lenguaje de señas.

El presente documento se organiza de la siguiente manera: en la sección 2 se describe la base de datos generada, el procesamiento de las imágenes y la extracción de características de la mano y el modelo de clasificación utilizado. En la sección 3 se detalla la experimentación y finalmente en la sección 4 se exponen las conclusiones generales.

2. Métodos

2.1. Base de datos de configuraciones de Lengua de Señas Argentina (LSA16)

La base de datos de configuraciones de Lengua de Señas Argentina [1], creada con el propósito de producir un diccionario de LSA y entrenar un traductor automático de señas, contiene 800 imágenes en donde 10 sujetos realizaron 5 repeticiones de 16 tipos distintos de configuraciones de mano utilizadas en distintas señas de dicho lenguaje. Las configuraciones fueron elegidas dentro de las más utilizadas en el léxico, y se pueden observar en la figura 1. Cada configuración fue realizada repetidamente en diferentes posiciones y diferentes rotaciones en el plano perpendicular a la cámara, para generar mayor diversidad y realismo en la base de datos.

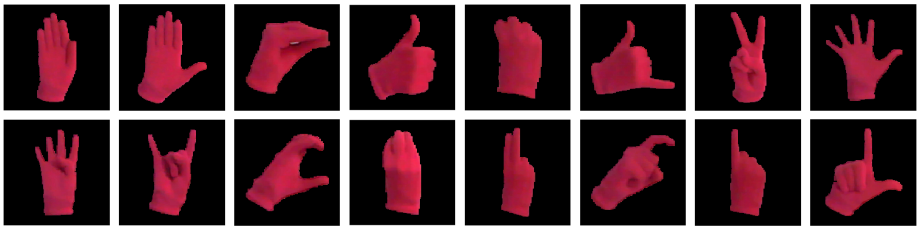


Figura 1. Ejemplos de cada clase de la base de datos LSA16

[1] Se puede encontrar más información sobre esta base de datos en http://facundoq.github.io/unlp/lsa16/.

Los sujetos vistieron ropa negra, sobre un fondo blanco con iluminación controlada, como se observa en la figura 2. Para la simplificar el problema de segmentación de la mano dentro de una imagen, los sujetos utilizaron guantes de tela con colores fluorescentes en sus manos. Esto resuelve parcialmente pero de un modo muy eficaz el reconocimiento de la posición de la mano y carece de los problemas existentes en los modelos de piel. Por otro lado, propone un artefacto simple y económico al momento de realizar pruebas o confeccionar una aplicación real.

Figura 2. Imágenes no segmentadas de la base de datos LSA16

2.2. Preprocesamiento y Descriptores

A continuación se detalla el preprocesamiento realizado de las imágenes de manos segmentadas, los descriptores calculados en base a la imagen preprocesada y el contorno de la mano calculado en base a la misma, y el modelo de clasificación presentado. La entrada a la etapa de preprocesamiento consiste de una imagen donde los únicos pixeles no negros corresponden a la mano.

Preprocesamiento Para cada imagen, y en base a la componente conexa única determinada por la máscara de segmentación de la mano, se calculan los ejes principales de los pixeles de la mano y con ellos la inclinación ϕ de la misma. Luego, se rota la imagen $-\phi^{o}$ para llevarla a una orientación canónica. Como esta orientación es insensible a rotaciones de 180^{o} de la mano, puede que la imagen quede orientada hacia arriba o abajo. Para corregir esto, se calcula la cantidad de cruces de cada línea horizontal posible en la imagen, y se estima la posición de los dedos en base si la moda de la cantidad de cruces se encuentra en la parte superior o inferior de la imagen.

La imagen se re-samplea sin afectar su relación de aspecto a una tamaño de 128×128 y se re-posiciona de modo de que la misma quede centrada. El contorno de la mano se obtiene aplicando un filtro de bordes a la máscara de segmentación de la mano, la cual contiene una sola componente conexa.

Descriptores A continuación se describen dos descriptores, uno basado en la transformada de Radon, y otro en los Scale-Invariant Feature Transform (SIFT).

Figura 3. De izquierda a derecha: Imagen segmentada, imagen orientada, imagen con rotación corregida, máscara de segmentación y contorno.

Transformada de Radon La transformada de Radon ha sido utilizada en el pasado para reconocer objetos y también para identificar a personas en base a las características de su mano [4].

La transformada de Radon de una imagen 2D $f : \mathbb{R}^2 \to \mathbb{R}$ se define como una integral de línea sobre la imagen. La línea L a través de la cual se integra está dada por un par (b, θ), donde b es distancia al origen de la línea y θ el ángulo con el eje horizontal de la imagen. Está dada por la fórmula:

$$R_{(b,\theta)} = \iint\limits_{L_{(b,\theta)}} f(\boldsymbol{x})|d\boldsymbol{x}| = \int_{-\infty}^{\infty} f(x(t), y(t))dt$$
$$= \int_{-\infty}^{\infty} f(tsin\theta + bcos\theta, -tcos\theta + bsin\theta)dt$$

Aplicando la versión discreta de la misma a la imagen segmentada para todas las combinaciones de valores enteros de (b, θ) posibles (1.,180 para θ, un valor K dependiente del tamaño de la imagen para b), obtenemos un descriptor $R \in \mathbb{R}^{180\times K}$. Luego para reducir la dimensionalidad r se re-samplea a un tamaño fijo $r \in \mathbb{R}^{32\times 32}$. Este descriptor se utiliza como global considerándolo un vector en $r' \in \mathbb{R}^{32^2}$, o como 32 descriptores locales tomando cada fila $r_i,\ i = 1, \ldots, 32,\ r_i \in \mathbb{R}^{32}$ como un descriptor local. Cada r_i entonces contiene una aproximación suave a los $R_{(b,\theta)}$ para todo b, y donde θ corresponde aproximadamente la media de un subconjunto de ángulos contiguos.

En particular, como el clasificador que se presenta, el ProbSom, tiene como entrada un conjunto de cardinalidad arbitraria de vectores, se utilizaron los vectores r_i para el mismo, y el vector completo r' para el resto de los clasificadores probados.

SIFT Un descriptor SIFT es un histograma espacial 3D de los gradientes de una imagen, que caracteriza la apariencia de un punto de interés. Para ello, con el gradiente de cada pixel se calcula un descriptor más elemental formado por la ubicación del pixel y la orientación del gradiente. Dado un posible punto de interés, estos descriptores elementales son pesados por la norma del gradiente y acumulados en un histograma 3D que representa el descriptor SIFT de la región alrededor del punto de interés. Al formar el histograma, se le aplica a

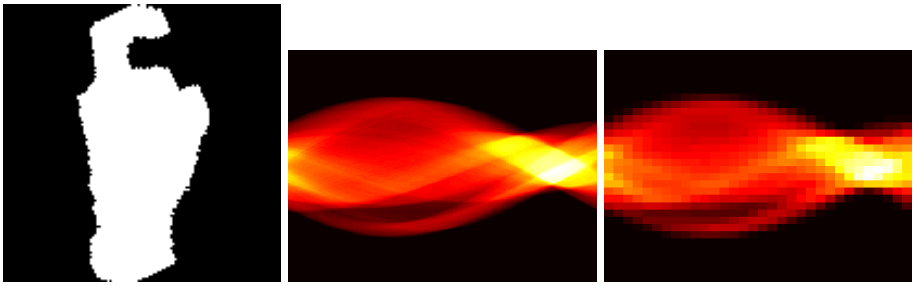


Figura 4. Imagen original, transformada de Radon, transformada de Radon resampleada

los descriptores elementales una función de peso gaussiana para darle menos importancia a los gradientes que están más lejos del centro punto de interés.

Los descriptores SIFT han sido aplicados a varias tareas de visión por computadoras, incluyendo el reconocimiento de configuraciones de mano [13] y reconocimiento de rostros [7].

2.3. Modelo de clasificación ProbSom

ProbSOM [3] es una adaptación probabilística de los mapas auto-organizados de Kohonen(SOM)[6]. Estos mapas son redes competitivas no supervisadas que configuran sus neuronas para representar la distribución de los datos de entrada procesados durante la fase de entrenamiento. Como resultado de esta fase de aprendizaje se obtiene una red donde cada neurona aprende a representar un área del espacio de entrada donde agrupa vectores de datos por su similitud o cercanía.

El proceso de entrenamiento del ProbSOM se realiza de la misma manera que en el algoritmo SOM convencional. ProbSOM agrega una etapa adicional luego del entrenamiento para pesar la proporción de representación de cada neurona. Para ello se repasan todos los patrones de entrada y se agrega a cada una de las neuronas ganadoras información acerca de la clase que representa y en que proporción.

El proceso de reconocimiento también es similar al SOM. El mecanismo de respuesta que decide la identificación de una clase consiste en un sistema probabilista. Como cada vector no permite por si solo la identificación de una clase, una secuencia de vectores es requerida. Cuando un conjunto de vectores de características son introducidos en la red, se obtiene un conjunto de neuronas ganadoras donde cada una representa a varias clases con una proporción determinada. La clase identificada será aquella cuya suma de proporciones sea máxima.

ProbSOM ha demostrado ser un algoritmo robusto para resolver problemas de clasificación [7,3,11] donde las clases se representan por un conjunto de vectores de características, donde dichas clases pueden tener en común vectores muy similares dentro de este conjunto.

3. Resultados

3.1. Metodología y Resultados

A continuación se compara la performance resultante de las pruebas llevadas a cabo con distintos métodos y descriptores. En el caso del ProbSom, se realizaron pruebas con los descriptores SIFT y Radon. Además, para el descriptor basado en Radon, se realizaron pruebas con los modelos estándar del estado del arte Máquinas de Soporte Vectorial (SVM), Random Forest, y Feedforward Neural Networks. [2]. En los casos en que los métodos a comparar se comportan con distinta performance dependiendo de sus parámetros internos, reportamos la mejor.

Método	Performance CV
ProbSom con Radon	92,3(±2,05)
ProbSom con SIFT	88,7(±2,50)
Random Forest con Radon	91,0(±1,91)
SVM con Radon	91,2(±1,69)
Feedforward Neural Net con Radon	78,8(±3,80)

Cuadro 1. Porcentajes de reconocimiento correcto de CV para la base de datos LSA16 utilizando validación cruzada aleatoria.

La medida de performance es el porcentaje de ejemplos reconocidos correctamente sobre el total de cada clase. La tabla 3.1 muestra los resultados obtenidos bajo validación cruzada aleatoria estratificada con $n = 30$ repeticiones independientes, utilizando 90 % de las imágenes para entrenar y 10 % para evaluar. Los resultados muestran una performance comparable del ProbSOM frente a otras técnicas de clasificación. Por otro lado, los descriptores de radón mostraron ser mucho más representativos que los vectores SIFT. Esto puede deberse a que generalmente los descriptores SIFT buscan puntos con información particular, para luego realizar *matching* de imágenes, o describir una situación particular. En las imágenes de LSA16 existen diversos puntos muy similares (como las puntas de los dedos) que resultan comunes a muchas clases, lo que dificultad la utilización de SIFT como se había utilizado en [7] para reconocer rostros, utilizando el mismo modelo de clasificación.

Validación inter-sujeto Utilizando la mejor configuración obtenida (descriptor Radon y ProbSOM) se llevó a cabo una validación cruzada inter-sujeto, dejando un sujeto para testeo y entrenando con el resto. La media de los 10 sujetos con $n = 30$ repeticiones independientes fue de 87,9 %(±4,7 %). Como es de esperar, al dejar un sujeto fuera, la tasa de acierto decae, ya que cada persona realiza las configuraciones de forma particular, con tamaños y apariencia de mano propia

[2] Se realizaron además pruebas con descriptores de Fourier, Banco de filtros de Gabor, Local Binary Patterns (no descriptos en este artículo) con resultados inferiores en casi todos los casos a los presentados.

del individuo. No obstante, el sistema sigue mostrando buenos resultados, dando como posibilidad el reconocimiento correcto de una configuración realizada por un nuevo individuo desconocido por el sistema. La figura 5 muestra los resultados obtenidos para cada individuo de la base de datos.

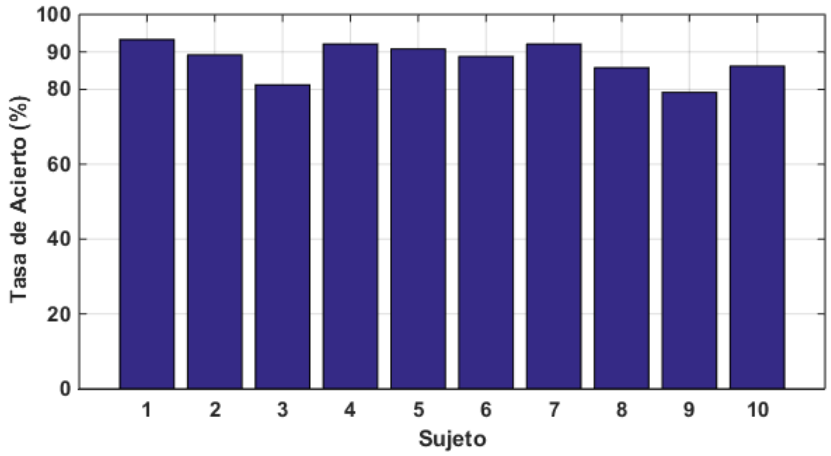


Figura 5. Validación cruzada inter-sujeto para LSA16.

3.2. Discusión

Los descriptores utilizados junto con el modelo de clasificación mostraron ser robustos en la clasificación de las configuraciones de manos en LSA16, incluso con una validación inter-sujeto, dando la posibilidad de incorporar un nuevo individuo desconocido por el sistema. Por otro lado, cabe destacar que la tasa de acierto es similar en todas las clases de la base de datos.

Ya que el ProbSOM funciona de modo probabilístico realizando un ranking de posibles clases candidatas, resulta interesante observar qué ocurre con las imágenes clasificadas erróneamente por el sistema. Si se observa el orden generado por el modelo y la tasa de acierto se obtiene considerando como clasificación correcta tanto a la primer o a la segunda opción, la tasa de acierto general sube de 92,25 % a 96,6 %. Esto demuestra que el modelo, en casi todos los ejemplos de testeo la confusión es entre sólo dos clases. Esto resulta muy interesante si el modelo funciona como un diccionario, ya que podría utilizarse la probabilidad del modelo para mostrar una o dos posibilidades. Del mismo modo, podría volverse a aplicar un clasificador más específico para solucionar la ambigüedad en las situaciones que lo requieran.

4. Conclusión

En este trabajo se presenta una base de datos de configuraciones de manos para el Lenguaje de Señas Argentino (LSA), junto con un modelo de preprocesamiento de las imágenes y clasificación de las configuraciones.

Los resultados de los experimentos de clasificación fueron favorables, mostrando una alta tasa de acierto tanto en la validación aleatoria como en la intersujeto. También se llevaron a cabo comparaciones con diferentes descriptores y métodos de clasificación existentes.

El modelo presentado permite la correcta clasificación de las configuraciones de manos, dando la posibilidad de utilizar esto para generar una sub-unidad léxica parte de un descriptor general para una seña de LSA. Se espera también probar la técnica en otras bases de datos existentes en el estado del arte para determinar su aplicabilidad, así como extenderla para utilizar también imágenes de sensores de profundidad.

Referencias

1. Cooper, H., Holt, B., Bowden, R.: Sign language recognition. In: Moeslund, T.B., Hilton, A., Krüger, V., Sigal, L. (eds.) Visual Analysis of Humans: Looking at People, chap. 27, pp. 539 – 562. Springer (Oct 2011), http://www.springer.com/computer/image+processing/book/978-0-85729-996-3
2. Cooper, H., Ong, E.J., Pugeault, N., Bowden, R.: Sign language recognition using sub-units. Journal of Machine Learning Research 13, 2205–2231 (Jul 2012), http://jmlr.csail.mit.edu/papers/volume13/cooper12a/cooper12a.pdf
3. Estrebou, C., Lanzarini, L., Hasperue, W.: Voice recognition based on probabilistic SOM. In: Latinamerican Informatics Conference. CLEI 2010. Paraguay. October 2010. (2010)
4. Gangopadhyay, A., Chatterjee, O., Chatterjee, A.: Hand shape based biometric authentication system using radon transform and collaborative representation based classification. In: Image Information Processing (ICIIP), 2013 IEEE Second International Conference on. pp. 635–639 (Dec 2013)
5. Kadir, T., Bowden, R., Ong, E.J., Zisserman, A.: Minimal training, large lexicon, unconstrained sign language recognition. In: British Machine Vision Conference (2004)
6. Kohonen, T.: Self-organizing formation of topologically correct feature maps. Biological Cybernetics 43(1), 59–69 (1982)
7. Lanzarini, L., Ronchetti, F., Estrebou, C., Lens, L., Fernandez Bariviera, A.: Face recognition based on fuzzy probabilistic SOM. In: IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), 2013 Joint. pp. 310–314. IEEE (2013)
8. Pugeault, N., Bowden, R.: Spelling it out: Real-time ASL fingerspelling recognition. In: 1st IEEE Workshop on Consumers Depth Cameras for Computer Vision, in conjunction with ICCV'2011 (2011), http://info.ee.surrey.ac.uk/Personal/N.Pugeault/publications/PugeaultBowden2011b.pdf
9. Rioux-Maldague, L., Giguere, P.: Sign language fingerspelling classification from depth and color images using a deep belief network. In: Computer and Robot Vision (CRV), 2014 Canadian Conference on. pp. 92–97. IEEE (2014)

10. Roussos, A., Theodorakis, S., Pitsikalis, V., Maragos, P.: Hand tracking and affine shape-appearance handshape sub-units in continuous sign language recognition. In: Trends and Topics in Computer Vision - ECCV 2010 Workshops, Heraklion, Crete, Greece, September 10-11, 2010, Revised Selected Papers, Part I. pp. 258–272 (2010), http://dx.doi.org/10.1007/978-3-642-35749-7_20
11. Villamonte, A., Quiroga, F., Ronchetti, F., Estrebou, C., Lanzarini, L., Estelrrich, P., Estelrrich, C., Giannechini, R.: A support system for the diagnosis of balance pathologies. In: Congreso Argentino de Ciencias de la Computación. CACIC 2014. Argentina. October 2014. (2014)
12. Zhang, C., Yang, X., Tian, Y.: Histogram of 3d facets: A characteristic descriptor for hand gesture recognition. In: Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on. pp. 1–8. IEEE (2013)
13. Zhu, X., Wong, K.K.: Single-frame hand gesture recognition using color and depth kernel descriptors. In: Pattern Recognition (ICPR), 2012 21st International Conference on. pp. 2989–2992. IEEE (2012)

XV Workshop de Procesamiento Distribuido y Paralelo (WPDP)

7403 | Performance Improvements of an Atmospheric Radiative Transfer Model on GPU-based platform using CUDA
Jacobo Salvador (UNPA), Sofía Osiris (UNPA), Facuando Orte, Eder Dos Santos (UNPA), Hirofumi Oyama, Tomoo Nagahama, Akira Mizuno, Roberto Uribe-Paredes (UMAG)

7464 | Comparación de rendimiento de algoritmos de cómputo intensivo y de acceso intensivo a memoria sobre arquitecturas multicore. Aplicación al algoritmo de criptografía AES
Adrián Pousa (UNLP), Victoria Sanz (UNLP), Armando E. De Giusti (UNLP)

7596 | Análisis del impacto de distintas técnicas de optimización de rendimiento en multicore
Matías Dell Oso (UNLP), Juan Manuel Paniego (UNLP), Martin Pi Puig (UNLP), Marcelo Naiouf (UNLP), Armando E. De Giusti (UNLP)

7597 | Caracterización de una estrategia de detección de fallos transitorios en HPC
Diego Montezanti (UNLP), Dolores Rexachs (UAB), Enzo Rucci (UNLP), Emilio LUQUE (UAB), Marcelo Naiouf (UNLP), Armando E. De Giusti (UNLP)

7606 | Including accurate user estimates in HPC schedulers : an empirical analysis
Néstor Rochetti (UdelaR), Santiago Iturriaga (UdelaR), Sergio Nesmachnow (UdelaR)

7613 | Assessing metric structures on GPGPU environments
Eder Dos Santos (UNPA), Osiris Sofía (UNPA), Roberto Uribe-Paredes (UMAG)

7741 | Conversion de RGB a YCbCr en System Generator y HLS
Romina Molina (UNSL), Veronica Gil-Costa(UNSL)

Performance Improvements of an Atmospheric Radiative Transfer Model on GPU-based platform using CUDA

Jacobo Salvador[1,3], Osiris Sofia[1], Facundo Orte[3], Eder Dos Santos[1], Hirofumi Oyama[4], Tomoo Nagahama[4],Akira Mizuno[4], Roberto Uribe-Paredes[2].

Contact email: jacosalvador@gmail.com

[1.] Unidad Académica Río Gallegos – Universidad Nacional de la Patagonia Austral
Lisandro La Torre 1070 – 0054-2966-442313 – Río Gallegos – Santa Cruz – Argentina
[2.] Departamento de Ingeniería en Computación, Universidad de Magallanes, Chile.
[3.] *Centro de Investigaciones en Láseres y Aplicaciones, CEILAP UMI-IFAECI-CNRS-3351,UMI3351, Villa Martelli, Argentina*
[4.]*Solar-Terrestrial Environment Laboratory (STEL), Nagoya University, Nagoya, Japan.*

Abstract: Classical applications of Atmospheric Radiative Transfer Model (ARTM) for modelization of absorption coefficient line-by-line on the atmosphere consume large computational time since seconds up to a few minutes depending on the atmospheric characterization chosen. ARTM is used together with Ground-Based or Satellite measurements to retrieve atmospheric parameters such as ozone, water vapour and temperature profiles. Nowadays in the Atmospheric Observatory of Southern Patagonia (OAPA) at the Patagonian City of Río Gallegos have been deployed a Spectral Millimeter Wave Radiometer belonging Nagoya Univ. (Japan) with the aim of retrieve stratospheric ozone profiles between 20-80 Km. Around 2 GBytes of data are recorder by the instrument per day and the ozone profiles are retrieving using one hour integration spectral data, resulting at 24 profiles per day. Actually the data reduction is performed by Laser and Application Research Center (CEILAP) group using the Matlab package ARTS/QPACK2. Using the classical data reduction procedure, the computational time estimated per profile is between 4-5 minutes determined mainly by the computational time of the ARTM and matrix operations. We propose in this work first add a novel scheme to accelerate the processing speed of the ARTM using the powerful multi-threading setup of GPGPU based at *Compute Unified Device Architecture (CUDA)* and compare it with the existing schemes. Performance of the ARTM has been calculated using various settings applied on a NVIDIA graphic Card GeForce GTX 560 Compute Capability 2.1. Comparison of the execution time between sequential mode, Open-MP and CUDA has been tested in this paper.

Keywords: Radiative transfer model, GPU, OpenMP, Parallel Computing, CUDA.

1. Introduction

Computational speed of Graphical Processing Units (GPU) devices has exploded in recent years. Research studies in this field have indicated that the GPU is two order of magnitudes faster than typical Central Processing Unit (CPU) for many applications. However, such performance figures have also been shown as unfairly biased [Chhugani et al., 2010]. It is important to compare similar CPU and GPU setups, as well as utilizing all possible optimizations on both the GPU and the CPU [Gregg and Kim Hazelwood, 2011].

The current version of the atmospheric model called *am* is an Atmospheric Radiative Transfer model which is written in C and runs on Linux environment machines from desktops PC. Previous work have shown that fast radiative transfer model (RTM) is very suitable for the GPU implementation as it can take advantage of the hardware's efficiency and parallelism, where radiances of many channels can be calculated in parallel in the GPU [Mielikainen., et al 2011]. Previously, GPUs have also been used very successfully to accelerate Cooperative Institute for Meteorological Satellite Studies (CIMSS) RTM (Huang et al., 2010). It might be in the next years a novel tool for computational calculation applied to the whole field of science.

1.1 Atmospheric Radiative Transfer Model

The propagation of radiation through the atmosphere is described by the radiative transfer theory [Chandrasekhar, 1960]. This theory describes how the variation of the radiation intensity along a line of slight *ds* following the propagation direction is found considering the sum of source and absorption within a unit volume element along this line. The expression can be written as a differential equation [Janssen, 1993]:

$$\frac{dI_\upsilon}{ds} = -I_\upsilon \alpha + S \tag{1}$$

Where α is the absorption coefficient, S the Source term, which respectively describe the loss and gain of energy into a given direction and Iv , is the instantaneous radiant power that flows at each point in the medium, per unit area, per unit-frequency interval at a specific frequency, and in a given direction per unit solid angle.

Scattering can be generally neglected in the microwave range, due to the fact the corresponding wavelengths are several orders of magnitude larger than the cloud droplet radius and therefore the absorption cross sections are much larger than the scattering cross-sections. In addition microwave wavelengths are insensitive to scattering by stratospheric aerosols, having a cross-sections up to 1000 times smaller than the wavelength considered. Under rainy-weather conditions the droplets can be reach a size comparable with the wavelength observed. In this case the scattering cannot be neglected.

In most cases for ground-based remote sensing of stratospheric and mesospheric quantities, the scattering effects due to the rain droplet can be neglected if we assume that optical-path length due to the scattering is small along beam axis and the scattering medium (rain droplets) distribution in a plane perpendicular to the beam axis is isotropic.

Hence without scattering the source S represents only the locally generated contribution to the radiation , and α the loss radiation energy into the propagation medium [Janssen, 1993]. In particular if we assume that each point along a path has local thermodynamic equilibrium so each point can be characterize by a temperature T. The balance between the energy absorbed and emitted by a particular volume element can be characterized by Kirchoff's law for the source term as follow:

$$S = \alpha B_\upsilon(T) \tag{2}$$

where $B\nu(T)$ is the Planck function:

$$B_\upsilon(T) = \frac{2h\upsilon^3}{c^2}\frac{1}{e^{\frac{h\upsilon}{\kappa T}} - 1} \tag{3}$$

Here h is the Planck's constant, k is the Boltzmann's constant, c is the speed of light, and v is the frequency. Therefore, combining equation (1) and (3) and assuming that the scattering is negligible, the differential equation can be solved as follow:

$$I_\upsilon(0) = I_\upsilon(s_0)e^{-\tau(s_0)} + \int_0^{s_0} B_\upsilon(T)e^{-\tau(s)}\alpha ds \tag{4}$$

where τ is optical depth and is defined as:

$$\tau(s) = \int_0^s \alpha(s')ds' \tag{5}$$

The radiative atmospheric model selected to this work is the "*am"* (atmospheric model) which could be downloaded from Harvard Smithsonian Center for Astrophysics CFA at https://www.cfa.harvard.edu/~spaine/am/.

The *am* model works from microwave through submillimeter wavelengths. The primary application area for *am* is in radio astronomy, involving terrestrial propagation paths between ground and sky but also can be expanded to ground-based instrumentation. The propagation paths can be modeled following a sequence of planar layers where each layer might be composed with different gas tracer named as columns. A representation of the atmospheric state is given in Figure 1.

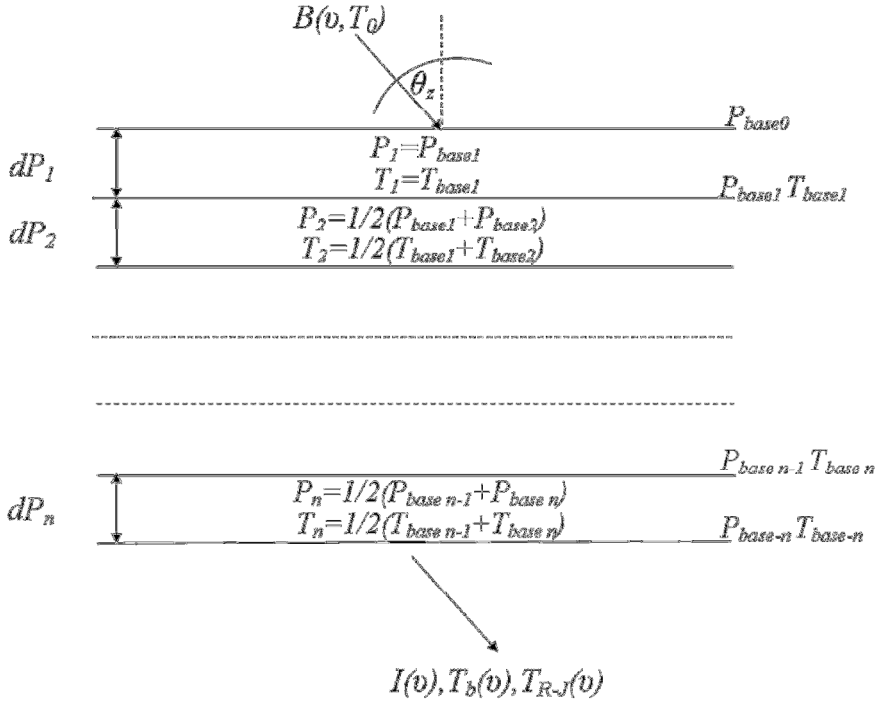


Figure 1: am layer model. Layer pressures and temperatures have been defined at the layer midpoints as P_n=1/2($Pbase_n$-1+$Pbase_n$). Adapted from *am* model manual [Paine., 2014].

Pn and *Tn* represents the mid-point pressure and temperature of the layer *n* respectively. $B(\upsilon,T_0)$ is the Plank function corresponding to background at top. Brightness temperature at surface is the sum of the contribution of each layer.

1.1.1 Application on Instrument

Atmospheric models at millimeter and submillimeter regions are useful tools in data reduction for remote sensing applications of ground-based and satellite techniques. The spectral millimeter wave measurements use molecular emission lines which are due to pure rotation transitions of molecules. Therefore, we can measure the emission spectra at day and night because they do not require any light sources unlike absorption and scattering measurements. This is a great advantage compared with other methods, such as optical and/or infrared measurements [Jenssen, 1993]. Continuous measurements provide us

information on variations of the vertical distributions of gases(ozone in our case) in the middle atmosphere with various timescales. The vertical profile of ozone mixing ratio can be calculated exactly from Eq. 4 from the radiative transfer at the millimeter wavelength.
A spectral millimeter wave radiometer (MWR) is basically composed of an antenna, a heterodyne receiver front end to process the spectral line signal emitted by the ozone at ~110.836 GHz, and a spectrometer backend. Figure 2 shows a block diagram of the MWR installed in the Observatorio Atmosférico de la Patagonia Austral (OAPA).
The flat rotating mirror installed before the parabolic mirror switches the input signal among the sky and the two calibration loads. The intensity scale is calibrated by using two temperatures blackbodies, hot and cold loads, as intensity references. The hot load (~300 K) is a radio-wave absorber at room temperature and the cold load (77 K) is achieved through an absorber by soaking in liquid nitrogen in a styrene foam vessel.

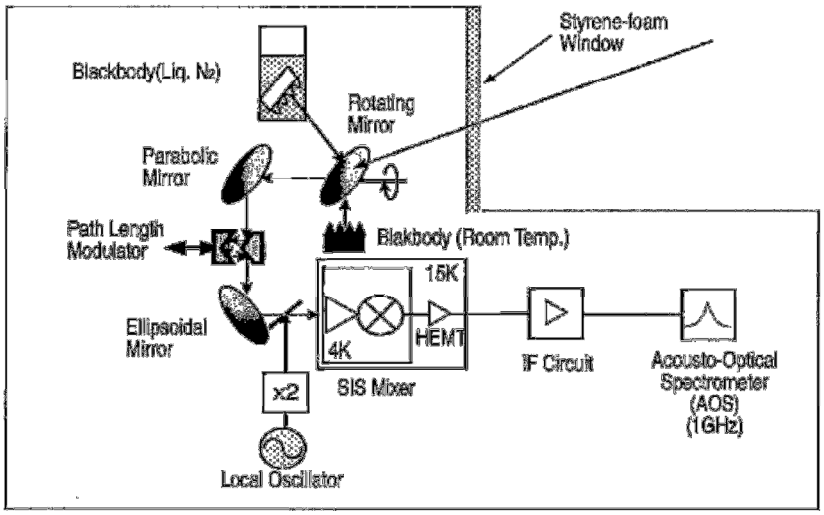


Figure 2. Block diagram of millimetric wave radiometer installed in the Observatorio Atmosferico de la Patagonia Austral used to measure ozone.

1.2 Accelerating based in OpenMP and Threshold Value

The OpenMP Application Program Interface (API) is a multi-platform shared-memory parallel programming model for the C, C++ and Fortran programming languages. Further information can be found at http://www.openmp.org/. When the model runs with high spectral resolution and with many layers and columns, it can take long computational time. The atmospheric model *am* used in the work was implemented using the OpenMP version applied on multi-core machines due the line-by-line absorption coefficient computations are easily parallelized. A second improvement to speed-up line-by-line computation consists in to stop the calculations when the relative error of the absorption coefficient is lower than a threshold value. This is set in the input file through a parameter called Tolerance and called as Tol in the next sections and graphics.
A third method of accelerating computations consists in set up the disk cache, creating a temporary directory where is contained absorption coefficient recently used. If we want to run again a determined input file with a little changes of the input file, then the model compare the actual with the previous calculation stored in the temporary file avoiding extra time.

2. Graphics Processing Units

The era of single-threaded processor performance increases has come to an end. Programs will only increase in performance if they utilize parallelism. However, there are different kinds of parallelism. For instance, multicore CPUs provide task-level parallelism. On the other hand, Graphics Processing Units (GPUs) provide data-level parallelism.

Current GPUs consist of a high number (up to 512 in current devices) of computing cores and high memory bandwidth. Thus, GPUs offer a new opportunity to obtain short execution times. They can offer 10x higher main memory bandwidth and use data parallelism to achieve up to 10x more floating point throughput than the CPUs [Wu-Feng and D.Manocha, 2007].

GPUs are traditionally used for interactive applications, and are designed to achieve high rasterization performance. However, their characteristics have led to the opportunity to other more general applications to be accelerated in GPU-based platforms. This trend is called General Purpose Computing on GPU (GPGPU). These general applications must have parallel characteristics and an intense computational load to obtain a good performance. To assist in the programming tasks of these devices, the GPU manufacturers, like NVIDIA or AMD/ATI, have proposed new languages or even extensions for the most common used high level programming languages. As example, NVIDIA proposes CUDA [NVIDIA CUDA], which is a software platform for massively parallel highperformance computing on the company powerful GPUs.

In CUDA, the calculations are distributed in a mesh or grid of thread blocks, each with the same size (number of threads). These threads run the GPU code, known as kernel. The dimensions of the mesh and thread blocks should be carefully chosen for maximum performance based on the specific problem being treated.

Current GPUs are being used for solving different problems like data mining, robotics, visual inspection, video conferencing, video-on-demand, image databases, data visualization, medical imaging, among other and it is increasingly the number of applications that are being parallelized for GPUs.

2.1 CUDA Programming Model

The NVIDIA's CUDA Programming Model [NVIDIA CUDA Compute Unified Device Architecture-Programming, 2009]considers the GPU as a computational device capable to execute a high number of parallel threads. CUDA includes C/C++ software development tools, function libraries, and a hardware abstraction mechanism that hides the GPU hardware to the developers by means of an Application Programming Interface (API). Among the main tasks to be done in CUDA are the following: allocate data on the GPU, transfer data between the GPU and the CPU and launch kernels.

A CUDA kernel executes a sequential code in a large number of threads in parallel. The threads within a block can work together efficiently exchanging data via a local shared memory and synchronize low-latency execution through synchronization barriers (where threads in a block are suspended until they all reach the synchronization point). By contrast, the threads of different blocks in the same grid can only coordinate their implementation through a high-latency accesses to global memory (the graphic board memory). Within limits, the programmer specifies how many blocks and the number of threads per block that are allocated to the implementation of a given kernel.

3. Atmospheric Radiative Tranfer Model on GPU

There are currently very low cost devices that allow the massive GPU based parallelization. Due to the large amount of data processing one intends efficient GPU parallelization in this section will be implemented to speed-up the execution time of the program. We will describe an implementation on GPU of line-by-line calculation over the *am* model [Paine., 2014]. The line-by-line calculations are implemented on the linesum function as part of *am* model. Previous tests over the program have shown that around 80% of the total execution time of the model is focusing in the linesum routine. The aim of the routine consists in produce the addition of the absorption coefficient in each layer in the whole spectral range set. It is important to remark that it is not our intention of this section to give a full description of physics aspects involved in general atmospheric models.

The line-by-line column types set at the input file are important parameter which are listed in Table B.4 [manual *am*]. These column types are for molecular species having line spectra associated with a molecular absorption coefficient, k [cm^2]. The absorption coefficient is computed using data from the HITRAN 2012 database (Rothman et al. 2013). For each column type, a default line shape is given. In our case we have implemented on GPU only the Gross and VVH_750 line shape function because they are defined majority by default in the *am* model. Default line shapes may be altered by layer and column type by using the lineshape keyword in the input dataset.

As described earlier, the GPU holds different types of memory. To best utilize the GPU the developer must understand these differences of memory types. Depending on their usage the memory can affect the performance of the whole application. The implementation using CUDA involves a basic kernel scheme using shared memory, single floating point and math built-in device function. Figure 3 shows as the linesum routine was modified to run on CUDA. Different variables used by the model need firstly to be transferred from the host to device memory using built-in functions provided by CUDA. After that we define and create local variables only visible by the thread. Our implementation uses shared memory due to its faster access than global memory.

The longitude of the blocks and thread per blocks depends strongly on the grid frequency. It is the main variable that regulates how many threads and blocks will be used for the kernel function. This variable is not fixed and depends on the input parameters that the user set at the input file. In our experiments described in the next section we have chosen the case of the thermal emission of the ozone molecule at 110.836 GHz with a bandwidth around the central frequency of 1GHz.

Frequency resolution value called in this paper as Δf will define the length of the grid frequency. In our experiments four different Δf values are chosen at 63, 30, 15 and 7 KHz.

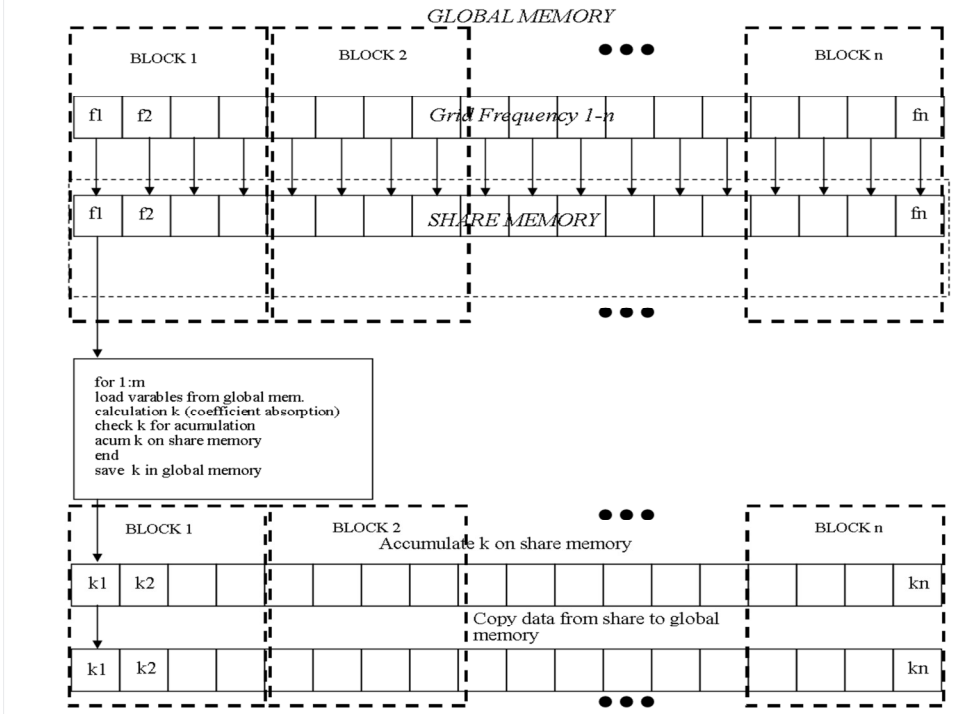


Figure 3: Scheme of the CUDA implementation in the GPU for the linesum routine at AM model. Each thread is the responsible of compute the absorption coefficient at a determined frequency. The frequency grid are organized by a grid composed by blocks and each block for a constant number of threads.

4. Results and discussion

The experiments were carried out in a PC Intel Core i7-4770 at 3.40GHz with 8Gb RAM with a graphical board GeForce GTX 560 with a total amount of global memory of 1Gb, 48 CUDA cores by Multiprocessors (eight in total) using CUDA driver 6.0, capability 2.1.

Three types of program versions are evaluated in this section as well as their performances in respect to the total time of execution classified as sequential, CUDA version and original model. In the sequential mode or commonly called force brute the original version of the program was modified to compile without OpenMP directives in all routines. For more details of CUDA version refers section above where the CUDA implementation inside the linesum function of the program was modified.

A thread is the fundamental building block of a parallel program. Most C programmers are familiar with the multicore programming. Even if you have never launched a thread in any code, you will be familiar with executing at least one thread, the single thread of execution through any serial piece of code. In GPU we launch many threads per block at the same time through a function called kernel executing just in the graphic board. Often the value of the threads by blocks are set in 1024 that is the máximum value but also we could choose another value as 512, 256, 128, 64 and 32 threads. For our implementation in CUDA we have performed the model four times taking different numbers of threads per block, tolerance and resolution. The determination of the optimal threads number was determined taken into account the results of figure 4.

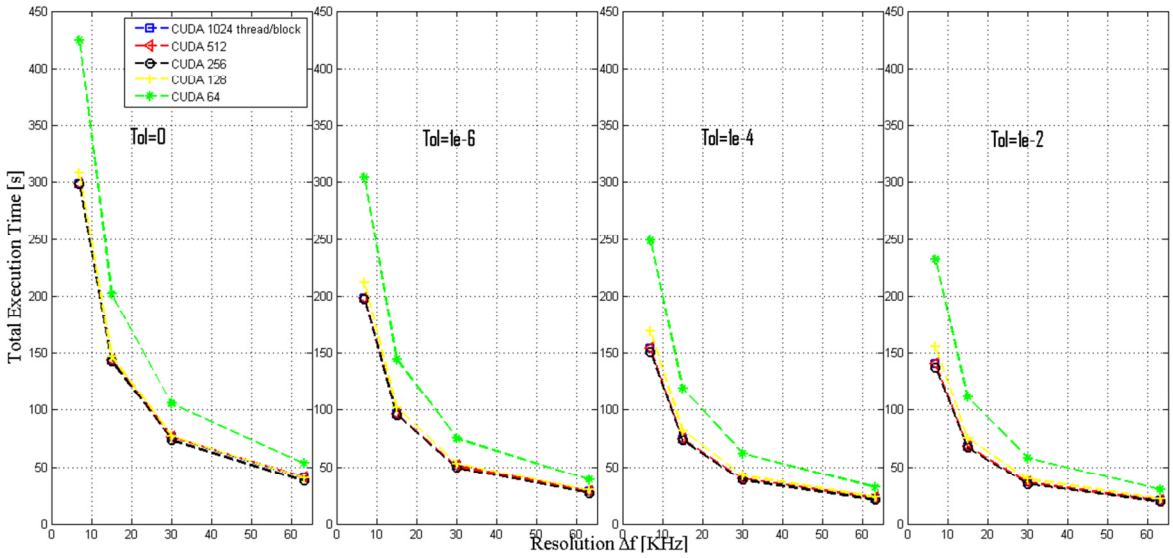


Figure 4: Evaluation of the performance of the CUDA implementation routine respect the total number of thread by block for 1024, 512, 256, 128,64 for 4 tolerance values.

We have used 256 threads per block in our graphical board, because this represents the optimal value that reduces the execution time of our code. Also, it is possible to see how the performance of the experiments is getting worse when the order of the threads is lower than 128 threads per block. NVCC compiler is used to compile the package. Respect the original version, it is used without any modification in its code and it has been installed following the manual instruction.

The models called sequential, CUDA and original versions were configured to run using a basic atmosphere with the number of 60 layers. The model is set with a central grid frequency at 110.836 GHz with 500MHz both sides. It means that the total bandwith is 1 GHz. Here we will focus in the performance and improvements of the codes. Figure 5 present the outputs of the three execution versions described above. The graphics are organized in rows and columns. Each row represents different outputs of a model for a specific resolution and tolerance, while columns represent the execution of the same input file but using different tolerance parameters (0, $1x10^{-6}$,$1x10^{-4}$,$1x10^{-2}$). As was described in details in the previous section, tolerance value is implemented in the model to speed-up the calculation. In the x-axis the number of threads of the Intel Core is set from 1 to 4 as maximum while y-axis represents the total time execution given by the program.

The figure below shows that CUDA implementation (triangle) using tolerance between 0, $1x10^{-6}$, $1x10^{-4}$ has a total execution time lower than sequential and original version. At a tolerance of $1x10^{-2}$ CUDA is much better when the thread number of the CPU is 1 or 2. For higher threads the original implementation shows a best performance.

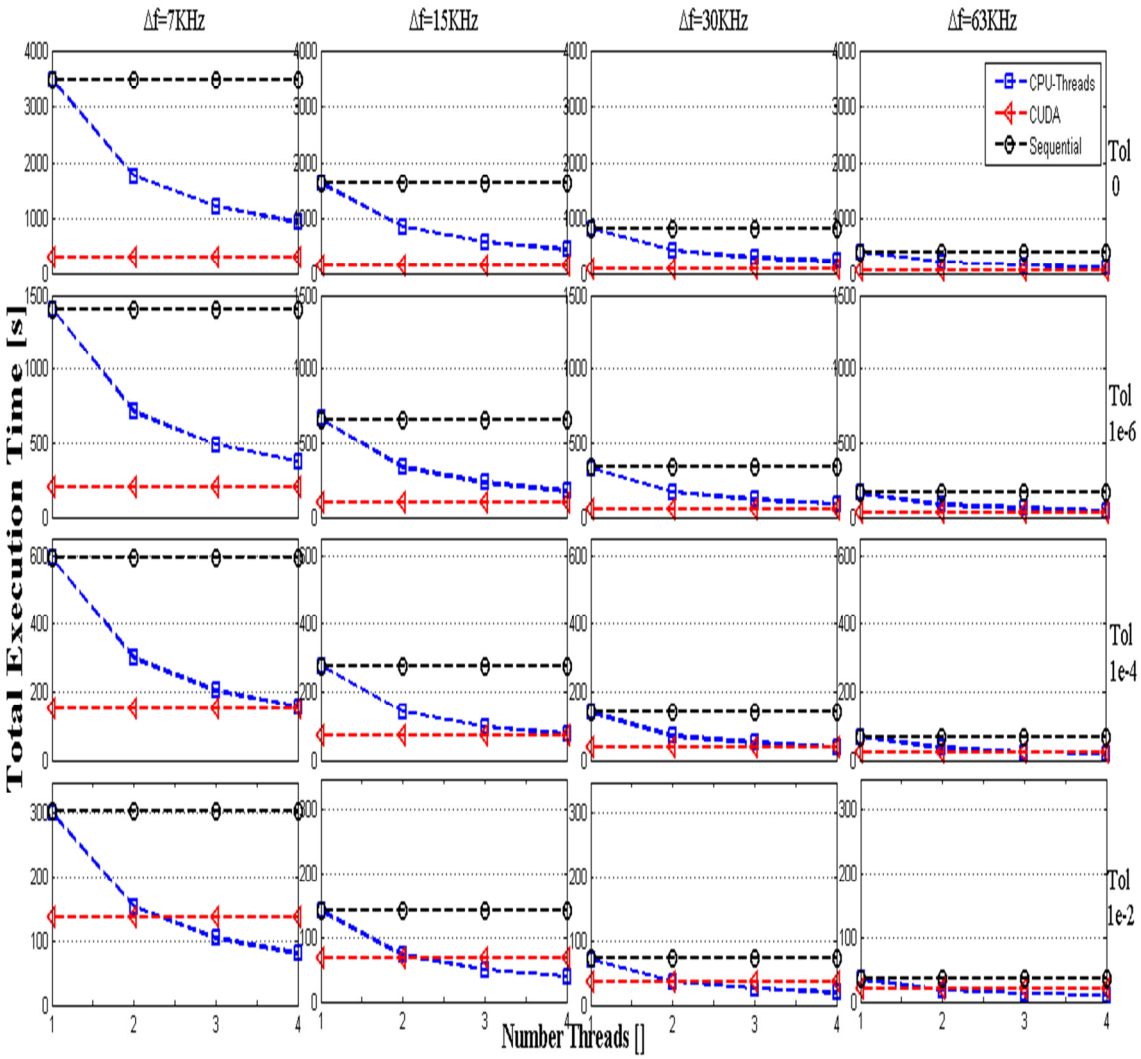


Figure 5: Evolution of the performance in total execution time vs threads numbers (CPU) to four resolutions and tolerance values for sequential (circle), CUDA implementation (triangle) and original program (square).

5. Conclusion and future work

This work presents a different point of view with respect to the optimization of an atmospheric radiative transfer model ARTM for ground-based application. We have presented an introduction to Graphics Processing Unit and an evaluation of the total execution time of the radiative transfer model denominated as *am* downloaded from the web belonging to Harvard-Smithsonian Center for Astrophysics was performed with different settings and compilation. Three comparisons have been presented and tested based in CUDA routines, sequential implementation and also they were compared with the original version working with OpenMP. For CUDA implementation the same concept given by the original model to avoid redundant calculation has been included to accelerate the execution. Another important point to reduce the execution time in CUDA is the tolerance parameter optimization that is described in the manual of *am*. The experiment shows that CUDA implementation is faster when the tolerance parameter is below $1x10^{-4}$ for all resolution frequencies (Fig. 5). An extreme case for example when the tolerance parameter is zero our test shows that for Δf=7KHz the implementation on CUDA is 3.09x

faster than original version with four CPU threads. This performance for the other resolution frequencies decreases a little for Δf=15, 30 and 63KHz reaching a performance of 2.55x for the last resolution at 63 KHz. For tolerance parameter of $1x10^{-6}$ has also been obtained an improvement in CUDA with respect to all threads using the CPU. For tol=$1x10^{-4}$ the speed-up improvement are acceptable when the threads number of the CPU are lower than four.

Finally we can conclude that CUDA implementation might be a useful tool to incorporate area of science to improve the computational time of models or math operations. In the future researching in code optimization based for GPU will continue, and we thought this kind of work might open a new way for general optimization codes.

6. References

- Jatin Chhugani, Michael Deisher, Daehyun Kim, Victor W Lee, Changkyu Kim, and Anthony D. Nguyen. *Debunking the 100X GPU vs. CPU myth: an evaluation of throughput computing on CPU and GPU*. In: Proceedings of the 37th anual international symposium on Computer architecture. ISCA '10. Saint-Malo, France: ACM, pp. 451–460. ISBN: 978-1-4503-0053-7. URL: http://doi.acm.org/10.1145/1815961.1816021. 2010.
- *GPGPU. general-purpose computation using graphics hardware*, http://www.gpgpu.org.
- Chris Gregg and Kim Hazelwood. *Where is the Data? Why you cannot debate CPU vs. GPU Performance without the answer*. Tech. rep. University of Virginia, 2011. URL: http://www.cs.virginia.edu/kim/docs/ispass11.pdf.
- Huang B., Mielikainen J., Oh H. and Huang H.-L, *Development of a GPU-based highperformance radiative transfer model for the Infrared Atmospheric Sounding Interferometer (IASI)*, Journal of computational Physics, DOI:10.1016/j.jcp.2010.09.011.
- Jenssen, M. A., *Atmospherc Remote Sensing by Microwave Radiometry*, A volume in the Wiley Series in Remote Sensing Jin Au Kong, Series Editor, 1993.
- Mielikainen Jarno, Bormin Huang, and Hung-Lung Allen Huang, *GPU-Accelerated Multi-Profile Radiative Transfer Model for the Infrared Atmospheric Sounding Interferometer*, IEEE Journal ofr Selected Topics in Applied Earth Observation and Remote Sensing, Vol. 4, Nº 3, September 2011.
- *NVIDIA CUDA Compute Unified Device Architecture-Programming Guide,* Version 2.3. NVIDIA. http://developer.nvidia.com/object/gpucomputing.html. 2009.
- Paine, S. "The *am* atmospheric model" Submillimeter Array Technical Memo #152, v. 8.0 (2014) URL: http://www.cfa.harvard.edu/sma/memos/152.pdf
- Rothman, L.S, I.E. Gordon, A. Barbe, D.Chris Benner, P.F. Bernath, M. Birk, V. Boudon, L.R. Brown, A. Campargue, J.-P. Champion, K. Chance, L.H. Coudert, V. Dana, V.M. Devi, S. Fally, J.-M. Flaud, R.R. Gamache, A. Goldman, D. Jacquemart, I. Kleiner, N. Lacome, W.J. Lafferty, J.- Y. Mandin, S.T. Massie, S.N. Mikhailenko, C.E. Miller, N. Moazzen-Ahmadi, O.V. Naumenko, A.V. Nikitin, J. Orphal, V.I. Perevalov, A. Perrin, A. Predoi-Cross, C.P. Rinsland, M. Rotger, M. Šimečkova, M.A.H. Smith, K. Sung, S.A. Tashkun, J. Tennyson, R.A. Toth, A.C. Vandaele and J. Vander Auwera 2009, The HITRAN 2008 molecular spectroscopic database, J. Quant. *Spectrosc. Radiat. Transfer* 110:533.
- Subrahmanyan Chandrasekhar. *Radiative Transfer*. Dover Publications Inc. p. 393. ISBN 0-486-60590-6, 1960.
- Wu-Feng, Manocha, D.: *High-performance computing using accelerators*. Parallel Computing **33**, 645–647 2007.

Acknowlegment

We would like wish to National Research Project 29A-274-1 UNPA-UARG by the financial support of this work and Harvard-Smithsonian Center for Astrophysics for the uses of the *am* model, version 8.0.

Comparación de rendimiento de algoritmos de cómputo intensivo y de acceso intensivo a memoria sobre arquitecturas multicore. Aplicación al algoritmo de criptografía AES.

Adrian Pousa[1], Victoria María Sanz[1,2], Armando E. De Giusti[1,2]

[1] III-LIDI Facultad de Informática UNLP La Plata, Argentina
[2] CONICET, Argentina
{apousa,vsanz,degiusti}@lidi.info.unlp.edu.ar

Abstract. En este trabajo se presenta una comparación de rendimiento de dos versiones del algoritmo de criptografía AES. La primera versión, AES-CI (AES Cómputo Intensivo), se caracteriza por ser intensiva en cómputo. La segunda versión, AES-AIM (AES Acceso Intensivo a Memoria), reduce la cantidad de cálculo reemplazando operaciones por acceso a datos pre-calculados almacenados en memoria. Se llevaron a cabo tres implementaciones de cada versión del algoritmo, desarrolladas con diferentes herramientas paralelas (OpenMP, MPI, CUDA), para ser ejecutadas sobre una máquina multicore, un cluster de máquinas multicore y una GPU respectivamente. El trabajo experimental muestra que las implementaciones AES-AIM reducen el tiempo de ejecución con respecto a los obtenidos por las implementaciones AES-CI. Del estudio se observa que las características del algoritmo determinan la arquitectura a utilizar para obtener el mejor rendimiento.

Keywords: Algoritmos de Cómputo Intensivo; Algoritmos de Acceso Intensivo a Memoria; AES; arquitecturas multicore.

1 INTRODUCCIÓN

En general las arquitecturas paralelas (clusters, multicores, GPUs, etc.) [1] [2] se han utilizado para resolver diversos problemas en menor tiempo. Para tal fin, se han desarrollado herramientas tales como MPI [3], OpenMP [4], CUDA [5], etc., que facilitan la programación de algoritmos paralelos [6] [7] y permiten explotar las ventajas de estas arquitecturas.

En los últimos años, las GPUs (Graphics Processing Units) [8] han ganado importancia debido al alto rendimiento que alcanzan al ejecutar aplicaciones de propósito general. Sin embargo, estas arquitecturas presentan algunas limitaciones relacionadas con la memoria: (a) La capacidad de memoria global es de un tamaño fijo y no se puede incrementar, a diferencia de otras arquitecturas (b) Las GPUs poseen una jerarquía de memoria de varios niveles y con distintas características, dependiendo del nivel que se acceda se tendrá menor o mayor latencia. Una técnica

que permite ocultar la latencia consiste en ejecutar una mayor cantidad de hilos, de esta forma, mientras un conjunto de hilos espera que se resuelva un acceso a memoria, otro conjunto de hilos ejecuta en los procesadores. Sin embargo, esto implica tener menor cantidad de recursos por hilo (registros, fracción de memoria compartida que le corresponde a cada hilo).

Por otro lado, el volumen de datos que se trasmiten en las redes se ha incrementado considerablemente, y en ocasiones suelen representar información sensible, por lo tanto es importante codificarlos para enviarlos por una red pública como lo es Internet de manera segura. El encriptado y desencriptado de datos requiere un tiempo de cómputo adicional, que dependiendo de su tamaño puede ser considerable.

AES (Advanced Encryption Standard) es un algoritmo de cifrado simétrico por bloques que se ha convertido en estándar en 2002 [9] y actualmente es ampliamente usado para codificar información. En 2003 el gobierno de los Estados Unidos anunció que el algoritmo era lo suficientemente seguro y que podía ser usado para protección nacional de información [10]. Hasta el momento no se conocen ataques eficientes, los únicos conocidos son los denominados ataques de canal auxiliar [11].

Este algoritmo se caracteriza por ser simple y por consumir pocos recursos. Sin embargo, el tiempo de cifrar y descifrar grandes cantidades de datos es importante, por lo que es adecuado aprovechar las posibilidades que brindan las arquitecturas multicores para reducir este tiempo.

En trabajos anteriores [12] [13] se presentaron tres implementaciones de una versión del algoritmo AES que realiza cómputo intensivo. Éstas fueron desarrolladas con las herramientas de programación paralela OpenMP, MPI y CUDA, para ser ejecutadas sobre una máquina multicore, cluster de multicore y GPU respectivamente. El trabajo experimental mostró que la implementación de AES para GPU alcanza una mayor eficiencia tanto en tiempo de cómputo como en consumo energético, en comparación con las implementaciones del algoritmo AES para las arquitecturas restantes.

Asimismo, en [14] se analizó el rendimiento de AES sobre una máquina con una GPU y sobre un cluster de GPU, para casos en que la memoria requerida por el algoritmo supere la memoria de una GPU. Para esto se realizaron dos implementaciones: una utilizando CUDA que se ejecuta sobre una única GPU dividiendo en fragmentos los datos a cifrar e invocando varias veces al kernel AES, y otra que utiliza una combinación de MPI y CUDA para ser ejecutada sobre un cluster de GPU. El trabajo experimental mostró que las comunicaciones en el cluster de GPU tienen un impacto negativo en el tiempo total de cómputo del algoritmo. Por lo tanto, para este algoritmo y arquitectura disponible, es más adecuado dividir los datos en fragmentos y cifrar cada uno utilizando una única GPU, que utilizar un cluster de GPU.

En este trabajo se realiza una comparación de rendimiento de dos versiones de AES sobre arquitecturas multicore. La primera versión es aquella que se presentó en el trabajo [12], a la cual se denomina AES-CI (AES Cómputo Intensivo). La segunda versión, AES-AIM (AES Acceso Intensivo a Memoria), reduce la cantidad de cálculo reemplazando operaciones por accesos a datos pre-calculados almacenados en memoria. Esto es posible porque AES se basa en un álgebra cerrada, por lo tanto el

resultado de ciertas operaciones se puede mantener en tablas. Esta versión de AES está disponible en la librería OpenSSL[15].

Para cada versión se realizaron tres implementaciones que utilizan herramientas de programación paralela OpenMP, MPI y CUDA, para ser ejecutadas sobre una máquina multicore, cluster de máquinas multicore y GPU respectivamente.

El trabajo experimental muestra que las implementaciones de AES-AIM reducen en todos los casos el tiempo de ejecución respecto a AES-CI. Sin embargo, las implementaciones de AES-AIM secuencial, OpenMP y MPI alcanzan una mejora similar con respecto a AES-CI, mientras que AES-AIM CUDA exhibe un menor grado de mejora respecto a AES-CI CUDA, debido a las limitaciones de latencia de memoria propias de la arquitectura GPU.

2 DESCRIPCIÓN DEL ALGORITMO AES

AES (Advanced Encryption Standard) se caracteriza por ser un algoritmo de cifrado por bloques. Los datos a encriptar se dividen en bloques de tamaño fijo (128 bits), donde cada bloque se representa como una matriz de 4x4 bytes llamada estado como se muestra en la Fig. 1.

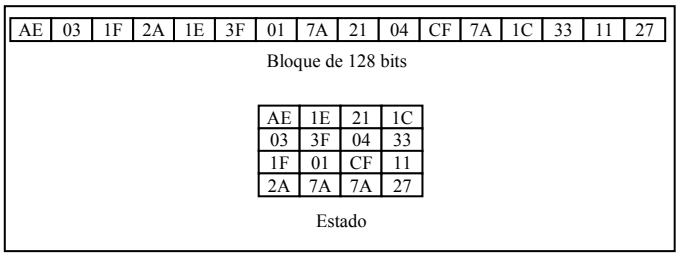


Fig. 1. Estado AES.

A cada estado se le aplican once rondas, cada una está compuesta por un conjunto de operaciones. Las once rondas se pueden clasificar en tres tipos: una ronda inicial, nueve rondas estándar y una ronda final.

Por ser AES un algoritmo simétrico, utiliza la misma clave para cifrar y descifrar los datos, cuyo tamaño es de 128 bits según lo indica el estándar. A esta clave se la denomina clave inicial, y a partir de ella se generan diez claves más mediante un procedimiento matemático. Las diez claves resultantes junto con la clave inicial son denominadas subclaves y cada una es utilizada en una de las rondas.

La ronda inicial realiza una sola operación:

- *AddRoundKey:* se hace un XOR byte a byte entre el estado y la clave inicial.

En cada una de las siguientes nueve rondas, denominadas estándar, se aplican 4 operaciones en este orden:

- *SubBytes:* se reemplaza cada byte del estado por otro de acuerdo a una tabla de sustitución de bytes con valores predeterminados. Este valor resultante se obtiene accediendo a la tabla tomando como índice de fila los primeros 4 bits

del byte a reemplazar y como índice de columna los últimos 4 bits. El tamaño de la tabla es de 16x16 bytes.

- *ShiftRows:* a excepción de la primera fila del estado, que no se modifica, los bytes de las filas restantes se rotan cíclicamente a izquierda: una vez en la segunda fila, dos veces en la tercera y tres veces en la cuarta.
- *MixColumns:* a cada columna del estado se le aplica una transformación lineal y es reemplazada por el resultado de esta operación.
- *AddRoundKey:* es igual a la ronda inicial pero utilizando la siguiente subclave.

La ronda final consiste de 3 operaciones:

- *SubBytes:* de la misma forma que se aplica a las rondas estándar.
- *ShiftRows:* de la misma forma que se aplica a las rondas estándar.
- *AddRoundKey:* al igual que las rondas anteriores pero utilizando la última subclave.

En la Fig. 2 se muestra un esquema del algoritmo.

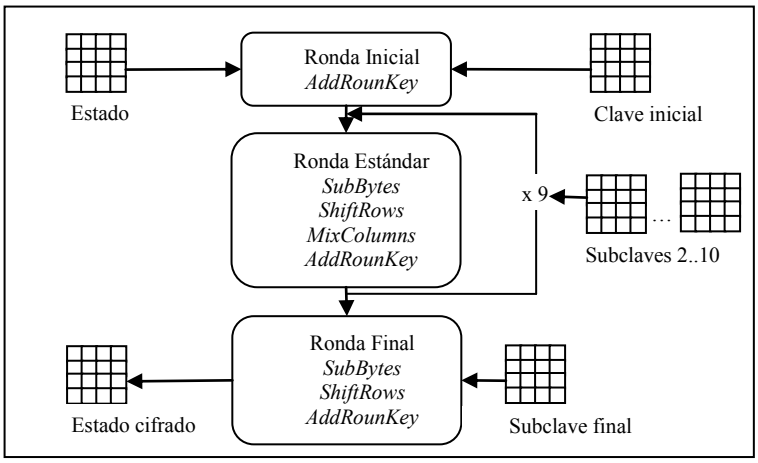


Fig. 2. Rondas del algoritmo AES sobre un estado.

Hay dos formas de implementar el algoritmo:

- La primera consiste en realizar intensivamente todos los cálculos de cada operación que componen los tres tipos de rondas. A esta implementación se le denominó *AES-CI* (AES Cómputo Intensivo).
- La segunda consiste en generar tablas que almacenan datos pre-calculados, que no dependen de la entrada y que se van a utilizar en algunas operaciones. De esta forma, se reemplaza cálculo por acceso a memoria. A esta implementación se le denominó *AES-AIM* (AES Acceso Intensivo a Memoria).

3 MODELO DE PROGRAMACIÓN CUDA

El sistema de cómputo para un programador CUDA está compuesto por la CPU, también llamada host, y una o más GPUs, llamadas devices.

Un programa CUDA se divide en fases a ejecutar en el host y fases a ejecutar en el device. El código a ejecutar en el device recibe el nombre de kernel.

Cuando se invoca a un kernel se debe especificar la cantidad de hilos a utilizar en la ejecución y su organización. CUDA permite organizar los hilos en Grids. Un Grid es un conjunto de bloques de hilos.

Una GPU posee un conjunto de procesadores, llamados Streamming Multiprocessors (SM), y una jerarquía de memoria compuesta por: la memoria global, la memoria de constantes y una memoria compartida ubicada en cada SM. La memoria global es la que posee mayor latencia en la jerarquía, mientras que la memoria compartida es la que posee menor latencia. La memoria de constantes tiene una latencia intermedia.

Dado que el acceso a memoria global es costoso, los hilos suelen trabajar en memoria compartida. Para esto los hilos deben traer todos los datos que necesiten desde la memoria global a la memoria compartida, esta transferencia de datos implica una gran una cantidad de accesos a memoria global, pero CUDA permite utilizar la coalescencia como técnica de optimización para minimizar la cantidad de accesos. Esta técnica permite mover varios datos desde memoria global a memoria compartida (y viceversa) en un único acceso.

4 IMPLEMENTACIONES DE AES-CI

Se realizaron cuatro implementaciones de AES-CI, una implementación secuencial y las restantes utilizando distintas herramientas de programación paralela tales como OpenMP, MPI y CUDA.

4.1 Implementación secuencial (AES-CI-SEC)

La implementación secuencial del algoritmo genera las subclaves a partir de la clave inicial. A continuación, para cada estado de 16 bytes de los datos a cifrar aplica las rondas utilizando las subclaves generadas inicialmente.

4.2 Implementaciones paralelas

Las implementaciones paralelas consideran a los datos de entrada como estados consecutivos. Además, se tiene una cantidad determinada de procesos o hilos y cada uno se encargará de cifrar un conjunto de estados. La distribución de los estados es proporcional a la cantidad de procesos o hilos, es decir si el tamaño de los datos de entrada es de N bytes, la cantidad de estados es $B = N/16$. Si se tienen P procesos o hilos, cada uno deberá cifrar B/P estados.

La generación de las subclaves se realiza secuencialmente en todos los casos por ser una tarea muy simple y el tiempo de ejecución de este cálculo es despreciable. Una vez generadas las subclaves, los procesos o hilos las utilizan para el proceso de cifrado de sus estados.

4.2.1 Implementación usando OpenMP (AES-CI-OMP)

La implementación de AES-CI propuesta con OpenMP genera en forma secuencial las subclaves a partir de la clave inicial. Luego se crean un conjunto de hilos, tantos como cores provea la arquitectura, cada uno de los hilos tomará un conjunto consecutivo de estados y le aplicará el proceso de encriptación.

4.2.2 Implementación usando MPI (AES-CI-MPI)

La implementación de AES-CI propuesta con MPI, parte de tener una cantidad determinada de procesos, tantos como procesadores se tengan. Uno de ellos genera secuencialmente las subclaves a partir de la clave inicial y las comunica. Luego distribuye proporcionalmente los estados entre los procesos, incluyéndose a sí mismo. Cada proceso encriptará los estados que le correspondan.

4.2.3 Implementación usando CUDA (AES-CI-CUDA)

El cálculo de las subclaves se realiza en el host, ya que el tiempo de ejecución es despreciable, dejando al device sólo el procedimiento de cifrado.

El host copia en la memoria de constantes del device las subclaves y la tabla de sustitución de bytes, dado que ambas solo serán leídas por los hilos. Luego copia los datos a cifrar en la memoria global del device.

A continuación, invoca al kernel especificando la organización del Grid, por lo tanto se debe indicar la cantidad de bloques de hilos y la cantidad de hilos por cada bloque.

Los hilos pertenecientes a un mismo bloque CUDA trabajarán sobre estados consecutivos, cada uno se encargará de cifrar un estado. Dado que el acceso a memoria global es costoso, previo a la etapa de cifrado del estado, cada hilo cooperará con los hilos de su mismo bloque para cargar en memoria compartida la información que deben cifrar. Estos accesos se hacen de manera coalescente. Una vez terminada la etapa de cifrado, los hilos cooperan para trasladar los datos desde la memoria compartida a la memoria global de manera coalescente.

5 IMPLEMENTACIONES DE AES-AIM

Se realizaron cuatro implementaciones de AES-AIM, una implementación secuencial (AES-AIM-SEC) y tres implementaciones paralelas denominadas AES-AIM-OMP, AES-AIM-MPI y AES-AIM-CUDA, desarrolladas con OpenMP, MPI y CUDA respectivamente.

Las implementaciones de AES-AIM realizan invocaciones a la librería OpenSSL para realizar el cifrado o descifrado de datos. En particular, las implementaciones paralelas de AES-CI y AES-AIM son similares en la forma en que distribuyen los datos entre los procesos o hilos.

La librería OpenSSL no está pensada para utilizarse en algoritmos paralelos por lo que fue necesario hacer una adaptación para ser usada con OpenMP, MPI y CUDA. En el caso de OpenMP y MPI la librería no requiere de grandes modificaciones. Sin embargo, en el caso de CUDA es necesario indicar que las variables, constantes y funciones deben conocerse dentro de la GPU para que la ejecución del algoritmo de cifrado se lleve a cabo en el device.

6 TRABAJO EXPERIMENTAL

El algoritmo secuencial fue ejecutado en una máquina con arquitectura Intel Xeon E5405 [16] y 2GB de memoria RAM.

El algoritmo de memoria compartida, que utiliza OpenMP, fue ejecutado en una máquina con 2 procesadores Intel Xeon E5405 con 4 cores cada uno, y 2GB de memoria RAM. El algoritmo MPI fue ejecutado utilizando un cluster de 4 máquinas con la arquitectura anteriormente mencionada, conectadas por una red 1Gbit Ethernet y utilizando 8, 16 y 32 cores.

El algoritmo CUDA fue ejecutado en una tarjeta gráfica Nvidia Geforce GTX 560TI [17] con 1GB de RAM que posee 384 Scalar Processors, distribuidos en 8 Streaming Multiprocessors.

Generalmente, los algoritmos CUDA pueden alcanzar distinto rendimiento según la cantidad de hilos por bloques especificada. Tanto para el algoritmo AES-CI-CUDA como AES-AIM-CUDA el mejor rendimiento se obtiene con 256 hilos por bloque.

Para realizar el análisis de rendimiento se cifraron datos de tamaño 512KB, 1MB, 15MB, 128 MB y 255MB.

En este trabajo, sólo se tiene en cuenta el proceso de cifrado para los distintos tamaños de datos de entrada. El proceso de descifrado no se tuvo en cuenta por tener un rendimiento similar.

En el trabajo previo [12] se mostró que la implementación de AES-CI para GPU alcanza mayor rendimiento, en comparación con las implementaciones del algoritmo AES-CI para las arquitecturas restantes. En la Fig. 3 se puede ver el alto speedup alcanzado por AES-CI-CUDA con respecto a las otras implementaciones paralelas de AES-CI, para archivos a cifrar de distinto tamaño. El Speedup se calculó en relación al tiempo de AES-CI-SEC sobre una máquina del cluster.

Por lo antes mencionado, si se tiene un algoritmo con las características de AES-CI, donde el cómputo es intensivo, será conveniente utilizar una GPU dado que garantiza un alto rendimiento.

Por otro lado, AES-AIM realiza menor cantidad de cálculo y permite obtener mejor rendimiento que AES-CI. La Fig. 4 muestra el porcentaje de reducción en el tiempo de ejecución del algoritmo secuencial AES-AIM-SEC con respecto al algoritmo secuencial AES-CI-SEC. Como se puede observar, en general AES-AIM-SEC reduce en aproximadamente un 99% el tiempo de AES-CI-SEC.

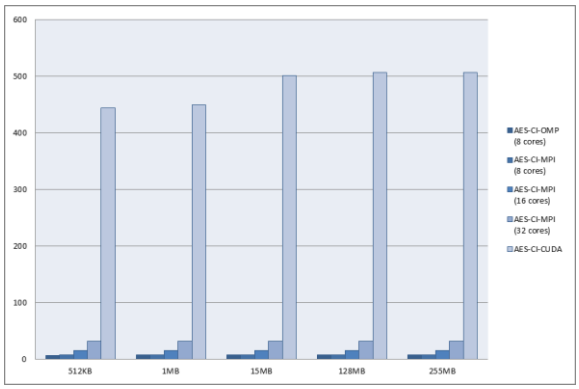


Fig. 3. Speedup de las implementaciones paralelas de AES-CI.

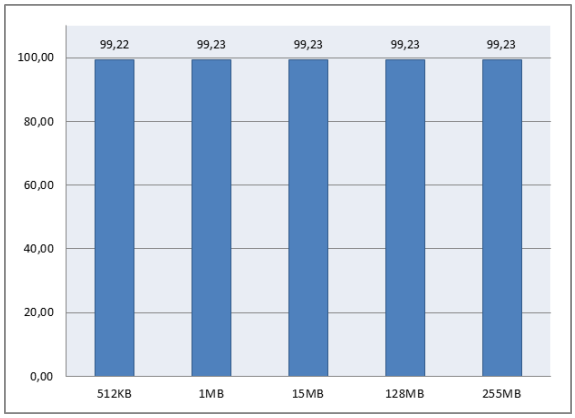


Fig. 4. Porcentaje de reducción de AES-AIM-SEC respecto a AES-CI-SEC

La Fig. 5 muestra el porcentaje de reducción en el tiempo de ejecución de las implementaciones paralelas AES-AIM con respecto las implementaciones paralelas respectivas de AES-CI. En particular, AES-AIM-OMP y AES-AIM-MPI alcanzan una reducción similar con respecto a AES-CI-OMP y AES-CI-MPI respectivamente, siendo la misma aproximadamente de 99%.

En contraste con lo anterior, la reducción obtenida por AES-AIM-CUDA con respecto a AES-CI-CUDA varía entre 56% y 62%.

La Fig. 6 muestra el speedup de las implementaciones paralelas de AES-AIM. El Speedup se calculó en relación al tiempo de AES-AIM-SEC sobre una máquina del cluster. De los resultados se observa que el rendimiento obtenido por AES-AIM-CUDA es menor al obtenido por AES-AIM-MPI cuando utiliza más de una máquina del cluster. El menor rendimiento de la GPU se debe a la manera en la cual estos dispositivos gestionan los hilos para intentar ocultar la latencia de memoria. Cuando un conjunto de hilos hace operaciones de memoria (load o store), la GPU coloca otro conjunto de hilos a procesar mientras se resuelven los accesos a memoria de los primeros. Como AES-AIM prácticamente no realiza procesamiento, existe un punto en el cual la latencia no puede ser ocultada.

Por lo tanto, si se tiene un algoritmo con las características de AES-AIM, el mejor rendimiento se obtiene al utilizar un cluster de máquinas multicore.

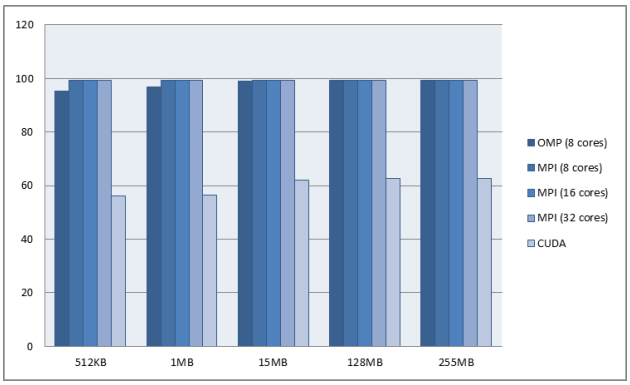


Fig. 5. Porcentaje de reducción de las implementaciones paralelas de AES-AIM vs. AES-CI

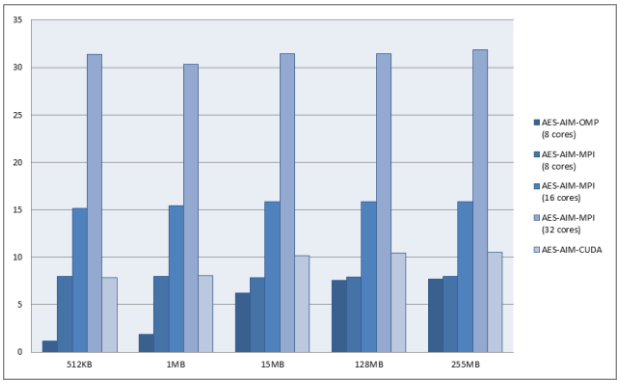


Fig. 6. Speedup de las implementaciones paralelas de AES-AIM.

7 CONCLUSIONES

Este trabajo presentó una comparación del rendimiento obtenido por dos versiones del algoritmo AES sobre distintas arquitecturas multicore (máquina multicore, cluster de máquinas multicore y GPU). Una versión está caracterizada por realizar cómputo intensivo (AES-CI) y otra por realizar acceso intensivo a memoria (AES-AIM), la cual reemplaza cálculo por acceso a datos pre-calculados. Para cada versión del algoritmo se presentaron tres implementaciones paralelas utilizando OpenMP, MPI y CUDA.

El trabajo experimental mostró que las implementaciones de AES-AIM reducen en todos los casos el tiempo de ejecución respecto a AES-CI. Sin embargo, las implementaciones de AES-AIM secuencial, OpenMP y MPI reducen el tiempo de ejecución en un 99% con respecto a AES-CI, mientras que AES-AIM CUDA exhibe

un menor grado de reducción respecto a AES-CI CUDA, debido a las limitaciones de latencia de memoria propias de la arquitectura GPU.

De los resultados experimentales se concluye lo siguiente: si se tiene un algoritmo con las características de AES-AIM, el cual realiza acceso intensivo a memoria, no será conveniente utilizar una GPU frente a un cluster, si estos poseen características similares a las utilizadas en este estudio. Esto contrasta con lo observado para AES-CI, algoritmo que realiza cómputo intensivo, donde se observó la gran eficiencia alcanzada por la versión para GPU respecto a las demás implementaciones paralelas para multicore y cluster de multicore.

Referencias

1. Chapman B., The Multicore Programming Challenge, Advanced Parallel Processing Technologies; 7th International Symposium, (7th APPT'07), Lecture Notes in Computer Science (LNCS), Vol. 4847, p. 3, Springer-Verlag (New York), November 2007.
2. Suresh Siddha, Venkatesh Pallipadi, Asit Mallick. "Process Scheduling Challenges in the Era of Multicore Processors"Intel Technology Journal, Vol. 11, Issue 04, November 2007.
3. MPI Specification http://www.mpi-forum.org/docs/mpi-2.2/mpi22-report.pdf.
4. The OpenMP API specification for parallel programming. http://openmp.org/wp/.
5. Cuda Home Page http://www.nvidia.com/object/cuda_home_new.html
6. Grama A., Gupta A., Karypis G., Kumar V. "Introduction to Parallel Computing". Second Edition. Addison Wesley, 2003.
7. Bischof C., Bucker M., Gibbon P., Joubert G., Lippert T., Mohr B., Peters F. (eds.), Parallel Computing: Architectures, Algorithms and Applications, Advances in Parallel Computing, Vol. 15, IOS Press, February 2008.
8. General-Purpose Computation on Graphics Hardware http://gpgpu.org/.
9. FIPS PUB 197: the official AES Standard http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.
10. Lynn Hathaway (June 2003). "National Policy on the Use of the Advanced Encryption Standard (AES) to Protect National Security Systems and National Security Information"http://csrc.nist.gov/groups/ST/toolkit/documents/aes/CNSS15FS.pdf.
11. D.J.Bernstein-Cache-timing attacks on AES (2005) http://cr.yp.to/antiforgery/cachetiming-20050414.pdf.
12. Pousa A., Sanz V., De Giusti A. Performance Analysis of a Symmetric Cryptographic Algorithm on Multicore Architectures. Computer Science & Technology Series - XVII Argentine Congress of Computer Science - Selected Papers. Edulp 2012.
13. Romero F., Pousa A., Sanz V., De Giusti A. Consumo Energético en Arquitecturas Multicore. Análisis sobre un Algoritmo de Criptografía Simétrica. Proceedings del XVIII Congreso Argentino de Ciencias de la Computación. ISBN 978-987-1648-34-4
14. Pousa, A, Sanz, V., De Giusti, A. Performance Analysis of a Symmetric Cryptography Algorithm on GPU and GPU Cluster. VI Latin American Symposium on High Performance Computing HPCLatAm 2013. http://hpc2013.hpclatam.org/papers/HPCLatAm2013-paper-12.pdf
15. OpenSSL. The OpenSSL Project. Cryptography and SSL/TLS Toolkit. https://www.openssl.org/
16. Intel Product Specifications http://ark.intel.com/products/33079/Intel-Xeon-Processor-E5405-(12M-Cache-2_00-GHz-1333-MHz-FSB).
17. Nvidia Geforce GTX 560TI Specifications http://www.nvidia.com/object/product-geforce-gtx-560ti-us.html.

Análisis del impacto de distintas técnicas de optimización de rendimiento en multicore

Matías Dell'Oso[1], Juan Manuel Paniego[1], Martín Pi Puig[1],
Marcelo Naiouf[1], Armando De Giusti[1,2]

[1] Instituto de Investigación en Informática LIDI (III-LIDI) – Facultad de Informática –UNLP
[2] Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
Argentina
{mdelloso, jmpaniego, mpipuig, mnaiouf, degiusti}@lidi.info.unlp.edu.ar

Abstract. Este trabajo se enfoca en la comparación de distintas técnicas para reducir el tiempo de ejecución de un algoritmo. En primer lugar se analizan dos APIs para la programación multiproceso de memoria compartida, Pthreads y OpenMP realizando una breve comparación entre ellas y resaltando ventajas y desventajas de cada una. Luego, se demuestra la importancia del buen uso de la memoria caché y cómo impacta en la performance de programas tanto paralelos como secuenciales. Por último, se utilizan distintas optimizaciones brindadas por el compilador para aumentar la performance de los algoritmos. Al finalizar todas las pruebas se presenta una comparación entre las tres técnicas estudiadas, resaltando los escenarios en los cuales cada una de ellas presenta mejores resultados. El caso de estudio elegido es el problema clásico de multiplicación de matrices, utilizado para demostrar el impacto de la utilización óptima de la jerarquía de memoria existente en una arquitectura paralela.

Keywords: Arquitecturas paralelas, Multicore, Jerarquía de memoria, Pthreads, OpenMP.

1 Introducción

Tradicionalmente, la simulación numérica de sistemas complejos como por ejemplo dinámica de fluidos, clima, circuitos electrónicos, reacciones químicas, modelos ambientales, ha impulsado el desarrollo de computadores cada vez más potentes. Inicialmente, esto se alcanzaba aumentando la velocidad de los procesadores (según la conocida Ley de Moore [5]), incrementando los puntos de acceso a los datos (lo que disminuye los tiempos de disponibilidad de los mismos) e implementando algoritmos y técnicas optimizadas para resolver las tareas eficientemente.

Estas técnicas alcanzaron pronto sus límites, ya que los procesadores disipaban cada vez más calor generando fallas y eran más costosos en relación a sus prestaciones. Además, ya no era posible seguir optimizando los algoritmos para obtener mejores resultados en cuanto a tiempo. Para dar solución a estos problemas surgen los multiprocesadores, y con ellos la programación paralela. Actualmente, las arquitecturas dominantes son los *multicore*, donde en un mismo chip se incluyen varios núcleos de procesamiento [6][7].

El hecho de dividir el trabajo entre 2 o más procesadores permite bajar sus frecuencia de reloj (reduciendo sus costos) y obtener resultados superiores a los obtenidos utilizando un único procesador. Sin embargo, a pesar del buen rendimiento de los multiprocesadores, el desarrollo de hardware y software paralelo ha sido tradicionalmente dificultoso.

En ese sentido, surgieron nuevas problemáticas como la coherencia de los datos y el acceso compartido a los recursos. Para hacer frente a esto, se desarrollaron APIs y metodologías para asegurar la confiabilidad de las aplicaciones paralelas.

En el caso de los multicore, debe tenerse en cuenta la jerarquía de memorias involucradas, generalmente con diferentes niveles de caché, para obtener buenas prestaciones. En este trabajo se realiza una comparación entre la API Pthreads y la API OpenMP, resaltando las ventajas y desventajas de cada una de ellas, y se mostrará la importancia del buen uso de la memoria caché y cómo impacta en la performance de programas tanto paralelos como secuenciales.

Como caso de estudio se utiliza la multiplicación de matrices, un problema conocido y que suele ser parte de soluciones más generales. Para ello se realizan una serie de multiplicaciones de matrices utilizando ambas librerías y se comparará tanto el speedup alcanzado por cada una de ellas respecto del algoritmo secuencial, como también la simplicidad de cada código. Luego, se implementan distintas técnicas de optimización de la memoria caché, para demostrar que en muchos casos su buen uso puede arrojar mejores resultados que paralelizar la aplicación. Por último, se utilizaron comandos especiales del compilador GCC para compilar más "agresivamente" y así obtener mejores resultados.

El trabajo está organizado de la siguiente manera: la Sección 2 presenta el caso de estudio y el equipo utilizado. A continuación, en las Secciones 3, 4 y 5 se presentan los resultados obtenidos para las tres optimizaciones propuestas, así como los basamentos teóricos de cada una de ellas. El análisis de los resultados obtenidos así como la comparación entre las tres técnicas utilizadas se muestra en la Sección 6. Por último, en la Sección 7, se exponen las conclusiones.

2 Caso de estudio y equipo utilizado

Se eligió la multiplicación de matrices como caso de estudio, ya que representa un problema ampliamente tratado y que en muchos casos es parte de soluciones a problemas de mayor envergadura. Dadas dos matrices *A* de *m* x *p* y *B* de *p* x *n* elementos, la multiplicación de ambas consiste en obtener la matriz *C* de *m* x *n* elementos (*C* = *A* x *B*), donde cada elemento se calcula por medio de la Ecuación 1, tal como muestra la Figura 3.

$$C_{i,j} = \sum_{k=1}^{p} A_{i,k} * B_{k,j} \quad (1)$$

Para encontrar el producto de las matrices existen diferentes algoritmos, entre otros el que utiliza 3 iteraciones y la solución por bloques [8][9].

En este trabajo la experimentación se realizó sobre un equipo con procesador Intel Xeon [13] y 8 GB de memoria RAM. Todos los algoritmos fueron compilados con el GNU Compiler Collection (GCC) corriendo sobre Fedora release 14 (Laughlin). Las especificaciones del equipo son las siguientes: Procesador: Intel Xeon X3430; Cores: 4; Threads: 4; Frecuencia de reloj: 2400 MHz; Intel Turbo Boost: 2800 MHz (deshabilitado); Caché de datos L1: 32 KB por core; Caché L2: 256 KB por core; Caché L3: 8 MB compartida.

3 Pthreads vs OpenMP

Históricamente, los vendedores de hardware implementaban sus propias versiones de threads. Estas implementaciones diferían completamente unas de otras, haciendo dificultosa la programación de aplicaciones portables. Para hacer frente a este problema y poder explotar todo el potencial de los threads, fue necesario crear un estándar. Para los sistemas UNIX, la IEEE especificó el estándar 1003.1c-1995, también conocido como Pthreads [11], el cual es soportado por la mayoría de los vendedores de hardware [1]. Si bien la API Pthreads ha ido evolucionando, su uso continúa siendo restringido a un sector acotado de programadores. Esto se debe a que su modelo de programación se basa en primitivas de bajo nivel y puede resultar complejo para los programadores afines a utilizar lenguajes de alto nivel.

Por ello surgió OpenMP [10] (Open Multi-Processing), una API que provee un modelo de programación portable y escalable que proporciona a los programadores una interfaz simple y flexible para el desarrollo de aplicaciones paralelas. A su vez, las directivas de OpenMP proveen soporte para concurrencia, sincronización y manejo de datos, obviando el uso explícito de semáforos, variables condición, alcance de los datos e inicialización de threads [2].

Para realizar la comparación entre las dos APIs, se utiliza el tiempo que tardan en ejecutarse tres algoritmos de multiplicación de matrices cuadradas (dimensión *N*x*N*) y que almacenan datos en punto flotante de doble precisión. La primera multiplicación se realiza utilizando un algoritmo secuencial, la segunda utilizando Pthreads y la última utilizando OpenMP, variando la dimensión de las matrices. En la Tabla 1 se pueden observar los resultados obtenidos.

Tabla 1. Tiempos obtenidos para cada algoritmo (en segundos).

N	Secuencial	Pthread (4 hilos)			OMP (4 hilos)		
	Tiempo	Tiempo	Speedup	Eficiencia	Tiempo	Speedup	Eficiencia
512	1,6643	0,4732	3,5171	0,8792	0,6394	2,6029	0,6507
1024	24,3859	6,2967	3,8727	0,9681	8,0689	3,0221	0,7555
2048	284,3292	81,2401	3,4998	0,8749	112,0459	2,5376	0,6344
4096	2524,209	693,5912	3,6393	0,9098	934,7879	2,7003	0,6751
8192	21854,622	5941,7058	3,6781	0,9195	8379,496	2,6081	0,6525
16384	205989,06	54546,853	3,7763	0,944	71807,286	2,8686	0,7171

Como se observa en la tabla anterior, al utilizar la API Pthreads se obtiene un mayor speedup y eficiencia que usando OpenMP. Esto se debe a que con Pthreads se tiene un mejor control sobre las tareas a realizar, es decir, el programador se encarga de la concurrencia y de la creación de los threads. Por otra parte, programar aplicaciones paralelas utilizando OpenMP es muy sencillo, por lo que simplifica de manera notable la creación de dichas aplicaciones.

Vale la pena destacar que si la programación se realiza de manera correcta, resulta muy difícil obtener un mayor speedup utilizando OpenMP que mediante el uso de Pthreads. De todas maneras, si el programador no está acostumbrado a trabajar con primitivas de bajo nivel o si el programa a resolver es muy complejo, es probable que se obtengan mejores resultados utilizando OpenMP. Es por esto que resulta una herramienta sumamente útil para programadores menos experimentados en programación paralela.

4 Optimización de la memoria caché

Los algoritmos con los que se realizaron los cálculos anteriores no presentan optimización alguna. Ambas matrices A y B se inicializan por filas y luego se realiza la multiplicación accediendo a cada una de ellas del mismo modo. Al inicializar las matrices A y B de la misma forma, a la hora de multiplicarlas, los fallos de caché son muchos, lo que provoca un incremento considerable en el tiempo de ejecución del algoritmo. Como se observa en la Figura 1, para calcular el valor C[1,1] se necesitan todos los valores de la fila 1 de la matriz A y todos los valores de la columna 1 de la matriz B.

Cuando un dato es copiado desde la memoria RAM hacia la memoria caché, también son copiados los datos que están contiguos en dicha memoria (por principio de localidad espacial [1]). Al estar cargadas ambas matrices por filas, una porción de una fila de cada matriz es copiada a la memoria caché para su rápida utilización. Para la matriz A, esto no presenta problemas, ya que toda la fila es necesaria para realizar el cálculo. La matriz B, por otro lado, al necesitar una columna entera, produce un fallo de caché en cada iteración de la multiplicación (si consideramos matrices de dimensiones grandes). Esto genera que todos los datos de B deban ser buscados en la memoria principal, lo que agrega un overhead de comunicaciones sumamente alto.

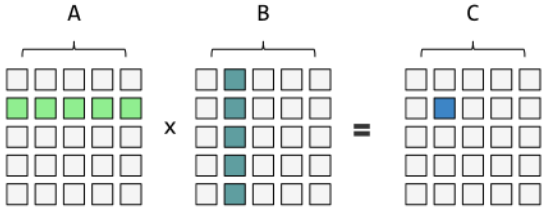


Fig.1. Multiplicación de matrices.

Para hacer frente a este problema se proponen una serie de modificaciones a los algoritmos buscando optimizar el acceso a los datos:

a) Optimizar la carga

Como se mencionó anteriormente, la matriz B presenta problemas debido a que está inicializada por filas. Al modificar el algoritmo para que B sea inicializada por columnas, se reducirán significativamente los fallos de caché (ver Tabla 2). De esta manera, una porción de la fila de la matriz A y una porción de la columna de la matriz B son cargadas en caché, lo que reduce notablemente el acceso a la memoria principal [4].

Tabla 2. Tiempos obtenidos para cada algoritmo optimizado (en segundos).

N	Secuencial	Pthread (4 hilos)			OMP (4 hilos)		
	Tiempo	Tiempo	Speedup	Eficiencia	Tiempo	Speedup	Eficiencia
512	1,1375	0,3166	3,5921	0,898	0,4266	2,6658	0,6665
1024	9,1716	2,5337	3,6197	0,9049	3,4052	2,6933	0,6733
2048	74,0484	20,459	3,6193	0,9048	27,4889	2,6937	0,6734
4096	594,9148	164,6297	3,6136	0,9034	221,2108	2,6893	0,6723
8192	4759,606	1324,293	3,59407	0,8985	1767,229	2,6932	0,6733
16384	38061,901	10626,83	3,58167	0,8954	14146,057	2,6906	0,6726

Se observa que la relación entre los tiempos obtenidos con los tres algoritmos (secuencial, Pthreads y OMP) se asemeja a la obtenida sin la optimización, pero ahora todos los tiempos se redujeron notablemente.

b) Optimización de caché L1 - Multiplicación por bloques

Para optimizar aún más el acceso a los datos, se utiliza un algoritmo de multiplicación por bloques [3]. La idea es dividir las matrices A y B en sub-matrices más pequeñas que quepan en la caché L1 con el fin de realizar la multiplicación con un número menor de fallos de caché. La clave del éxito de este algoritmo es elegir el tamaño óptimo del bloque el cual puede ser calculado según la Ecuación 2:

$$2 * (tamañoDeBloque)^2 * tamañoPalabra = tamañoCacheL1 \quad (2)$$

En este caso, *tamañoPalabra* es 8 bytes debido a que las matrices son del tipo double. Nótese que se multiplica por 2, debido a que es necesario que quepan 2 bloques. Sustituyendo los valores, se calcula que el tamaño de bloque óptimo es de 45. Sin embargo, se encontró que el tamaño que arroja mejores resultados (utilizando potencias de 2) es 32. Esto puede deberse a que las pruebas no se realizaron sobre un sistema aislado por lo que otro proceso en segundo plano pudo estar ocupando parte de la memoria caché. En la Tabla 3 se muestra los resultados obtenidos.

Tabla 3. Tiempos obtenidos para cada algoritmo optimizado por bloques (en segundos).

N	Secuencial	Pthread (4 hilos)			OMP (4 hilos)		
	Tiempo	Tiempo	Speedup	Eficiencia	Tiempo	Speedup	Eficiencia
512	1,0817	0,2987	3,6208	0,9052	0,3930	2,7519	0,6879
1024	8,6606	2,3916	3,6211	0,9052	3,1478	2,7512	0,6878
2048	69,2909	19,1524	3,6178	0,9044	25,1873	2,7510	0,6877
4096	554,0851	153,1872	3,6170	0,9042	201,6417	2,7478	0,6869
8192	4433,513	1228,393	3,6091	0,9022	1612,661	2,7491	0,6872
16384	35482,77	9837,132	3,6070	0,9017	12903,712	2,7498	0,6874

Si bien la mejora no es sustancial respecto de la anterior optimización, utilizando el algoritmo por bloques se consigue disminuir nuevamente el tiempo de ejecución de las tareas. Esto se debe a que los dos bloques a multiplicar entran por completo en la memoria caché L1, lo que reduce aún más los fallos de dicha memoria.

5 Optimización del compilador

GCC provee varias opciones de compilación para mejorar el rendimiento de una aplicación. A continuación se observan las opciones de compilación utilizadas para realizar las pruebas:

Secuencial	gcc –O3 -o nombreEjecutable nombreArchivo.c
Pthreads	gcc –O3 -o nombreEjecutable nombreArchivo.c -lpthread
OpenMP	gcc –O3 -o nombreEjecutable nombreArchivo.c -fopenmp

No es el objetivo de este trabajo entrar en detalle sobre los comandos existentes y su funcionamiento. Es posible visualizar una lista detallada de todas las opciones de compilación en la página web de GCC [12] o utilizando el comando “man” en sistemas Unix.

La optimización se realiza sobre el algoritmo de multiplicación por bloques. En la Tabla 4 se encuentran los resultados obtenidos.

Tabla 4. Algoritmo optimizado por compilador (tiempos en segundos).

N	Secuencial	Pthreads (4 hilos)			OMP (4 hilos)		
	Tiempo	Tiempo	Speedup	Eficiencia	Tiempo	Speedup	Eficiencia
512	0,1564	0,0504	3,1030	0,7757	0,0453	3,4509	0,8627
1024	1,2442	0,3940	3,1575	0,7893	0,3622	3,4345	0,8586
2048	9,9620	3,1570	3,1554	0,7888	2,8960	3,4398	0,8599
4096	79,6851	25,3469	3,1437	0,7859	23,2905	3,4213	0,8553
8192	638,314	205,845	3,1009	0,7752	187,6127	3,4022	0,8505
16384	5124,94	1664,209	3,0795	0,7698	1526,994	3,3562	0,8390

En la tabla 4 se puede observar una notable reducción de los tiempos de ejecución de todos los algoritmos. Cabe destacar que en esta ocasión la API OpenMP arrojó mejores resultados que la API Pthreads. Dado que el programador desconoce el funcionamiento interno de cada librería, no es posible predecir su comportamiento al combinarlas con optimizaciones del compilador. Es posible que utilizando otros compiladores los resultados varíen.

6 Análisis de resultados obtenidos

A lo largo de este trabajo se implementaron distintas técnicas para mejorar el tiempo de ejecución de una aplicación. Cada una de ellas aportó una mejora sobre la anterior llegando a obtener resultados muy interesantes. En la Tabla 5 se puede observar el speedup obtenido respecto del algoritmo secuencial sin optimizar, utilizando los tres tipos de optimizaciones vistos: paralelismo, optimización de caché y optimización del compilador.

Tabla 5. Tiempos obtenidos utilizando todas las optimizaciones (en segundos)

N	Algoritmo sin optimizar	Algoritmo optimizado	
	Tiempo	Tiempo	Speedup
512	1,6643	0,0453	36,72143
1024	24,3853	0,3622	67,3110
2048	284,3292	2,8960	98,1784
4096	2524,2094	23,2905	108,3797
8192	21854,6228	187,6127	116,4879
16384	205989,063	1526,9949	134,8983

Para la optimización por paralelismo se utilizó la API OpenMP (4 hilos) y para la optimización de acceso a los datos se utilizó el algoritmo de multiplicación por bloques. La Figura 2 muestra el aporte de cada una de las optimizaciones para las seis dimensiones de matrices analizadas.

En los gráficos se puede observar que, a medida que crece la dimensión de las matrices, la optimización de acceso a los datos juega un papel cada vez más relevante. Esto indica la importancia de hacer un buen uso de la memoria caché al manejar grandes volúmenes de información y de cómo impactan los fallos de dicha memoria en diferentes aplicaciones.

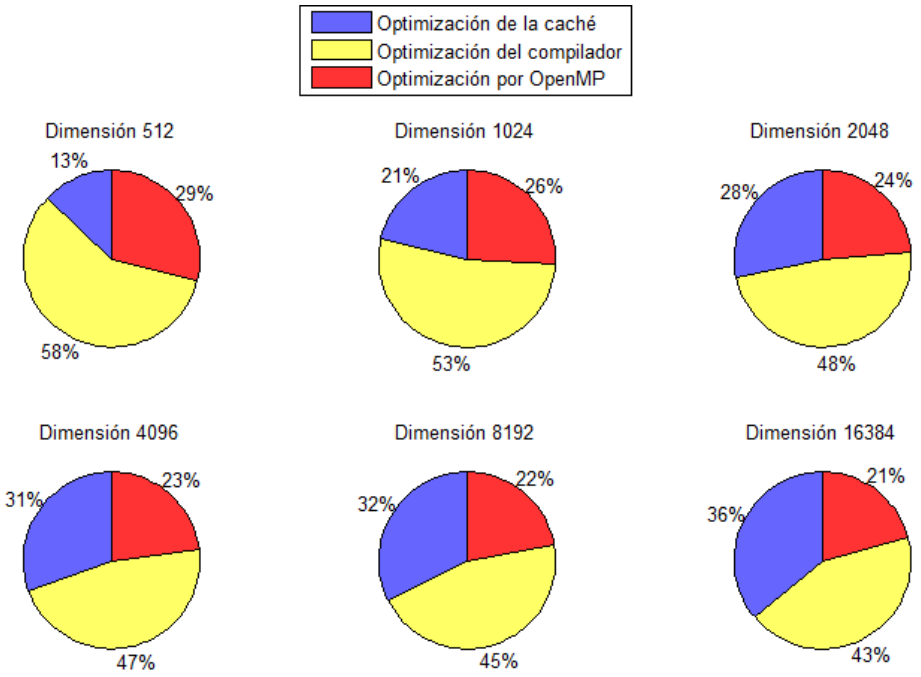


Fig. 2. Aporte de cada optimización.

A su vez, se observa que la optimización del compilador aporta una gran parte del speedup final obtenido, con la ventaja de que no implica esfuerzo adicional por parte del programador. Si bien el aporte de la librería OpenMP es el menor de los tres, no siempre es posible realizar optimizaciones por software sobre el algoritmo, ya sea por su complejidad o porque ya se alcanzó el nivel máximo de optimización. En estas situaciones, la programación multicore juega un papel fundamental a la hora de aumentar la performance de un algoritmo. Como en todas las aplicaciones la clave es saber elegir la técnica que mejor se aplique al problema y que otorgue el mayor beneficio al menor costo.

7 Conclusiones

A lo largo del trabajo se analizaron distintas técnicas utilizadas para optimizar una aplicación obteniendo resultados muy satisfactorios. Se demostró que aumentar el hardware paralelo no es la única opción para reducir su tiempo de ejecución. Mediante el uso de técnicas como reescribir el algoritmo para optimizar el acceso a los datos o compilar la aplicación utilizando comandos especiales, se pueden obtener mejoras que se asemejen a las que se obtendrían incrementando el número de procesadores o incluso mayores.

De todas maneras, si el algoritmo es muy complejo o simplemente ya no es posible optimizarlo por software, es aquí donde las técnicas de paralelización entran en juego.

Para ello puede utilizarse la API Pthreads, si se quiere el máximo speedup posible en detrimento de agregar complejidad al código, u OpenMP, si lo que se busca es disminuir el tiempo de desarrollo a expensas de un menor control sobre las tareas a realizar.

8 Bibliografía

1. Grama A., Gupta A., Karpyis G., Kumar V. "Introduction to Parallel Computing". Pearson – Addison Wesley 2003. ISBN: 0201648652. Segunda Edición (Capítulo 3 y 7).
2. Andrews G., Wesley A. "Foundations of Multithreaded, Parallel, and Distributed Programming". Disponible en Internet en: www.cs.arizona.edu/people/greg/mpdbook
3. Parello D., Temam O., Verdun JM. "On increasing architecture awareness in program optimizations to bridge the gap between peak and sustained processor performance – Matrix-Multiply revisited", Supercomputing, 2002.
4. Lam M., Rothberg E., Wolf M. "The Cache Performance and Optimizations of Blocked Algorithms", ASPLOS IV, 1991.
5. Hennessy J., Patterson D. "Computer Architecture: A Quantitaive Approach". Quinta Edición (Capítulo 1).
6. Chapman B., "The Multicore Programming Challenge, Advanced Parallel Processing Technologies"; 7th International Symposium, (7th APPT'07), Lecture Notes in Computer Science (LNCS), Vol. 4847, p. 3, Springer-Verlag (New York), November 2007.
7. Kumar V., Gupta A., "Analyzing Scalability of Parallel Algorithms and Architectures". Journal of Parallel and Distributed Computing. Vol 22, nro 1.Pags 60-79. 1994.
8. Pardines I. "Técnicas paralelas aplicadas a optimización no lineal en sistemas de memoria distribuída".
9. Guerequeta R., Vallecillo A. "Técnicas de Diseño de Algoritmos". Servicio de Publicaciones de la Universidad de Málaga. Segunda Edición (Capítulo 3).
10. OpenMP Tutorial. Disponible en Internet en: https://computing.llnl.gov/tutorials/openMP/
11. POSIX Threads Programming Tutorial. Disponible en Internet en: https://computing.llnl.gov/tutorials/pthreads/
12. GCC online documentation. Disponible en Internet en: https://gcc.gnu.org/onlinedocs/gcc-5.1.0/gcc/
13. Intel Xeon X3430. Disponible en Internet en: http://www.cpu-world.com/CPUs/Xeon/Intel-Xeon%20X3430%20-%20BV80605001914AG%20(BX80605X3430).html

Caracterización de una estrategia de detección de fallos transitorios en HPC

Diego Montezanti[1,4], Dolores Rexachs[2], Enzo Rucci[1,3]
Emilio Luque[2], Marcelo Naiouf[1] y Armando De Giusti[1,3],

[1] III-LIDI, Facultad de Informática, UNLP
Calle 50 y 120, 1900 La Plata (Buenos Aires), Argentina
{dmontezanti, erucci, mnaiouf, degiusti}@lidi.info.unlp.edu.ar
[2] Departamento de Arquitectura de Computadoras y Sistemas Operativos, UAB
Campus UAB, Edifici Q, 08193 Bellaterra (Barcelona), Spain
{dolores.rexachs, emilio.luque}@uab.es
[3] Consejo Nacional de Investigaciones Científicas y Tecnológicas
[4] Instituto de Ingeniería y Agronomía, UNAJ
Av. Calchaquí 6200, 1888 Florencio Varela (Buenos Aires), Argentina

Resumen. El manejo de fallos es una preocupación creciente en HPC; en el futuro, se esperan mayores variedades y tasas de errores, intervalos de detección más largos y fallos silenciosos. Se proyecta que, en sistemas de exa-escala, los errores ocurran varias veces al día y se propaguen para generar desde caídas de procesos hasta corrupciones de resultados, con fallos no detectados en aplicaciones que siguen operando. En este trabajo se estudia una metodología de detección de fallos transitorios (SMCV) en aplicaciones MPI basada en replicación de software, asumiendo que la corrupción en los datos se manifiesta produciendo mensajes diferentes entre réplicas. SMCV permite obtener ejecuciones fiables con resultados correctos o, en su defecto, conducir al sistema a una parada segura. Se presenta una caracterización completa, definiendo formalmente el comportamiento frente a fallos y validándolo experimentalmente para mostrar la eficacia y viabilidad para detectar fallos transitorios en sistemas de HPC.

Palabras clave: fallos transitorios, detección, aplicaciones paralelas científicas, corrupción silenciosa de datos, HPC, inyección de fallos.

1 Introducción

El estancamiento en las frecuencias de reloj de los procesadores ha llevado a que la mejora de rendimiento se logre mediante el aumento en la cantidad de componentes. El hecho de escalar los sistemas conlleva el problema de la disminución de la tensión que, junto con los desafíos de la miniaturización submicrónica generan grandes incrementos en las tasas de fallos. Las interferencias electromagnéticas generan pulsos de corriente que alteran valores almacenados o en lógica combinatoria. Las variabilidades mayores en los procesos de fabricación ocasionan comportamientos inconsistentes, mientras que el envejecimiento vuelve más frecuentes los errores permanentes, y también los fallos múltiples han aumentado su probabilidad [1,2]. Debido a todo esto la fiabilidad de los sistemas se ha vuelto crítica, especialmente en

el área de de HPC con más de cientos de miles de *cores*. Estudios recientes en supercomputadoras modernas muestran que los tiempos medios entre fallos (MTBF-*Mean Time Between Failures*) son de pocas horas [3] y se estima que podrían llegar a unos 30 minutos en grandes aplicaciones paralelas en plataformas de exa-escala. En consecuencia, estas aplicaciones no podrán progresar eficientemente sin la ayuda adecuada [4,5]. La preocupación principal recae sobre los fallos silenciosos (SDC-*Silent Data Corruption*), habiendo aparecido numerosos reportes y estudios sobre sus probabilidades e impactos [2,6,7]. Al tener la capacidad de invalidar los resultados, los SDC crean graves problemas para la ciencia, que confía cada vez más en simulaciones de gran escala. Por todo esto, la mitigación de SDC es uno de los grandes retos de la resiliencia presente y futura.

Los SDC se generan en la forma de *bit-flips* (cambio en el valor de un bit) que afectan al almacenamiento o a los *cores*. Para detectarlos o corregirlos, los fabricantes colocan ECCs (*Error Correcting Code*) más potentes en la memoria, protegen los buses con bits de paridad y agregan redundancia a los circuitos de algunas unidades lógicas [8]. Sin embargo, resulta demasiado costoso incorporar redundancia hardware en los registros y en las ALUs del procesador [9].

El pequeño mercado de los supercomputadores, que requieren de alta fiabilidad, se puede satisfacer con soluciones de redundancia doble y triple para lograr detección y corrección, respectivamente. Aunque esto conlleva un alto costo, es preferible a tener resultados corruptos. Los SDC permanecen latentes hasta que los datos alterados son utilizados, siendo las latencias de detección dependientes de las aplicaciones.

El método estándar para manejar errores más utilizado en los sistemas paralelos actuales (en especial los que ejecutan aplicaciones MPI), es la realización de *checkpoints* periódicos. En caso de falla, el método de *Checkpoint/Restart* (C/R) relanza la aplicación desde el último *checkpoint*. Desafortunadamente, el *overhead* de C/R aumenta con el número de *cores*. Tomando en cuenta el tiempo requerido en C/R y re-ejecución, se podría desperdiciar gran cantidad de tiempo de cómputo útil si el MTBF es muy bajo. La situación empeora si el cómputo es fuertemente acoplado, porque un error en un nodo puede propagarse a los demás en microsegundos [1,10].

El modelo tradicional basado en C/R asume que la detección se produce casi de inmediato. Además, si el *checkpoint* almacenado contiene fallos no detectados, no se podrá efectuar la recuperación. Las pocas técnicas generales de detección introducen altos *overheads* en aplicaciones paralelas [2,11]. En base a lo anterior, se espera que aumenten los rangos de latencia de detección, incrementando el problema debido a los SDC. Tampoco hay mecanismos eficientes de contención, por lo que un fallo que afecta a una tarea puede resultar en la caída de la aplicación o en salidas incorrectas que, en el mejor caso, sólo se detectan cuando ha finalizado la ejecución y que son muy difíciles de corregir.

La replicación a nivel de procesos se ha mostrado como una alternativa fiable, pero para que sea atractiva en HPC aún debe resolver desafíos como minimizar el *overhead* temporal y de utilización de recursos, garantizar que los estados internos de las réplicas sean equivalentes (lo cual no es trivial, ya que se podrían ejecutar operaciones no determinísticas) y reducir el consumo energético. La forma clásica de detectar los SDC consiste en replicar ejecuciones y comparar sus resultados. RedMPI [2] lo hace a nivel de procesos aunque existen otros que lo hacen a nivel de *threads* [12]. También se han explorado soluciones que requieren menos recursos y relajan la

precisión, como la replicación aproximada, que proporciona límites superior e inferior para el resultado del cómputo [1].

En este contexto, en los últimos años se ha propuesto la metodología SMCV [13,14], diseñada para detectar fallos transitorios en HPC, específicamente para aplicaciones científicas que utilizan MPI en *clusters* de *multicores*. SMCV permite obtener ejecuciones fiables con resultados correctos o, al menos, reportar la ocurrencia de SDC y conducir al sistema a una parada segura luego de una latencia de detección acotada, ahorrando así un tiempo significativo, especialmente en aplicaciones largas.

El resto del documento se organiza de la siguiente manera. La Sección 2 repasa algunos conceptos básicos mientras que la Sección 3 describe trabajos relacionados. La Sección 4 presenta una caracterización de la estrategia SMCV, en la cual se define formalmente su comportamiento frente a fallos, su esfera de replicación (SoR-*Sphere of Replication*) y sus vulnerabilidades. La Sección 5 describe los experimentos realizados de inyección de fallos controlada de forma de validar el comportamiento definido y mostrar la eficacia y viabilidad de SMCV para detectar fallos transitorios en sistemas de HPC. Por último, la Sección 6 presenta las conclusiones y las líneas de trabajo futuras.

2 Conceptos básicos

Dependiendo de sus efectos sobre la ejecución de las aplicaciones, los fallos transitorios pueden clasificarse de la siguiente forma [13]:

- Error Latente (LE-*Latent Error*): afecta datos que no son utilizados posteriormente, por lo que no tiene impacto en los resultados.
- Error Detectado Irrecuperable (DUE-*Detected Unrecoverable Error*): causa una anomalía detectable para el software del sistema, sin posibilidad de recuperación; suele producir que la aplicación finalice de forma abrupta.
- Error por *Time-Out* (TO): el programa no finaliza dentro de un lapso de tiempo determinado.
- Corrupción Silenciosa de Datos (SDC): no es detectada por ningún nivel de software del sistema y sus efectos se propagan para producir la finalización con salida incorrecta. En aplicaciones paralelas con paso de mensajes, pueden causar: Corrupción de Datos Transmitidos (TDC-*Transmitted Data Corruption*), que afecta a datos que forman parte del contenido de mensajes a transmitir (si no se detecta se propaga a otros procesos); o Corrupción de Estado Final (FSC-*Final Status Corruption*), donde los datos alterados no se transmiten, pero se propagan localmente, corrompiendo el estado final del proceso afectado.

3 Trabajo relacionado

Las tecnologías actuales no puedan lidiar con SDC frecuentes. Las soluciones algorítmicas existentes [15] sólo se pueden aplicar a *kernels* específicos, por lo que hay que evaluar mecanismos que permitan tratar con los errores que escapan de su alcance. En tanto, las estrategias de detección que se basan en el compilador o en

software de tiempo de ejecución se pueden aplicar a cualquier código aunque resultan más complejas.

La contención busca evitar que el daño causado por el fallo se propague a otros nodos o que corrompa datos de un *checkpoint*, volviendo imposible la recuperación [1]. En [16] se propone redundancia en sistemas de HPC, lo que permite incrementar la disponibilidad del sistema y ofrece un compromiso entre cantidad y calidad de componentes. En [17] se muestra que la replicación es más eficiente que C/R en situaciones donde MTBF es bajo y el *overhead* temporal de C/R es alto. Soluciones de redundancia por software se enfocan en replicación a nivel de *threads* [12], de procesos [9] y de estado de máquina para eliminar la necesidad de hardware costoso.

MR-MPI [19] es otra propuesta para redundancia transparente en HPC, que ofrece replicación parcial (sólo se replican algunos procesos); se puede complementar con C/R en los procesos no replicados [20,21].

rMPI [18] es un protocolo para ejecución redundante de aplicaciones MPI, enfocado en fallos que causan la parada del sistema, y que utiliza la capa de *profiling* para realizar interposición de funciones MPI. Cada nodo tiene una réplica, de modo que ante un fallo permanente, el nodo redundante continúa sin interrupciones; la aplicación falla si fallan dos réplicas correspondientes. La redundancia escala, es decir, la probabilidad de fallo simultánea entre un nodo y su réplica decrece cuando aumenta la cantidad de nodos, a costa de duplicar la cantidad de recursos utilizados y cuadruplicar el número de mensajes. En tanto, RedMPI [2] es una bliblioteca MPI que aprovecha la replicación de procesos de rMPI para detectar y corregir SDC, comparando en el receptor mensajes enviados por emisores replicados. Implementa una optimización basada en *hashing* para evitar enviar todos los mensajes y comparar sus contenidos completos. No requiere modificaciones al código de la aplicación y asegura que las réplicas se ejecutan de manera determinística. Los resultados muestran que puede proteger las aplicaciones incluso con tasas de fallos altas con *overheads* temporales menores a 30%, por lo que potencialmente puede ser utilizado en sistemas de gran escala. Una contribución de [2] es el análisis de la propagación de los SDC entre nodos a través de las comunicaciones MPI, mostrando que incluso un único fallo transitorio puede tener un profundo efecto sobre la aplicación, causando un patrón de corrupción en cascada hacia todos los demás procesos.

Al igual que SMCV, al enfocarse en los mensajes, RedMPI monitorea los datos más críticos para la aplicación; la corrección en las comunicaciones es necesaria para la corrección de la salida. Como el SDC puede afectar a datos que no se comunican inmediatamente, el fallo es detectado al momento de la trasmisión. Sin embargo, a diferencia de SMCV, RedMPI realiza la validación del lado del receptor. Esto se debe a que, en el emisor, todas las réplicas se deben comunicar con las demás para verificar internamente sus contenidos antes de enviar el mensaje. Esto incurre en *overhead* y latencia adicionales, ya que el receptor pierde todo ese tiempo antes de poder continuar. Como SMCV replica a nivel de *threads* y no de procesos, no necesita hacer circular mensajes entre los emisores para validar. Al enviar sólo un mensaje luego de la validación, no congestiona la red. Al igual que SMCV, RedMPI permite que, aún sin corrección, la corrupción quede confinada en un proceso. También permite personalizar el *mapping* de las réplicas en el mismo nodo físico que los procesos nativos (o en sus vecinos con menor latencia de red).

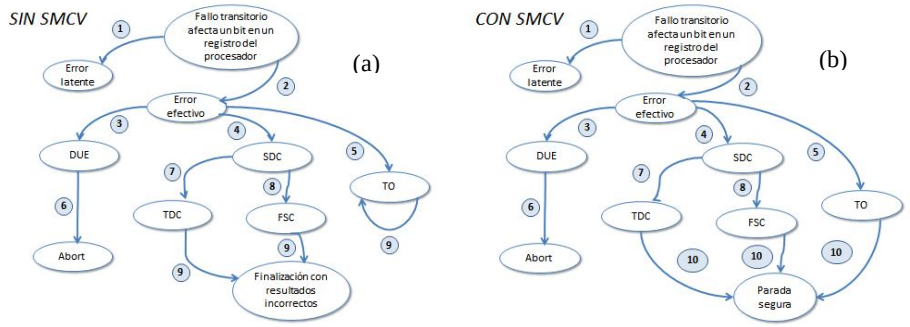
4 Caracterización de la metodología SMCV

4.1 Breve revisión de la metodología SMCV

SMCV es una estrategia de detección basada en la validación de contenidos de mensajes que se van a enviar entre procesos de aplicaciones paralelas determinísticas. Está diseñado para detectar los fallos que causan SDC (en sus dos formas) y TO. SMCV duplica en un *thread* cada proceso de la aplicación, requiriendo mecanismos de sincronización entre ambas réplicas concurrentes. Cuando se va a realizar una comunicación, el *thread* detiene su ejecución a la espera de que su réplica alcance el mismo punto, y todos los campos del mensaje, calculados por ambas replicas, se comparan en la búsqueda de diferencias. En caso de coincidencia, sólo uno de los hilos envía el mensaje, evitando que un error se propague a otros procesos sin consumir ancho de banda adicional. El receptor se sincroniza con su réplica, recibe el mensaje y realiza una copia para ella, continuando ambas con su ejecución. Al concluir, se verifican los resultados para detectar fallos que se puedan haber propagado localmente hasta el final de la aplicación.

4.2 Comportamiento frente a fallos

En esta sección se define el comportamiento de la metodología de detección. En la Fig. 1(a) se muestra un diagrama de los estados posibles de una ejecución cuando no hay ninguna estrategia implementada, mientras que en la Fig. 1(b) se ve el mismo diagrama cuando se aplica SMCV. Las elipses representan estados y los arcos representan eventos que producen transiciones de un estado a otro. Las transiciones aparecen numeradas, por lo que se describe cada una de ellas.



1. El bit afectado no se utiliza.
2. El bit afectado es utilizado por la aplicación.
3. El bit alterado afecta datos controlados por el sistema operativo.
4. El bit alterado afecta datos de la aplicación del usuario.
5. El bit alterado produce que la aplicación no responda en un tiempo límite.
6. El sistema operativo detecta el fallo y aborta la aplicación.
7. El dato afectado se transmite a otro proceso de la aplicación paralela.
8. El dato afectado sólo es utilizado por el proceso local.
9. Transcurre tiempo de ejecución.
10. SMCV detecta el fallo al cabo de un tiempo y conduce a una parada segura.

Fig. 1(a) Diagrama de estados de la ejecución sin estrategia de detección de fallos.
(b) Ejecución aplicando la estrategia de detección SMCV.

4.3 Esfera de replicación

La esfera de replicación [9] es un concepto comúnmente aceptado para describir el dominio lógico de ejecución redundante de una técnica particular y especificar los límites para la detección de fallos. Todos los datos que ingresan en la SoR son replicados, la ejecución dentro de su ámbito es redundante de alguna forma y los datos de salida son comparados para asegurar su corrección antes de abandonarla. Toda la ejecución por fuera de la SoR no está cubierta frente a fallos y debe ser protegida de otra manera. El concepto original de SoR se utilizó para definir los límites de la fiabilidad en diseños de hardware redundante, ubicándola alrededor de unidades específicas. Sin embargo, su aplicación resulta inadecuada para las propuestas implementadas en software, a pesar de lo cual algunas soluciones que utilizan el compilador para insertar instrucciones redundantes han intentado imitar una SoR centrada en hardware [22]. En tanto, el paradigma de detección de fallos centrado en software ubica la SoR alrededor de capas de software [9]; esto pone de manifiesto que, aunque los fallos afectan al hardware, sólo son relevantes los que influyen sobre la corrección de la aplicación, mientras que los que permanecen latentes pueden ignorarse sin riesgo. Sin embargo, esto tiene la desventaja de retardar la detección hasta que se produce certeza del error por datos inválidos que salen de la SoR, lo que implica que un fallo puede permanecer latente por tiempo indeterminado.

SMCV es una técnica de software y, en consecuencia, adopta una SoR centrada en software. Su objetivo es detectar fallos que afectan datos que se manipulan dentro de los registros del procesador, que constituyen la parte más vulnerable del computador, debido a la dificultad de implementar protección por hardware. Como se explicó, SMCV replica en un *thread* el cómputo que realiza cada proceso de la aplicación paralela. Cada *thread* opera sobre una copia local de los datos de entrada, que se genera para que pueda computar de forma independiente de su réplica. Por lo tanto, la SoR se encuentra alrededor de la aplicación del usuario y sus datos, sin incluir al sistema operativo ni a la biblioteca de comunicaciones. Aunque la memoria se halla por fuera de la SoR de SMCV, es aconsejable no utilizar variables globales, ya que constituyen puntos de falla centralizados. Si ocurre un fallo que altera su valor, ambos *threads* redundantes utilizarán el valor erróneo y, si no ocurre otro fallo, SMCV no detectará ningún error.

4.4 Fallos múltiples y vulnerabilidades

La mayoría de las propuestas existentes son capaces de detectar fallos si se asume que ocurre un único *bit-flip* a lo largo de la ejecución, pero no son tan efectivas para fallos que afectan a múltiples bits. Afortunadamente, existen sólo dos situaciones en las que fallos múltiples se pueden combinar para causar inconvenientes. La primera es aquella en la que el mismo bit es alterado en ambas réplicas, por lo que la comparación resulta correcta y el fallo no se detecta. La segunda se da cuando un fallo afecta a una de las réplicas, y el resultado de la verificación también es alterado, enmascarando el fallo original. Sin embargo, estas combinaciones tienen una probabilidad de ocurrencia muy baja, por lo que pueden ser ignoradas sin riesgos serios. Todas las demás combinaciones de fallos múltiples son detectadas como fallos simples ni bien se registra la primera diferencia en una comprobación [22]. SMCV es

capaz de detectar cualquier fallo transitorio simple que cause SDC o TO, pero no soporta fallos múltiples relacionados.

Todas las técnicas de tolerancia a fallos tienen vulnerabilidades, es decir, circunstancias en las cuales no son capaces de detectar los fallos que influyen efectivamente sobre la ejecución. Las características de diseño de una estrategia y las pruebas a las que es sometida (usualmente mediante inyección de fallos) deben permitir explicitar esas vulnerabilidades.

Las vulnerabilidades están normalmente asociadas con fallos que afectan al propio mecanismo de detección [9], y SMCV no es la excepción. SMCV minimiza el retardo entre la comprobación y la utilización de los valores validados, ya que realiza la verificación al momento de utilizar los datos en un mensaje; de esta manera reduce la probabilidad de fallo durante el lapso que transcurre entre ambos eventos (como ocurre en [22]); una vez que los datos están en el *buffer* de salida se encuentran fuera de la SoR. En tanto, la comprobación de valores a enviar resulta un punto centralizado de falla. Si la comprobación resulta incorrecta luego de una ejecución correcta, se ha producido un falso positivo y se genera una parada segura cuando en realidad el problema fue introducido por el mismo detector. Esta vulnerabilidad se puede mejorar duplicando la comparación; sin embargo, a pesar de no ser completamente fiable, la redundancia parcial en general es suficiente para cubrir los requerimientos de los usuarios [9]. En tanto, si la comprobación resulta correcta tras una ejecución incorrecta, el fallo ha quedado oculto tras la comprobación, debido a un segundo fallo. Como se mencionó, SMCV no puede lidiar con esta situación, aunque la probabilidad de que ocurra es extremadamente baja [22]. Además, resulta importante considerar el hecho de que SMCV es capaz de detectar como TO otros fallos que constituirían vulnerabilidades si no se contara con dicho mecanismo. Por ejemplo, si un código de operación es modificado, de modo que la instrucción resultante es el envío de un mensaje, o si ocurre un fallo durante la ejecución de la herramienta, ambas réplicas separan sus flujos de ejecución. Ante el envío de un mensaje por parte de una de ellas, no se sincronizan adecuadamente, por lo que el fallo se detecta al transcurrir un lapso mayor al determinado.

5 Validación de la eficacia de detección

Se realizaron una serie de pruebas para validar la eficacia de detección de SMCV. La aplicación utilizada fue una multiplicación de matrices paralela MPI (`C=A×B`) bajo el paradigma *Master*/*Worker*, en la que el *Master* toma parte en el cómputo del resultado [13]. La aplicación opera de siguiente forma:

- El proceso *Master* divide la matriz `A` entre todos los *Workers*, y, mediante la función `MPI_Scatter`, envía un trozo a cada uno, manteniendo él mismo uno para calcular su parte de la matriz resultado.
- El *Master* envía a cada *Worker* una copia de la matriz B completa mediante la función `MPI_Broadcast`.
- Todos los procesos computan su trozo correspondiente de la matriz `C`, enviando luego lo que han calculado al proceso *Master* mediante la función `MPI_Gather`.

- El *Master* construye la matriz `C` a partir de lo que los *Workers* le han enviado y lo que él mismo ha calculado.

Para la validación, se adaptó la aplicación para integrarla con la funcionalidad de la herramienta SMCV de la forma descripta en [14]. Para esto se requiere modificar el código fuente de la aplicación y la posterior recompilación. El experimento consistió en inyectar fallos, controladamente, en varios puntos de la aplicación, por medio de la herramienta de depuración GDB[1]. Para esto, se inserta un *breakpoint* en la ejecución de uno de los procesos, se modifica el valor de una variable y se retoma la ejecución; de esta manera se simula un *bit-flip* en un registro del procesador, ya que la corrupción se manifiesta si se puede observar una diferencia entre los estados de memoria de las réplicas. Pese a que un fallo transitorio puede ocurrir aleatoriamente en cualquier lugar y momento de la ejecución, para la inyección controlada se seleccionaron puntos significativos, tanto en el cómputo realizado por el *Master* como por los *Workers*.

Para los experimentos se utilizaron cinco procesos (un *Master* y cuatro *Workers*) y matrices cuadradas de 10 x 10, por lo que cada uno de los cinco procesos calcula dos filas de la matriz `C`. Aunque este tamaño no requiere realmente de una ejecución paralela, se utiliza con la única finalidad de mostrar las consecuencias de la inyección de fallos y la capacidad de detección de SMCV. La plataforma experimental se compone de una CPU Intel Core i5-2310 2.90Ghz con 6MB de memoria caché L3 y 8GB RAM, y el sistema operativo es GNU/Linux Ubuntu 14.04.

En la Fig. 2(a) se muestra una ejecución normal de la aplicación, sin inyección de fallos. El conteo inicial corresponde al lapso utilizado para adjuntar el depurador a alguno de los procesos, de manera de simular un fallo que afecta un dato utilizado por dicho proceso. En la Fig. 2(b) se muestra la forma en la que se adjunta el depurador para realizar los experimentos de inyección.

(a)

```
diego@Lidl137:~/Dropbox/diego/Para trabajo de Especialización/Experimentos$ mpirun -np 5 mm-SMCV 10
PID 4583 on 0 ready for attach
PID 4586 on 3 ready for attach
PID 4584 on 1 ready for attach
PID 4587 on 4 ready for attach
Restan 10 segundos...
PID 4585 on 2 ready for attach
Restan 9 segundos...
Restan 8 segundos...
Restan 7 segundos...
Restan 6 segundos...
Restan 5 segundos...
Restan 4 segundos...
Restan 3 segundos...
Restan 2 segundos...
Restan 1 segundos...
MM-SMCV;5;10;11.065258;11.049720;0.015538
```

(b)

```
diego@Lidl137:~/Dropbox/diego/Para trabajo de Especialización/Experimentos$ sudo gdb -q -pid=4746
Adjuntando a process 4746
Leyendo símbolos desde /home/diego/Dropbox/diego/Para trabajo de Especialización/Experimentos/mm-SMCV...hecho.
```

Fig. 2(a) Salida de una ejecución sin fallos. Se muestra el tiempo para adjuntar el depurador. **(b)** Ejemplo de cómo adjuntar el depurador para inyectar fallos.

En la Fig. 3(a) se muestra el procedimiento realizado para inyectar un fallo durante la operación del *Master*, en uno de los primeros 20 elementos de la matriz `A` (los que conserva para su cómputo local), después de la ejecución de la función `MPI_Scatter` pero antes de la multiplicación. Esta situación simula la ocurrencia de un fallo que corrompe un dato que interviene en el cómputo del resultado, pero nunca es transmitido a otro proceso de la aplicación, produciendo FSC. En la Fig. 3(b) se ve la salida de la aplicación, con detección del error y parada segura.

[1] GDB se encuentra disponible en www.gnu.org/software/gdb/

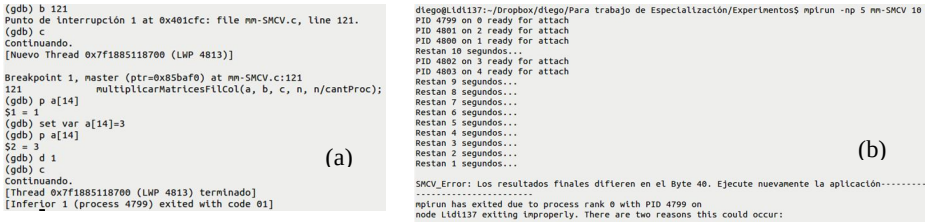


Fig. 3(a) Inyección de un fallo que causa FSC. **(b)** Salida con la detección de SMCV.

En la Fig. 4(a) se muestra la inyección de un fallo durante la operación de un *Worker* en un elemento de la matriz `B`, después de la ejecución de `MPI_Broadcast` pero antes de la multiplicación. De esta forma, se simula la corrupción de un dato que interviene en el cálculo que realiza ese *Worker*. Los resultados de estos cálculos se transmiten al *Master* en el `MPI_Gather` posterior, por lo que el resultado incorrecto (calculado a partir del valor alterado) es detectado como TDC. La Fig. 4(b) nuevamente muestra la salida de la aplicación, con detección del error y parada segura. Como el fallo ha causado TDC, el mensaje es diferente del caso anterior.

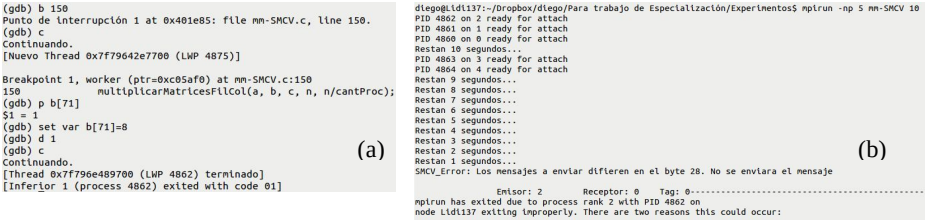


Fig. 4 (a) Inyección de un fallo que causa TDC. **(b)** Salida con la detección de SMCV.

En la Fig. 5(a) se muestra la inyección de un fallo en un elemento de la matriz `C` en uno de los *Workers*. La multiplicación posterior sobrescribe el valor alterado, por lo que el fallo produce un LE. En consecuencia, en la Fig. 5(b) se ve que la salida es normal y correcta.

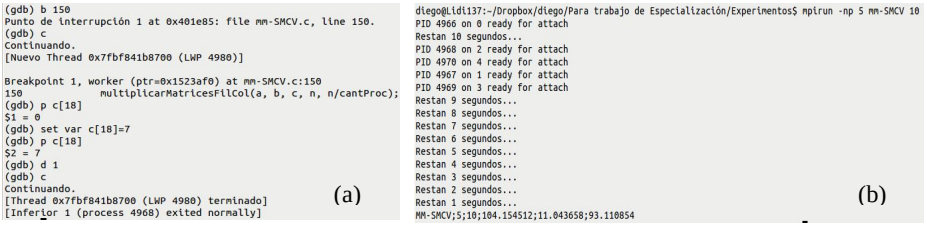


Fig. 5(a) Inyección de un fallo que causa LE. **(b)** Salida de la ejecución.

Finalmente, en la Fig. 6 se observa la salida de la aplicación cuando ha ocurrido un fallo que produce TO; se puede ver la detección y parada segura. En este caso, el fallo se inyecta en una variable que actúa como índice, produciendo que una de las réplicas del *Worker* recomience su cómputo cuando ya ha realizado parte de su tarea. Esto ocasiona un desfase temporal entre los progresos de ambos hilos redundantes, que es

detectado como un error por TO. La consecuencia ideal de un fallo que produzca TO es que el proceso ingrese a un bucle infinito, pero en la aplicación seleccionada no se puede provocar este comportamiento mediante un fallo simple.

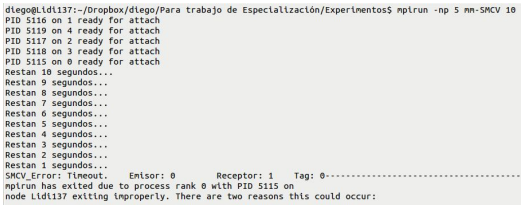


Fig. 6 Salida con la detección de TO por SMCV.

Es importante destacar que el lapso transcurrido, el cual se asume la ocurrencia de un fallo, es configurable. No existe un valor óptimo, sino que depende de la aplicación particular. Para clarificar esto, se debe considerar que la detección por TO parte de la premisa de que, en una aplicación que se ejecuta sobre un sistema homogéneo dedicado, los tiempos de ejecución de dos réplicas que realizan el mismo cómputo deben ser similares [14]. Por lo tanto, una asimetría notoria en los tiempos de procesamiento supone que ambas réplicas han separado sus flujos a causa de un fallo silencioso. Por lo tanto, se debe configurar el lapso de TO de acuerdo a lo esperable para la aplicación: si el lapso resulta muy alto, aumentará la latencia de detección; si es demasiado bajo, una pequeña asimetría en los tiempos de cómputo resultará en la detección de un falso positivo. En el caso de prueba anterior, el fallo inyectado sólo provoca una demora anormal en la sincronización. Se configuró adrede un lapso breve para mostrar que el mecanismo es capaz de reaccionar frente a este evento. Sin embargo, si uno de los procesos ingresara en un bucle infinito, SMCV detectaría efectivamente un error.

6 Conclusiones y trabajo futuro

A medida que los sistemas de HPC escalan y aumenta la probabilidad de fallos de nodo y SDC, se vuelve aún más crítica la necesidad de proteger datos y obtener disponibilidad a bajo costo. La redundancia es una solución viable para la detección de SDC en el ámbito de HPC. El hecho de que un único SDC produzca efectos profundos en todos los procesos que se comunican permite concluir que la protección de aplicaciones a nivel de mensajes de MPI es un método factible y efectivo para detectar, aislar y prevenir corrupción de datos posterior.

A partir de las pruebas realizadas, se concluye que SMCV es capaz de detectar los fallos que afectan contenidos de mensajes (TDC), notificando al usuario y produciendo parada segura para que la corrupción no se pueda propagar. Por otra parte, los fallos que afectan a datos que se mantienen para cómputo local, y los que ocurren en la fase final (correspondientes a la fracción FSC) son detectados durante la comparación de resultados. En tanto, los fallos que producen asimetrías considerables en los tiempos de cómputo de las réplicas son detectadas por medio del mecanismo de TO.

El trabajo futuro consiste en completar una metodología tolerante a fallos transitorios, incorporando a la detección un mecanismo de recuperación basado en múltiples *checkpoints* incrementales distribuidos, de forma de poder almacenar en un proceso la información sobre el fallo ocurrido en otro, y así determinar si el último *checkpoint* es válido o si hay que retroceder a uno anterior para recuperar [10].

Referencias

1. Cappello, F., Geist, A., Gropp, W., Kale, S., Kramer, B., & Snir, M.: Toward exascale resilience: 2014 update. Supercomputing frontiers and innovations, 1(1) (2014).
2. Fiala, D., Mueller, F., Engelmann, C., Riesen, R., Ferreira, K., & Brightwell, R.: Detection and correction of silent data corruption for large-scale high-performance computing. In Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (p. 78). IEEE Computer Society Press (2012).
3. Zheng, Z., Yu, L., Tang, W., Lan, Z., Gupta, R., Desai, N., & Buettner, D.: Co-analysis of RAS log and job log on Blue Gene/P. In Parallel & Distributed Processing Symposium (IPDPS), IEEE International (pp. 840-851) (2011).
4. Borkar, S., & Chien, A.: The future of microprocessors.Communications of the ACM, 54(5), 67-77 (2011).
5. Moody, A., Bronevetsky, G., Mohror, K., & De Supinski, B. R.: Design, modeling, and evaluation of a scalable multi-level checkpointing system. In High Performance Computing, Networking, Storage and Analysis (SC), 2010 International Conference for (pp. 1-11). IEEE (2010).
6. Elliott, J., Hoemmen, M., & Mueller, F.: Evaluating the impact of SDC on the GMRES iterative solver. In Parallel and Distributed Processing Symposium, 2014 IEEE 28th International (pp. 1193-1202). IEEE (2014).
7. Li, D., Vetter, J. S., & Yu, W.: Classifying soft error vulnerabilities in extreme-scale scientific applications using a binary instrumentation tool. In Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (p. 57). IEEE Computer Society Press (2012).
8. Snir, M., Wisniewski, R. W., Abraham, J. A., Adve, S. V., Bagchi, S., Balaji, P., ... & Van Hensbergen, E.: Addressing failures in exascale computing. International Journal of High Performance Computing Applications (2014).
9. Shye, A., Blomstedt, J., Moseley, T., Reddi, V. J., Connors, D. A.: PLR: A software approach to transient fault tolerance for multicore architectures; IEEE Transactions on Dependable and Secure Computing. 6(2), pp. 135-148 (2009).
10. Lu, G., Zheng, Z., & Chien, A.: When is multi-version checkpointing needed? In Proceedings of the 3rd Workshop on Fault-tolerance for HPC at extreme scale (pp. 49-56). ACM (2013).
11. Hari, S. K. S., Adve, S. V., & Naeimi, H.: Low-cost program-level detectors for reducing silent data corruptions. In Dependable Systems and Networks (DSN), 2012 42nd Annual IEEE/IFIP International Conference on (pp. 1-12). IEEE (2012).
12. Yalcin, G., Unsal, O. S., & Cristal, A.: Fault tolerance for multi-threaded applications by leveraging hardware transactional memory. InProceedings of the ACM International Conference on Computing Frontiers (p. 4). ACM (2013).
13. Montezanti, D., Frati, F.E., Rexachs, D., Luque, E., Naiouf, M.R., De Giusti, A.: SMCV: a Methodology for Detecting Transient Faults in Multicore Clusters.; CLEI Electron. J. 15(3), pp. 1-11 (2012).
14. Montezanti, D., Rucci, E., Rexachs, D., Luque, E., Naiouf, M.R., De Giusti, A.: A tool for detecting transient faults in execution of parallel scientific applications on multicore clusters; Journal of Computer Science & Technology , 14(1), pp. 32-38 (2014).
15. Chen, Z.: Algorithm-based recovery for iterative methods without checkpointing. In Proceedings of the 20th international symposium on High performance distributed computing (pp. 73-84). ACM (2011).
16. Engelmann, C., Ong, H., & Scott, S. L.: The case for modular redundancy in large-scale high performance computing systems. In Proceedings of the IASTED International Conference (Vol. 641, p. 046) (2009).
17. Ferreira, K., Stearley, J., Laros III, J. H., Oldfield, R., Pedretti, K., Brightwell, R., ... & Arnold, D.: Evaluating the viability of process replication reliability for exascale systems. In Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis (p. 44). ACM (2011).
18. Ferreira, K., Riesen, R., Oldfield, R., Stearley, J., Laros, J., Pedretti, K., & Brightwell, T.: rMPI: increasing fault resiliency in a message-passing environment. Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2011-2488 (2011).
19. Engelmann, C., & Böhm, S.: Redundant execution of HPC applications with MR-MPI. In Proceedings of the 10th IASTED International Conference on Parallel and Distributed Computing and Networks (PDCN) (pp. 15-17) (2011).
20. Elliott, J., Kharbas, K., Fiala, D., Mueller, F., Ferreira, K., & Engelmann, C.: Combining partial redundancy and checkpointing for HPC. InDistributed Computing Systems (ICDCS), 2012 IEEE 32nd International Conference on (pp. 615-626). IEEE (2012).
21. Ni, X., Meneses, E., Jain, N., & Kalé, L. V.: ACR: automatic checkpoint/restart for soft and hard error protection. In Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (p. 7). ACM (2013).
22. Reis, G. A., Chang, J., Vachharajani, N., Rangan, R., August, D. I.: SWIFT: Software Implemented Fault Tolerance. In: Proceedings of the International Symposium on Code generation and optimization, pp. 243–254. IEEE Press, Washington DC (2005).

Including accurate user estimates in HPC schedulers: an empirical analysis

Nestor Rocchetti, Santiago Iturriaga, and Sergio Nesmachnow

[1] Universidad de la República
Montevideo, Uruguay
{nrocchetti, siturria, sergion}@fing.edu.uy

Abstract. This article focuses on the problem of dealing with low accuracy of job runtime estimates provided by users of high performance computing systems. The main goal of the study is to evaluate the benefits on the system utilization of providing accurate estimations, in order to motivate users to make an effort to provide better estimates. We propose the Penalty Scheduling Policy for including information about user estimates. The experimental evaluation is performed over realistic workload and scenarios, and validated by the use of a job scheduler simulator. We simulated different static and dynamic scenarios, which emulate diverse user behavior regarding the estimation of jobs runtime. Results demonstrate that the accuracy of users runtime estimates influences the waiting time of jobs. Under our proposed policy, in a scenario where users improve their estimates, waiting time of users with high accuracy can be up to *2.43 times* lower than users with the lowest accuracy.

Keywords: high performance computing, scheduling, execution time estimation, quality of service.

1 Introduction

Parallel supercomputers are high-end machines designed to support the execution of parallel jobs [1]. Nowadays, supercomputers have become a common commodity in scientific oriented companies and research institutions, especially those working on High Performance Computing (HPC). Along with the development of HPC infrastructures, the main trend has been using commercial cluster management software suites. These software suites offer a wide variety of features, which include queue management, process prioritization, and scheduling algorithms [2].

Due to the increasing usage of supercomputers, job scheduling has become a critical task, where small differences in policies can result in great changes in resource utilization, and in performance [3]. The most popular scheduling policy used in batch schedulers is first-come, first-served (FCFS) [2]. This scheduling policy often comes in combination with a backfilling method called EASY-Backfilling. The idea of this method is to select small jobs (i.e., jobs with low number of requested cores or walltime) for execution before the time they were supposed to, whenever holes of idle

resources appear [4]. Backfill systems relay on users job runtime estimates to accomplish their task.

Execution time estimation has a significant impact on how a scheduler treats different jobs, and on general performance [4]. The inaccuracy of user estimates worsens the overall performance of the parallel system [5]. For this reason, many studies have been performed in order to improve runtime estimates, to make a positive impact on both system-related and user-related performance metrics.

This article focuses on the problem of dealing with low accuracy of job runtime estimates provided by users. The main goal of the study is to evaluate the benefits of providing accurate estimation, in order to motivate users to make an effort to better estimate the system utilization.

The main contributions of this article are: i) the study of the impact of user runtime estimations in the system utilization for current HPC infrastructures; ii) the design and implementation of a novel scheduling strategy, named *Penalty Scheduling Policy* (PSP), which prioritizes jobs from users that provide good estimates on jobs runtime; and iii) the experimental evaluation of PSP using realistic workloads, on both static and dynamic scenarios, which emulate diverse user behavior regarding the estimation of jobs runtime.

We present an empirical evaluation of PSP under five scenarios that represent different user behaviors regarding the estimated runtime of jobs. For each scenario, four simulations are performed considering different workload patterns that models the real situation of our HPC infrastructure, Cluster FING. Then, we analyze the impact of their accuracy on the queuing time of their jobs.

The paper is organized as follows. Section 2 introduces some general concepts about scheduling. A review of related work is presented in Section 3. Section 4 describes the proposed PSP algorithm. Section 5 presents the workload analysis and the problem instances characteristics. Then, the experimental evaluation of PSP is presented in section 6. Finally, section 7 presents the conclusions and formulates the main lines for future work.

2 Background

This section presents a brief description of Cluster FING at Facultad de Ingeniería and the SLURM simulator [6], the tool used to perform the scheduling evaluation.

2.1 Cluster FING

Cluster description. Cluster FING [7] is the HPC infrastructure at Facultad de Ingeniería, Universidad de la República, Uruguay. It is an heterogeneous cluster of computing resources with 1672 cores, which has been operational since 2008, with a steady growth in components. It is used mostly for the batch execution of scientific and engineering computing jobs.

Job scheduling. Cluster FING uses Maui [8] for job administration. Maui is a policy engine to manage resources (such as processors, memory, and disk) that are assigned to jobs. It also provides other features like mechanisms for resource usage

optimization, monitor system performance, help diagnose problems, and general system manage. The default behavior of Maui is defined by a first-come, first-served (FCFS) batch scheduler, plus EASY Backfilling [8].

FCFS is a queue policy where the jobs are attended in the same order that they arrive: the first job to arrive is the first to get access to the requested resources. Backfilling is a policy that requires users to estimate the runtime of their jobs. Provided this information of runtime, short (runtime) jobs are allowed to execute before a larger job at front of the queue [9]. The EASY Backfilling algorithm only moves ahead jobs that do not delay the job at the head of the queue.

2.2 The SLURM workload manger

SLURM (Simple Linux Utility for Resource Management) [10] is an open-source workload manager designed for clusters running Linux. SLURM provides the basic workload manager tasks for allocating resources to users for a requested amount of time. It also provides tools for starting, executing, and monitoring jobs on a set of allocated nodes. Besides that, it manages a queue of pending work that is configured by the administrators of the application/infrastructure.

SLURM design is modular, including many optional built-in plugins. Two relevant plugins that are used in this work are *SLURM Priority Plugin API* and *SLURM Accounting Storage Plugin API*. SLURM Priority Plugin API allows computing the priority of the queued jobs in every iteration. The default configuration of this plugin is the basic implementation, which provides a basic FIFO job priority. It also comes with a multifactor job priority plugin that can be configured easily. SLURM Accounting Storage Plugin API allows the storage of accounting data collected during the execution of the scheduler, it can be configured to use a MySQL database in order to store accounting data for future processing. We use SLURM Accounting Storage Plugin API to store the accounting data in the simulations performed to evaluate the priority scheduler considering user runtime estimates proposed in this work.

In this work, we have adapted SLURM Priority Plugin API to implement our proposed priority policy. It is important to state that, in SLURM, the larger the priority number, the higher the job will be positioned in the queue, and the sooner the job will be executed.

2.3 The SLURM simulator

The SLURM simulator [6] is a job trace simulator that uses the SLURM scheduler as the simulation tool with minor SLURM code changes. The implementation of the simulator was left outside the SLURM source code; this way the simulation mode can be used with future releases of the scheduler. The simulator contains two programs, external to SLURM: *sim_mgr*, the simulation manager, which keeps control of the simulation time and *sim_lib*, the simulation library, which captures time-related calls and synchronizes with *sim_mgr* for sleep calls or getting simulation time.

A workload generator for SLURM is provided with the simulator. This workload generator, with slight changes in its source code, is used in this article to create the synthetic workloads used in the experimental evaluation of the proposed scheduler. The workload generated is based on real workload registered on Cluster FING. The hardware infrastructure used on the simulations is also based on Cluster FING (see details about the problem instances on Section 5).

3 Related work

This section describes the related work about analyzing user runtime estimates, its impact on job scheduling, and proposed techniques to improve the accuracy of the estimations.

Several relevant related works reported that user runtime estimates of jobs are usually inaccurate. For example, Cirne and Berman [1] showed that in four traces of different supercomputers, 50% to 60% of jobs made use of less than 20% of their requested time. Other features were also reported, for example the relation between failed jobs and accuracy, and between job length and accuracy.

The impact of user runtime estimates has been a matter of study in many articles. As stated by Tsafrir [5], some of the studies performed gave surprising, counterintuitive results. While some researchers found that inaccurate estimates are usually preferable over accurate ones, other studies show that performance is insensitive to accuracy of users runtime estimates [3,11 –14]. Tsafrir reported results showing that performance is affected by the quality of users runtime estimates.

The empirical study by Tang et al. [3] showed that FCFS is not sensitive to user runtime estimates. However, using accurate runtime estimates improve performance on scheduling policies that give precedence to short jobs, like Shortest Job First. It is also presented a scheme that uses historical information about the quality of estimates of both user and project scopes, to redefine the runtime estimate of a given job. The proposed adjusting scheme is transparent to users and easy to deploy.

In Iturriaga et al. [15], we studied the problem of energy consumption in heterogeneous computing scenarios proposing novel scheduling algorithms and reporting their experimental evaluation performed over realistic workloads and scenarios. We analyzed three real-world task workloads and proposed a workload generation model considering uncertainties. We computed improvements of up to 32% in computing performance and up to 18% in energy consumption.

In this line of work, this article focuses on analyzing the impact of users improving their runtime estimates when using the proposed PSP in HPC clusters. PSP is based on lowering the priority of jobs submitted by users whose runtime estimates have been inaccurate in the past, as it is described in the following section.

4 The proposed Penalty Scheduling Policy

The penalty policy applied in PSP consists in affecting the priority of jobs according to the historical precision of runtime estimates of the users.

We define the accuracy of a users job runtime estimate as $A = \frac{t_{run}}{t_{req}}$, where $t_{run}$is the real runtime of the job, and t_{req} is the requested time. Accuracy can take values between 0.0 and 1.0, thus the average accuracy is also between that interval. The bigger the average accuracy, the better the user is when estimating runtime, and the PSP method will assign higher priority to the users newly submitted jobs.

To affect the priority in the PSP scheduler, the accuracy of users estimates is used. We used a dynamic update scheme for estimating the accuracy of users, by computing the average deviations (i.e., ratio) between estimated time and real execution time for the last ten completed jobs for each user.

Table 1 shows the intervals used to assign priority to jobs. The priority is a number between 1 and 5, a higher number means that the jobs is closer to the head of the queue. For example, a job whose user has an accuracy of 0.35 will have a priority of 2. This priority is first calculated when the job is submitted, and it is updated every time a new job is submitted or when releasing resources (i.e., a job ends).

Table 1. Intervals for accuracy of estimates and priorities for each tag names.

tag name	***accuracy interval***	***priority***
a1	[0.0,0.2)	1
a2	[0.2,0.4)	2
a3	[0.4,0.6)	3
a4	[0.6,0.8)	4
a5	[0.8,1.0]	5

We consider that a job runtime estimate is "good" when its accuracy is over 0.6, under that it is considered a poor quality estimate. That consideration is based on the study of workload trace at Cluster FING, in which the users with coefficient of accuracy of estimates higher than 0.6 is just 4% of the total platform users.

Algorithm 1 presents a pseudocode for the implementation of the proposed scheduler into SLURM.

Algorithm 1. PSP implementation in SLURM

```
priority_thread_tasks()
  while (true)
    waitEvent(job_completion, job_submission, time_lap, ...);
    jobs_list.computeNewPriority();
  end;
end;
scheduling_thread_tasks()
  while(true)
  //scheduling thread tasks
  end;
end;
main()
…
scheduling_thread.create();
priority_thread.create();
…
end;
```

We included our code in the Multifactor implementation of SLURM Scheduler Priority Plugin API. This priority API is used by the *Job Manager*, which is the component that accepts jobs requests and includes pending jobs in a priority ordered queue. The function computeNewPriority() called by the priority_thread communicates with that API and updates the priority of all jobs in pending state based on data retrieved from the database in which job accounting information is stored. This function is called periodically and when there is a change in a job state that may permit another job to begin execution.

5 Workload analysis and problem instances

The design of realistic problem instances is a very relevant issue when dealing with the evaluation the new approaches for scheduling and managing HPC infrastructures. We analyzed the workload of Cluster FING in order to gather real information for creating realistic instances of the scheduling problem (including workloads and user behavior when estimating jobs runtime). This section summarizes the main findings about workload analysis and users job runtime estimates and describes the problem instances generated.

5.1 Workload analysis

We analyzed the complete trace of jobs submitted to Cluster FING between April 2010 and March 2015, containing a total of 276803 jobs. As the main results of the statistical analysis of jobs, we found that almost half (49.3%) were small jobs, with less than a minute of execution time, sequential jobs were 44.2%, and parallel jobs were 6.5%. We found a predominance of power of two number of cores requested in parallel jobs (85.1%).

We computed the average accuracy of users runtime estimates by applying the model described in the previous section. Regarding this average, we divided the users in six groups (a bigger group number means a higher accuracy of estimates): *g1*–0 to 0.05, *g2*–0.05 to 0.15, *g3*–0.15 to 0.25, *g4*–0.25 to 0.35, *g5*–0.35 to 0.60, and *g6*–0.60 to 1.0. These groups have 21%, 21%, 18%, 17%, 19%, and 4% of the 117 regular users of the cluster respectively.

5.2 Problem instances

Using the information gathered in the workload analysis, we created problem instances to evaluate the PSP scheduling algorithm under different scenarios.

The simulated infrastructure consists of 37 machines with 12 cores each (a total number of 444 cores). We also configured an execution queue that accepts serial, and parallel jobs requesting up to 16 cores, and up to ten days of execution time. Regarding the task workload generation, we used the software included with the SLURM simulator, and customized its source code to generate specific instances for the problem to study. The changes include adding constraints on the number of cores

requested and maximum requested job runtime, so the jobs generated fulfill the constraints of the execution queue configured.
Each generated workload has 1000 jobs, from 20 users. Each job demands a number of cores that is a power of two between 1 to 16, and up to 10 days of runtime execution. The distribution of the number of cores and runtime execution is representative of the workload at Cluster FING.

In order to test different accuracy of estimates situations, six scenarios simulating different user behavior were generated, the main characteristics of those scenarios are shown on Table 2. Four scenarios are *static* regarding the accuracy on runtime estimates, while the other two emulate users that learn and improve the accuracy of their execution time estimation.

Table 2. Scenarios generated to simulate different user behavior.

scenario	***type***	***accuracy***	***learning schema***
BE	static	group a1 (0.0–0.2)	none
GE	static	group a6 (0.8–1.0)	none
CF	static	groups g1 to g6	none
CFG	static	groups g1 to g6	none
DI	dynamic	incremental improving	all users
HDI	dynamic	incremental improving	half of the users

The first scenario is BE (*Bad Estimates*), in which all users have the worst level of job runtime estimate accuracy (group a1, defined in Section 4), with accuracy between 0.0 and 0.2. In the second scenario, GE (*Good Estimates*), every user has the highest level of accuracy of job runtime estimates (group a6), with an accuracy between 0.8 and 1.0. The third scenario is CF (Cluster FING), where the users accuracy was generated so it is representative of the one accounted at Cluster FING. We divided the accuracy in the six groups, g1 to g6, defined in the previous subsection.

The fourth scenario is CFG (Cluster FING with good estimates), introducing changes in CF scenario to model a situation where almost half the user estimations are in group a6. In order to keep coherence we changed the weight of the groups as follows: a1: 12%, a2: 12%, a3: 9%, a4: 8%, a5: 10%, and a6: 49%.

The last two are dynamic scenarios, which emulate a rational behavior of users that gradually learn how to estimate job execution times. The improvement of estimations is calculated with a frequency of 5 jobs (i.e. every 5 jobs submitted by the user) as follows: if $accuracy \leq 0.5$, then it is increased by $(1-accuracy)\times 0.1$; else *accuracy* is increased by $accuracy\times 0.1$. In the fifth scenario, DI (Dynamic Improvement), all users improve their estimations as described. On the other hand, in the last scenario HDI (Half Dynamic Improvement), only half of the users were modeled to correctly learn how to improvement on their estimates.

The source code of the SLURM synthetic workload generator was modified in order to emulate this six different scenarios of user behavior, and generate them at once, using the exact same workload trace.

6 Experimental analysis

This section reports the experimental analysis of the proposed PSP algorithm over 24 scenarios defined from the combinations of infrastructure, workload, and estimations. All simulations were performed in a virtual machine running Ubuntu v14.04. The results of each simulation were stored in a MySQL database, a functionality provided by SLURM scheduler.

In order to get information about each class of accuracy of users job runtime estimates, we studied the average waiting time per user for the five classes. Table 3 reports, for each of the six scenarios evaluated, the average waiting time (in minutes) of each of the accuracy classes defined. We identify the empty classes with a "–".

Table 3. Average waiting time for each scenario divided by accuracy class.

scenario	*average waiting time (minutes)*				
	a1	***a2***	***a3***	***a4***	***a5***
BE	820.05	–	–	–	–
GE	–	–	–	–	796.18
CF	1093.12	637.53	519.30	450.30	–
CFG	1488.60	986.33	759.15	706.35	517.35
DI	–	1030.25	804.93	534.15	424.43
HDI	1091.58	837.51	781.38	509.70	484.20

We compared the waiting time for users that did not increase their accuracy, and the users that did. First we discuss the results of static scenarios (CF and CFG), then we continue with dynamic scenarios (DI and HDI).

CF is a static scenario in which the highest accuracy class with users is *a4* (0.6 to 0.8). This class has an average waiting time of *450.3 minutes*, which is *2.43, 1.82, and 1.77 times* lower than the average waiting time of class a1, BE, and GE respectively. Only one user was on class *a4*, and his jobs were of the highest priority on the simulation.

Scenario CFG is quite different from CF: there are 6 users on the higher class (*a5*). The average waiting time per user in a5 is *517.35 minutes*, which is *2.88, 1.59,* and *1.54 times* lower than the average waiting time of class a1, BE, and GE respectively. The waiting time in CFG is also higher than the one in CF; the reason is that more users are on class a5, and they compete with each other for computing resources.

DI and HDI are dynamic scenarios, with improvements on the accuracy of users runtime estimates. Due to the dynamism, we decided to include the average waiting time of each user in the class where he belongs *at the end of the simulation*. In DI, the highest accuracy class is *a5*, having an average waiting time of *424.23 minutes*, which is *2.43*, *1.93*, and *1.88 times* lower than the average waiting time of class a2, BE, and GE respectively. The highest accuracy class in scenario HDI is *a5*, in which the average waiting time is *484.2 minutes*, this value is *2.25*, *1.69*, and *1.64* times lower than the average waiting time of class a1, BE, and GE respectively.

Scenario DI has the lower waiting time of all 6 scenarios for its highest populated class, which is *a5*. DI also has an overall average waiting time of 698.44 minutes, lower than the overall average waiting time for HDI (740.87 minutes). In a scenario

where all users improve their estimates, the waiting time improves for all users, even for the ones belonging to the higher class.

Fig. 1 summarizes the average waiting time (in minutes) for jobs submitted for the scenarios CF, CFG, DI, and HDI. The average waiting times are grouped by class, from a1 to a5. For a particular scenario, results show that the waiting time for a job significantly reduces when moving to a higher accuracy class. In a scenario where all users improve their accuracy of runtime estimates, every class experience a drop in their average waiting time, which means a general improvement in the quality of service of the HPC infrastructure. For example, class a2 in DI (the lowest class of the scenario) has lower waiting time than class a1 in all other scenarios in the figure.

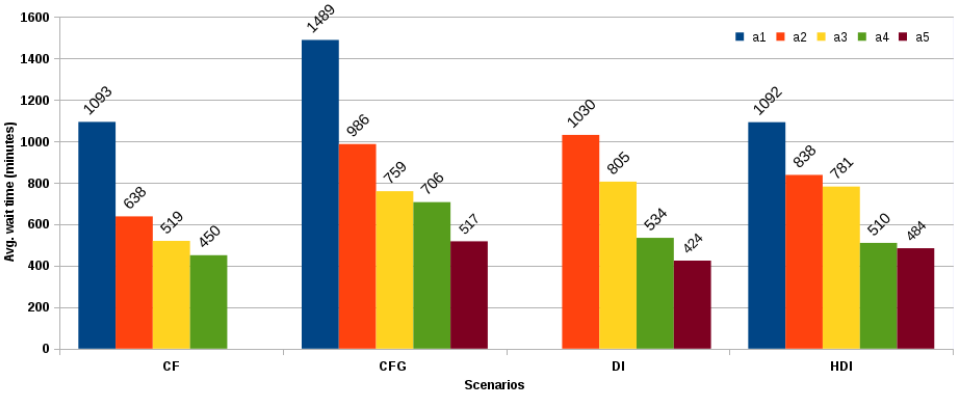


Fig. 1. Average waiting time of each accuracy group, grouped by scenario.

7 Conclusions and future work

This article presented a scheduling policy called *Penalty Scheduling Policy (PSP)*, which focuses on the problem of dealing with low accuracy of job runtime estimates provided by users. The goals of the study are twofold. First, promoting users to accurately estimate the execution time of their jobs. Second, to evaluate the benefits of including a policy for a wise planning of computing resources by prioritizing requests from those users that provide accurate estimate for job execution times.

An experimental evaluation of the use of Penalty Scheduling Policy in a simulated computer system environment, developed using the SLURM simulator, was presented. The empirical study analyzed the PSP performance on six different scenarios regarding the user behavior when estimating job time execution. The main results of the experimental analysis show that in an environment where all users improve their estimates, every users experience a improvement on their quality of service. The proposed strategy was included in SLURM, but it can be easily included in other popular resource management systems such as Maui.

The main lines for future work are related to extend the experimental evaluation of the proposed scheduler using different workloads and statistics. The performance of PSP could be studied over synthetic workloads, created with the generator implemented by Tsafrir [5] using realistic (modal) job runtime estimates. This study

will help to evaluate the real difference between the impact of bad user job runtime estimates, and good ones. We also plan to test the PSP method in a real environment, for example on Cluster FING, in order to test users acceptance of this policy.

8 References

1. Cirne, W., Berman, F.: A comprehensive model of the supercomputer workload. IEEE International Workshop on Workload Characterization, pp. 140–148 (2001).
2. Etsion, Y., Tsafrir, D.: A short survey of commercial cluster batch schedulers. Technical Report 2005-13. School of Computer Science and Engineering, The Hebrew University of Jerusalem (2005).
3. Tang, W., Desai, N., Buettner, D., Lan, Z.: Analyzing and adjusting user runtime estimates to improve job scheduling on the Blue Gene/P. IEEE International Symposium on Parallel & Distributed Processing, pp. 1–11 (2010).
4. Tsafrir, D., Etsion, Y., Feitelson, D.: Modeling user runtime estimates. In: 11th international conference on Job Scheduling Strategies for Parallel Processing, pp. 1–35 (2005).
5. Tsafrir, D.: Using inaccurate estimates accurately. In: 15th international conference on Job Scheduling Strategies for Parallel Processing, pp. 208-221 (2010).
6. Lucero, A.: Simulation of batch scheduling using real production-ready software tools. In: 5th Iberian Grid Infrastructure Conference, pp. 345–356 (2011).
7. Nesmachnow, S. Computación científica de alto desempeño en la Facultad de Ingeniería, Universidad de la República, Revista de la Asociación de Ingenieros del Uruguay 61:12–15, 2010 (text in Spanish).
8. Jackson, D., Snell, Q., Clement, M.: Core algorithms of the Maui scheduler. In: 7th international conference on Job Scheduling Strategies for Parallel Processing, pp. 87–102 (2001).
9. Mu'alem, A. W., Feitelson, D. G.: Utilization, predictability, workloads, and user runtime estimates in scheduling the IBM SP2 with backfilling. IEEE Transactions on Parallel and Distributed Systems 12(6):529–543, 2001.
10. Yoo, A. B., Morris, A. J., Grondona, M.: Slurm: Simple linux utility for resource management. In: 9th international conference on Job Scheduling Strategies for Parallel Processing, pp 44–60 (2003).
11. Zotkin, D., Keleher, P. J.: Job-length estimation and performance in backfilling schedulers. In: 8th International Symposium on High Performance Distributed Computing, pp. 236–243 (1999).
12. Zhang, Y., Franke, H., Moreira, J., Sivasubramaniam, A.: Improving parallel job scheduling by combining gang scheduling and backfilling techniques. In: 14th IEEE International Parallel and Distributed Processing Symposium, pp. 133–142 (2000).
13. England, D., Weissman, J., Sadago-pan, J. : A new metric for robustness with application to job scheduling. In: 14th IEEE International Symposium on High Performance Distributed Computing, pp. 135–143 (2005).
14. Guim, F., Corbaĺan, J., Labarta, J.: Prediction f based models for evaluating backfilling scheduling policies. In: 8th IEEE International Conference on Parallel and Distributed Computing, Applications & Technologies, pp. 9–17 (2007).
15. Iturriaga, S., García, S., Nesmachnow, S.: An Empirical Study of the Robustness of Energy-Aware Schedulers for High Performance Computing Systems under Uncertainty. In High Performance Computing, pp. 143-157 (2014).

Assessing metric structures on GPGPU environments

DOS SANTOS, Eder[1], SOFIA, Albert A. O.[1],URIBE PAREDES, Roberto[2]
[1]Unidad Académica Río Gallegos - Universidad Nacional de la Patagonia Austral
Lisandro La Torre 1070 – 0054-2966-442313 – Río Gallegos – Santa Cruz – Argentina
[2]Departamento de Ingeniería en Computación, Universidad de Magallanes, Chile.
esantos@unpa.edu.ar, sistemasuarg@gmail.com, roberto.uribeparedes@gmail.com

Abstract. Similarity search consists on retrieving objects within a database that are similar or relevant to a particular query. It is a topic of great interest to scientific community because of its many fields of application, such as searching for words and images on the World Wide Web, pattern recognition, detection of plagiarism, multimedia databases, among others. It is modeled through metric spaces, in which objects are represented in a black-box that contains only the distance between objects; calculating the distance function is costly and search systems operate at a high query rate. Metrical structures have been developed to optimize this process; such structures work as indexes and preprocess data to decrease the distance evaluations during the search. Processing large volumes of data makes unfeasible the use of such structures without using parallel processing environments. Technologies based on multi-CPU and GPU architectures are among the most force due to its costs and performance.

Keywords: Similarity search, metric spaces, metric structures parallel processing, GPU.

1 Introduction

The search of similar objects in a large collection of stored objects in a metric database has become a most interesting problem. This kind of search can be found in different applications such as voice and image recognition, data mining, plagiarism and many others.

In general, various data structures are used in order to improve efficiency in terms of distance calculations, compared with the sequential search in the database (known as a brute force algorithm). On the other hand, parallel processing seeks to reduce costs in terms of time of processing large data volumes [1].

There are a number of technologies for parallel processing implementations. Technologies based on multi-CPU architectures, GPU and multi-GPU are among the most current. In this sense, the application of these new technologies on similarity searches in metric spaces [2][3] allow a high level of parallelism at a very low cost.

This paper focuses on similarity search and implementation of metric structures on parallel environments. Section 2 presents the state of the art on issues related to search by similarity metric structures and parallelization technologies. Comparative analysis of experiments are proposed section 3, seeking to identify the behavior of a set of

metric spaces and metric structures on processing platforms based on multicore and GPU. Finally, conclusions and suggestions for future works are in section 4.

1.1. Similarity Search in Metric Spaces

Similarity is modeled in many interesting cases through metric spaces, and the search of similar objects through range search or nearest neighbors. a metric space is a set X and a distance function$d:X2 \rightarrow R$, so that $\forall x, y, z \in X$, fulfills the properties of positiveness($d(x,y)>=0$ *and* $d(x,y)=0$ *iff* $x=y$), simetry$(d(x,y)=d(y,x))$and triangle inequality$(d(x,y)+d(y,z)>=d(x,z))$.

In a given metric space (X, d) and a finite data set $Y \subset X$, a series of queries can be made. The basic query is the **range query**, a query being an object $x \in X$ and a range $r \in R$. The query range around x with range r is the set of objects $y \in Y$ so that $d(x,y)<= r$. A second type of query that can be built using the range query is **k nearest neighbors**, a query being an object $x \in X$ and an object k. The k nearest neighbors to x are a subset A of objects Y, such that if $|A|=k$ and there is no object $y \in A$ such that $d(y,x)$ is lower than the distance of some object of A to x.

The aim of search algorithms is to minimize the number of distance evaluations performed to resolve a query. The methods for searching in metric spaces are mainly based on dividing the space using the distance to one or more selected objects. Not working with the particular characteristics of each application has the advantage of being more general, because the algorithms work with any type of object [4].

1.2. Metric data structures

There are different structures for searching metric spaces, which can use discrete or continuous distance functions. Metric space data structures can be grouped into two classes, clustering-based and pivots-based methods.

The clustering-based structures divide the space into areas, where each area has a so-called center. Some data is stored in each area, which allows easy discarding the whole area by just comparing the query with its center. Algorithms based on clustering are better suited for high-dimensional metric spaces, which is the most difficult problem in practice. Some clustering-based indexes are GNAT [5], M-Tree [6], SAT [7], Slim-Tree [8], EGNAT [9] and others [4].

In the pivots-based methods, a set of pivots are selected and the distances between the pivots and database elements are pre-calculated. When a query is made, the query distance to each pivot is calculated and the triangle inequality is used to discard the candidates. Its objective is to filter objects during a request through the use of a triangular inequality without really measuring the distance between the object under request and the discarded objects. Some pivots-based indexes are: LAESA [10], FQT and its variants [11], Spaghettis and its variants [12][13], FQA [14], SSS-Index [15] and many others [16].

A Generic Metric Structure (GMS) is an array-type structure based on pivots, which are obtained randomly. It contains the set of distances between the pivots and

the database objects. Each cell stores the distance d(yi, pj) being a database object yi. The algorithm for range search on this structure, given a query q and a range r is:

Figure 1 represents a GMS built with 4 pivots. For each query q being the distances to pivots d(q,pi) = {8, 7, 4, 6} and the search range r = 2, intervals {(6, 10), (5, 9), (2, 6), (4, 8)} are defined. Cells marked in dark gray are within the search range. The strikethrough cells are the candidate objects (2, 13 and 15), which will be evaluated directly with the query.

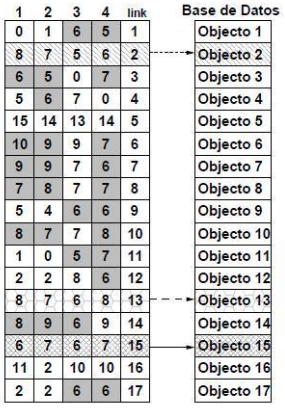


Figure 1. Search over a generic metric structure.

1.3. Parallel Processing Platforms

Currently, there is a wide variety of parallel platforms on which metric structures can be implemented. In this context, many studies have initially focused on distributed memory platforms using high level libraries such as MPI [MPI94] or PVM [PVM94] and shared memory using OpenMP [OMP07] [GKKG03] directives. In this sense, [GBMB10] proposes a strategy for organizing query processing on metric spaces in multi-core nodes. Finally, recent work has dealt newer shared memory technologies, including GPU-based platforms [16] [17] [18].

OpenMP is an application programming interface (API) for shared memory multiprocessing programming on multiple platforms. It is a specification of a set of compiler directives, library routines and environment variables that can be used for high-level parallelism in programs written in C, C ++ and Fortran, based on the fork-join execution model. It is available in many architectures, including Unix and Microsoft Windows platforms, and distributes processing to the microprocessor cores.

On the other hand, graphical processing units (GPU) have a high number of cores with high bandwidth. These devices can increase processing capacity with respect to CPU [19]. A trend called General Purpose Computing on GPU or GPGPU has guided the use of GPU on new types of applications. In this sense, the manufacturers of GPU devices have proposed new languages or extensions for high level languages. An example is CUDA, which is a software platform that allows use GPU devices for general purpose applications, with the ability to handle a large number of threads.

Finally, an application built with a hybrid parallel programming model can run on a cluster of computers using OpenMP and other technologies such as GPU or MPI, or through OpenMP extensions for distributed memory systems.

1.4. Parallelism on Metric Spaces

Metric structures have some unusual features that make it difficult to its direct implementation in real applications. The first is related to dynamic capabilities: most of these structures must be rebuilt if there are new items to index or if it is needed to remove objects from the database. Another feature, even rarer, is related with structures that enable an efficient data handling in secondary memory, which should be considered additional cost parameters such as the number of disk accesses and the index size, among others. Finally, metric structures generally do not consider the memory hierarchy; therefore, it is important to consider this aspect in order to achieve greater performance and efficiency with the ability to use technologies such as GPU, which have different memory levels in its system.

It is impractical to use metric structures in real applications if it does not consider its implementation in parallel environments. To parallelize metric structures it must be considered, among other criteria: the proper distribution of the database on the environment, eg. in a cluster of PCs; the parallelization of search methods; the efficiency in communication between processors; etc.

According to available data and previous work, it has been raised to implement different solutions on disparate spaces, such as color histograms, databases of words, coordinate vectors, Gauss vectors, NASA images and other data, in order to identify the optimal distribution of data in such spaces, both the database and the structures, under GPU and / or Multi-GPU environments, on memory managed by the CPUs and GPU dedicated memories. Finally, it aims is to assess the scalability of implementations according to the database size, and consequently the storage and processing of the Databases, and running queries on secondary memory. The following shows case studies, results and conclusions.

2. Analysis, Discussion and Results

Parallelizing metric structures on GPU and multicore clusters are little explored research areas. In this context, two research groups are developing works around metric structures on GPU-based platforms. The first group of the Complutense University of Madrid has focused his studies over two metrics tructures, List of Clusters and SSS-Index, and has submitted various proposals for kNN and queries by range**¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**. In the second group, of the University of Castilla La Mancha, Professor Roberto Uribe-Paredes is involved. The lines of research addressed by this group are aimed at developing and strengthening the generic structure presented above on a hybrid environment [16] [17] [20]. Finally, metric structures on a GPU are used in [2], and their results are compared with sequential versions; a metric structure is used and the results are compared with sequential and

multicore-CPU versions, showing a noticeable improvement when using this new platform.

Based on the experiences of the research group members, it has proposed to confirm or refute the convenience of using generic metric structures based on pivots in GPU environments. To do this, initially it is proposed:

- On one hand, it aims to evaluate the search performance on GMS structures compared to brute force search.
- Furthermore, the GPU performance is compared with other processing options. It seeks to determine spaces and dimensions in which is really convenient to use metric structures on GPU-based platforms.

The analysis has carried out an exhaustive set of laboratory tests on a wide range of parallel environments and metric spaces.

2.1. Study Cases

Different representative metric spaces have been selected in the experiments conducted by the research group. In order to classify those spaces, the distance function for similarity search is used. Thus, it is possible to identify two groups of metric spaces.

The first set of areas corresponds to different language dictionaries. The distance function used is the edit distance, which corresponds to the minimum number of insertions, deletions or substitutions needed for a word equal to another. Similarity ranges applied in the search vary from 1 to 4.

The second group of metric spaces contains databases vectors of different sizes, which represent different objects; they are predominantly colored histograms of different images (photos of faces, satellite images, diagrams), wherein such histograms are represented as vectors. For this second group, the Euclidean distance has been chosen as the distance, and ranges recovered 0.01%, 0.1% and 1% of similarity from the dataset used.

To perform the experiments, both databases are divided into two random sets; The first one contains 90% of the database objects, and it is used to build the metric structures; the remaining 10% is used as a set of query objects.

As a platform for experimental evaluation, the mentioned metric spaces have been used under multicore and GPU environments using the GMS structure based on different numbers of pivots and using also the brute force search algorithm. For each routine, the application runs 4 times and an average is obtained. The experiments generated for each metric space consist of a Cartesian product of the variables described below:

- Index / Structures / Algorithm: Brute Force and pivots-based GMS.
- Number of Pivots (for GMS): 1, 2, 4, 8, 16, 32, 64, 128, 256, 512 y 1024.
- Processors: 1 CPU (secuential), 4 CPU (multi-core), 8 CPU (multi-core with hyper-threading) and GPU (using GPU global and shared memory).

To quantify the experiments, a summary of the variables considered and experiments realized is shown in Table 3.

Database Types	Spaces	Parallelism Methods	Algorithms	Search Ranges	Repetitions	Total
Gauss Diagrams	5	5	12	3	4	3600
Other Vectors	3	5	12	3	4	2160
Dictionaries	10	5	12	4	4	9600
TOTAL						15360

Table 1. Summary of experiments realized.

Finally, the hardware used corresponds to an Intel® Core™ i7-2600 CPU @ 3.40GHz 4-core and Hyper-Threading support, 12GB of main memory and two EVGA Nvidia cards DDR5 384 CUDA cores and 1 GB of global memory each one. The codification of the metric structures and search algorithms has been performed using the C language (gcc 4.3.4), executed on Ubuntu Linux 12.04 LTS (Precise Pangolin); CUDA SDK v3.2 has been adopted for parallelizing GPU applications, and OpenMP library has been used for multi-CPU parallelization.

2.2. Results

Next, results of experimental evaluation are detailed. It should be noted that the experiments were performed so that it is possible make the following comparisons:

1. Efficiency of different processing options: experiments with codes designed for different processing options:
    a. Sequential: using 1 core in its maximum capacity.
    b. Multi-core: using a 4 cores processor at its maximum capacity with OpenMP.
    c. Hyper-Threading(TM): using a 4 cores processor with Hyper-Threading at its maximum capacity with OpenMP.
    d. GPU using its global memory.
    e. GPU using its shared memory.
2. GMS vs. Brute Force algorithm: first. Experiments using the brute force algorithm and GMS with different numbers of pivots (1, 2, 4, 8, 16, 32, 64, 128, 256, 512and1024) where performed in order to identify the most efficient number of pivots in each experiment.

Tests were performed over the different databases specified in the previous paragraph. Here is a selection of relevant results that guide the analysis.

2.2.1. Sequential Processing vs. Parallel Processing

This first block analysis includes the various processing options (sequential, multicore, GPU using its global memory, GPU using its shared memory) on the Brute

Force algorithm; This allows us to visualize the performance level of processing the entire database without using indexes or metric structures. A metric space of vectors (histograms with absolute size 112) and words in Spanish dictionary were used as case studies, which where adopted initially to show in a general way the performance of applications over the different distance functions described herein selection.

In order to quantify the results, processing times (in seconds) are used as a measurement unit. Then, the ratio is calculated between two results A and B, giving rise to the latency speed-up. In this first analysis, we can observe a significant difference between sequential processing and parallel processing. Upcoming it shows the processing times achieved and the corresponding speedup values.

Option	Time (s.)	SpeedUp			
		GPU	GPU SM	MC	SEC
GPU	647,30	--	*0,09*	*0,86*	5,25
GPU SM	56,48	11,46	----	9,83	60,20
MC	555,09	1,17	*0,10*	--	6,13
SEC	3400,15	*0,19*	*0,02*	*0,16*	--

Table 2. Speedup values on a search of rank 1 in a metric space of words.

The higher difference is observed with the use of shared memory (SM GPU) compared to sequential processing, which results in a speedup of 60.2, 56.2, 54.21 and 52.89 for ranks 1, 2, 3 and 4 respectively.It should be noted that those results are not enough to corroborate the efficiency of GPU processing, so more experiments were performed over different metric spaces. The results obtained in experiments on a vectors metric space are shown below.

Option	Time	SpeedUp (Range 0.01)			
		GPU	GPUSM	MC	SEC
GPU	144,16	--	*0,97*	*0,99*	3,76
GPU SM	139,84	1,03	--	1,02	3,88
MC	143,22	1,01	*0,98*	--	3,79
SEC	542,26	*0,27*	*0,26*	*0,26*	--
Option	**Time**	**SpeedUp (Range 0.1)**			
		GPU	GPUSM	MC	SEC
GPU	144,17	--	*0,97*	*0,99*	3,74
GPU SM	139,77	1,3		1,02	3,88
MC	143,16	1,01	*0,98*	--	3,76
SEC	538,92	*0,27*	*0,26*	*0,27*	--
Option	**Time**	**SpeedUp (Range 1)**			
		GPU	GPUSM	MC	SEC
GPU	144,26	--	*0,97*	*0,99*	3,72
GPU SM	139,82	1,03	--	1,03	3,83
MC	143,39	1,01	*0,98*	--	3,74
SEC	535,99	*0,27*	*0,26*	*0,27*	--

Table 3. Speedup obtained in the search for a metric space of vectors.

As shown, the speedup obtained in shared memory processing compared to sequential processing were 3.88, 3.86 and 3.83 for the search ranges 0.01%, 0.1% and 1%, respectively. In this sense, this second experiment shows more equity in processing times compared to the metric space previously used in regard to the performance of the various options for parallelism. In the former case, it is possible to

see speedup values near 11.4 (GPU shared memory vs. GPU global memory) and 9.8 (GPU shared memory vs. multicore with Hyper-Threading). In the last example, the speedup values obtained were respectively of 1.03 / 1.02, 1.03 / 1.02 and 1.03 / 1.03.

2.2.2. Metric structures vs. Brute force under GPU

The above analysis shows that parallel processing using the shared memory of the GPU has a substantially superior yield to the other techniques used under Brute Force algorithm. However, the analysis with metric structures must be made considering the costs of processing and storage of such structures in the hierarchy of memory devices, so it is necessary to make different assessments as to the size of the database metric and respective data structures, and their behavior in different amounts as to pivot and thus to structures of different sizes.

To assess the performance of metric structures under different techniques of parallelism, some experiments have been performed using structures with different numbers of pivots (1, 2, 4, 8, 16, 32, 64, 128, 256, 512 and 1024 pivots) on metric spaces with different properties. The results of experiments show that the search times on the GMS have been predominantly better than brute force processing different metric spaces using the shared memory of the GPU.

2.2.3. Impact of search range and the number of pivots on the performance of Metric Structures

The experiments reveal a variation in processing times according to the search range applied in routines that use the GMS structures, contrary to the brute force algorithm (in which processing times are uniforms for different search ranges). In this sense, it has also been possible to confirm that the processing time on metric structures tends to increase as the search range increases, as it tends to make more distance evaluations extent that the search range is increased.

In addition, experiments were performed using the GPU's shared memory on metric structures with 1, 2, 4, 8, 16, 32, 64, 128, 256, 512 and 1024 pivots, in order to visualize the behavior of the structures with different numbers of pivots. Two trends were identified: on the one hand, the decrease in performance as the search range is increased in structures with high amounts of pivots. Furthermore, the performance of structures with few pivots is better as that the search range increases.

2.2.4. Parallelism on GMS Structures

The last analysis proposed in this paper aims to evaluate the most efficient method of parallelism on generic metric structures in different metric spaces, in order to verify the convenience of using GPU metric structures. To do this, experiments on different metric spaces under various forms of parallel processing described in this paper (OpenMP, GPU GPU global memory and shared memory) and different amounts of pivots were made.

The results revealed two trends in the behavior of structures: first, search times with the generic metric structure increases as long as the search ranges increases under all parallel environments and any number of pivots. On the other hand, can be observed that the performance of shared memory processing with GPU is convenient as the search range increases (consequently the number of assessments carried away).

3. Conclusions and Future Work

The need to process large volumes of data requires to increase processing capacity and to reduce search times. In this context, it is relevant to study parallelization algorithms and the distribution of the databases. On the other hand, the increased size of the databases and the emergence of new types of data on which there's no interest in performing searches with exact matches create the need to establish new structures for similarity search.

This paper has presented the main features of the so-called metric spaces and the generic metric structure used to perform similarity searches in such spaces. It has deepened the discussion on those experiments based on the use of generic structures based on pivots, compared with linear or sequential processing known as brute force. It also has shows results of experiments using different strategies for parallel processing, analyzing the search performance in environments with GPU (CUDA) and multi-core (OpenMP).

As future work, it is suggested to discuss the scalability of the use of those platforms, as currently proposed experiments on individual computers. Hybrid aims to reach solutions to the distribution of metric spaces and their structures, so that it is possible to make applications on data volumes in production scale. Finally, another line of great interest to be addressed is related to the energy efficiency of various devices and technologies for parallel processing of large volumes of data with regard to the similarity search of metric spaces.

References

1. GRAMA, Ananth and KARYPIS, George and KUMAR, Vipin and GUPTA, Anshul. Introduction to Parallel Computing (2nd Edition). Addison Wesley. 2003.
2. Roberto URIBE-PAREDES, Pedro VALERO-LARA, Enrique ARIAS, José L. SÁNCHEZ and Diego CAZORLA. Similarity search implementations for multi-core and many-core processors. In: 2011 International Conference on High Performance Computing and Simulation (HPCS), pp. 656–663 (July 2011). Istanbul, Turkey.
3. E. ARIAS, D. CAZORLA, J. L. SÁNCHEZ, R. URIBE-PAREDES. "Una estructura Métrica Genérica para Búsquedas por Rango sobre una Plataforma Multi-GPU". XVII Jornadas de Ingeniería del Software y Bases de Datos (JISBD2012). Sept. 2012, Almería, España.
4. Edgar CHÁVEZ, Gonzalo NAVARRO, Ricardo BAEZA-YATES, and José Luis MARROQUÍN. Searching in metric spaces. ACM Computing Surveys, 33(3):273--321, 2001.

5. Sergei Brin. Near neighbor search in large metric spaces. In the 21st VLDB Conference, pages 574—584. Morgan Kaufmann Publishers, 1995.
6. Paolo Ciaccia, Marco Patella, and Pavel Zezula. M-Tree : An efficient access method for similarity search in metric spaces. In the 23st International Conference on VLDB, pages 426--435, 1997.
7. Gonzalo Navarro. Searching in metric spaces by spatial approximation. The Very Large Databases Journal (VLDBJ), 11(1):28--46, 2002.
8. Caetano Traina, Agma Traina, Bernhard Seeger, and Christos Faloutsos. Slim-trees: High performance metric trees minimizing overlap between nodes. In VII International Conference on Extending Database Technology, pages 51--61, 2000.
9. Roberto Uribe-Paredes and Gonzalo Navarro. Egnat: A fully dynamic metric access method for secondary memory. In Proc. 2nd International Workshop on Similarity Search and Applications (SISAP), pages 57--64, Prague, Czech Republic, August 2009. IEEE CS Press.
10. María Luisa MICÓ, José ONCINA, and Enrique VIDAL. A new version of the nearest-neighbour approximating and eliminating search algorithm (AESA) with linear preprocessing time and memory requirements. Pattern Recognition Letters, 15(1):9--17, January 1994.
11. R. BAEZA-YATES, W. CUNTO, U. MANBER, and S. Wu. “Proximity matching using fixed queries trees”. In 5th Combinatorial Pattern Matching (CPM’94), 1994, LNCS 807, pp. 198–212.
12. Edgar CHÁVEZ, José L. MARROQUÍN, and Ricardo BAEZA-YATES. Spaghettis: An array based algorithm for similarity queries in metric spaces. In 6th International Symposium on String Processing and Information Retrieval (SPIRE'99), pages 38--46. IEEE CS Press, 1999.
13. S. Nene and S. Nayar. A simple algorithm for nearest neighbor search in high dimensions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(9):989--1003, 1997.
14. E. CHÁVEZ, J. MARROQUÍN, and G. NAVARRO. “Fixed queries array: A fast and economical data structure for proximity searching”. Multimedia Tools and Applications, vol. 14, no. 2, pp. 113–135, 2001.
15. Oscar Pedreira and Nieves R. Brisaboa. “Spatial selection of sparse pivots for similarity search in metric spaces”. In 33rd Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM 2007), Harrachov, Czech Republic, 2007, vol. 4362 of LNCS, pp. 434–445, Springer.
16. Roberto Uribe-Paredes, Diego Cazorla, José L. Sánchez, and Enrique Arias. “A comparative study of different metric structures: Thinking on gpu implementations”. In International Conference of Computational Statistics and Data Engineering (ICCSDE’12), London, England, July 2012.
17. Roberto Uribe-Paredes, Enrique Arias, José L. Sánchez, and Diego Cazorla. “Improving the Performance for the Range Search on Metric Spaces using a Multi-GPU Platform”. To appear: 23rd International Conference on Database and Expert Systems Applications (DEXA 2012). Vienna, Austria, Sept. 2012.
18. Vincent Garcia, Eric Debreuve, and Michel Barlaud. Fast k nearest neighbor search using GPU. Computer Vision and Pattern Recognition Workshop, 0:1--6, 2008.
19. Wu-Feng and Dinesh Manocha. High-performance computing using accelerators. Parallel Computing, 33:645--647, 2007.
20. O. SOFIA, J. SALVADOR, E. DOS SANTOS, R. URIBE PAREDES. Búsquedas por Rango sobre Plataformas GPU en Espacios Métricos. pp 658 - 662. XV Workshop de Investigadores en Ciencias de la Computación - WICC 2013. Proceedings published in CD format by Universidad Autónoma de Entre Ríos. RedUNCI. ISBN13: 978-987-28179-6-1.

Conversión de RGB a YCbCr en System Generator y HLS

Romina Molina and Veronica Gil-Costa

Universidad Nacional de San Luis
San Luis, Argentina
{rmolina,gvcosta}@unsl.edu.ar

Abstract. Los sistema de recuperación de imágenes basados en contenidos (CBIR) típicamente ejecutan dos tareas de alto costo computacional: (1) generación de la base de datos e indexación y (2) recuperación. Este trabajo se enfoca en la primera tarea la cual es inherentemente paralela, debido a que las imágenes son generalmente divididas en varias partes y cada parte se procesa por separado y de manera similar. Para ello se presenta un conversor de espacio de color de RGB a YCbCr para plataformas de FPGAs basadas en SoCs. Esta conversión de color se utiliza como parte del proceso de extracción del descriptor de la distribución de color, el cual forma parte del estándar MPEG-7 y es utilizado en sistemas CBIR. Se presenta la implementación del conversor haciendo uso de System Generator y de Vivado HLS y se efectúa una comparación de los resultados sobre la plataforma ZYNQ - ZC7020 Evaluation Kit.

Keywords: MPEG-7, system generator, descriptores, HDL, multimedia.

1 Introducción

La recuperación de imágenes basada en contenido (CBIR) es un área cuyo objetivo es desarrollar herramientas para recuperar información visual y tiene aplicaciones importantes en muchas áreas incluyendo medicina, educación, diseño de arquitecturas, agricultura. Para representar una imagen, se utilizan los contenidos visuales como el color, la forma, la textura, etc..

El estándar de representación MPEG-7 (también conocido como Interfaz de Descripción de Contenido Multimedial) ha sido amplianente utilizado en los últimos años. Surge debido la necesidad de estandarizar la repesentación del contenido de la información multimedia. Trabajos como [1] y [2] exponen mejoras de MPEG-7 respecto a MPEG-4. Es un estándar ISO/IEC desarrollado por el grupo MPEG [3],[4],[5]. Dentro de sus especificaciones se definen la semántica y la sintáxis para lograr una adecuada descripción del contenido.

MPEG-7 se encuentra estructurado en 7 partes: sistemas, lenguaje de descripción y definición, visual, audio, entidades genéricas y esquemas de descripción multimedia(MDS), software de referencia y conformance testing. En

la parte visual de MPEG-7, los descriptores se clasifican en: descriptores normativos (describen colores, formas, texturas y movimiento de datos visuales), descriptores básicos y descriptores para localización. Dentro de los campos de aplicación de este estándar se encuentran: bibliotecas digitales, aplicaciones biomédicas [6], búsqueda de personas, aplicaciones educativas, control de tráfico, sistemas de vigilancia [7], edición multimedia, entre otras.

La contrucción de los descriptores MPEG-7 que representan las imágenes, es un proceso computacionalmente intensivo lo cual lo hace un buen candidato para ser paralelizado. Por lo tanto, en este trabajo se propone utilizar plataformas paralelas que permitan acelerar la construcción de estos descriptores. En particular, se propone utilizar las FPGAs basadas en SoC (System on Chip) que han mostrado ser una solución eficiente debido a su paralelismo inherente. Las FPGAs son de alta eficiencia energética y proporcionan un alto poder de cómputo debido a la posibilidad de adaptar los diseños basados en FPGA a una arquitectura específica. Estas características las hacen ideales para las operaciones de procesamiento de imagen o visión por computador. Los SoCs hacen referencia a las nuevas tecnologías que integran un microcontrolador, procesador, DSPs, módulos de memoria, osciladores, contadores, temporizadores, interfaces externas, AD/DA, entre otros componentes. Debido a la complejidad de los chips, esta tecnología puede ser programada, no solo con VHDL o Verilog, sino con HDL de más alto nivel como SystemVerilog, SystemC, C/C++. Trabajos como [8], [9] reflejan el uso de los SOC en el procesamiento de imágenes.

Recientemente se ha incrementado el uso de System Generator para realizar diseños hardware para FPGA. Áreas como control [12], electrónica de potencia [13], procesamiento de señales [14],[15], entre otras. En lo que respecta al procesamiento de imágenes, ha sido utilizado para implementar las operaciones básicas como binarización, conversión a escala de grises y mejora del contraste [16], [17], [18].

Este paper se enfoca específicamente en el desarrollo del bloque correspondiente a la etapa del cambio de espacio de color de RGB a YCbCr, implementado mediante System Generator y Vivado HLS, y se compara la utilización de los recursos que reporta cada una de esas herramientas. Este trabajo se organiza de la siguiente manera. La Sección 2 describe el proceso de construcción de descriptores para imágenes utilizando el Color Layout Descriptor de MPEG-7. En la Sección 3 de detalla la implementación de la etapa que involucra la conversión del espacio de color RGB a YCbCr utilizando System Generator y Vivado HLS. La Sección 4 presenta compara el número de recursos requeridos por System Generator y Vivado HLS. Finalmente, se concluye en la Sección 5.

2 Generación de descriptores de color en MPEG-7

Un descriptor es una representación de una característica definida sintáctica y semánticamente. Una imagen puede quedar caracterizado por más de un descriptor. Uno de los descriptores más utilizados es el Descriptor de la distribución de color (CLD - Color Layout Descriptor), el cual permite obtener la distribución

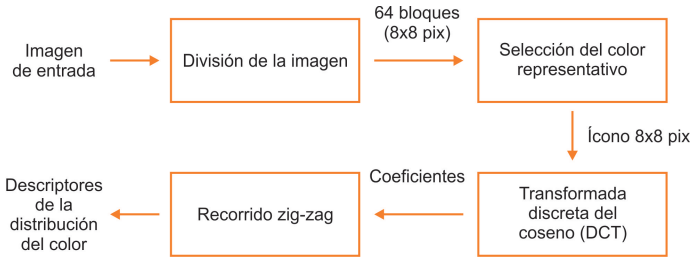


Fig. 1. Proceso de extracción de la distribución de color.

espacial del color de una imagen y es uno de los descriptores más rápidos y precisos del estándar MPEG-7.

El CLD está compuesto por 3 vectores, cada uno correspondiente a los canales Y (luminancia), Cb (crominancia azul) y Cr (crominancia roja) de la imagen. Los pasos para la obtención del mismo son los siguientes:

- Generación de imagen de 8x8 bloques: La imagen a procesar es dividida en 64 bloques de igual tamaño, para obtener una matriz de 8x8 bloques.
- Detección del color más representativo: De cada bloque se calcula el promedio con los píxeles pertenecientes a éste. Con estos valores se arma una nueva imagen de 8x8 píxeles.
- Cambio de espacio de color a YCbCr: Se realiza una transformación del espacio de color de RGB a YCbCr.
- Transformada Discreta del Coseno (DCT): La luminancia Y, la crominancia roja Cr y la crominancia azul Cb son transformadas en una matriz de 8x8 mediante la transformada Discreta del Coseno (DCT).
- Cuantización de coeficientes: Se realiza un reordenamiento y cuantización de los coeficientes de las matrices resultantes del paso anterior.

Una de las etapas de la extracción del descriptor de la distribución del color es la transformación de color de RGB a YCbCr. Este último espacio de color mencionado es el recomendado por el estándar MPEG-7 y hace referencia a la luminancia y crominancia (diferencia de azul y rojo respectivamente) de una imagen. Trabajos como [10], [11] efectúan una implementación de esta conversión de espacios en VHDL.

La transformación de RGB a YCbCr está dada por las siguientes relaciones:

$$Y = 0.299 * R + 0.587 * G + 0.114 * B \tag{1}$$

$$Cb = -0.169 * R – 0.331 * G + 0.500 * B \tag{2}$$

$$Cr = 0.500 * R – 0.419 * G – 0.081 * B \tag{3}$$

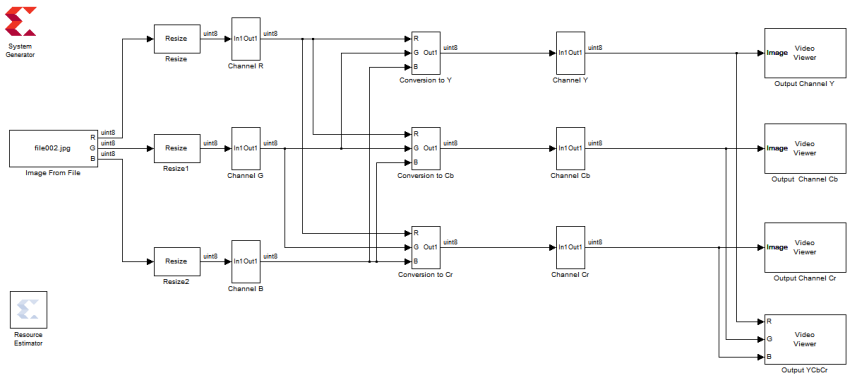


Fig. 2. Esquema en Simulink del bloque de conversión de espacio de color RGB a YCbCr.

3 Uso de plataformas paralelas para la conversión del espacio de colores

En esta sección se presentan los diseño paralelos de la etapa de conversión del espacio RGB a YCbCr utilizando System Generator y Vivado HLS. Para disminuir los recursos de hardware en la implementación, las ecuaciones presentadas en la seccion anterior se redefinen en punto fijo de la siguiente manera:

$$Y = 77 * R + 150 * G + 29 * B \tag{4}$$

$$Cb = -43 * R–85 * G + 128 * B + 32767 \tag{5}$$

$$Cr = 128 * R–107 * G–21 * B + 32767 \tag{6}$$

3.1 Implementación en System Generator

Es esquema general desarrollado en Simulink puede verse en la Fig. 2. Está compuesto por un bloque encargado de la lectura de la imagen, de una etapa de pre-procesamiento, de las etapas encargadas de la conversión, una etapa de recuperación de la imagen y, finalmente de la muestra en pantalla de los resultados obtenidos (tanto en canales separados como unidos).

El pre-procesamiento de la imagen en Simulink se efectúa mediante el esquema de la Fig. 3. Esta etapa está compuesta por los siguientes bloques: (a)Bloque Convert 2-D to 1-D: Convierte la imagen en un arreglo de 1D; y (b)Bloques Frame conversion and unbuffer: Especifica que el modo de muestreo de la señal de salida del bloque sea *frame-based*[1] y convierte la matriz de entrada en un arreglo de escalares.

La imagen de entrada es cargada al sistema mediante el bloque Image from file. La misma tiene un tamaño de 8x8 píxeles y se encuentra en RGB. Este

[1] http://www.mathworks.com/help/dsp/ref/frameconversion.html

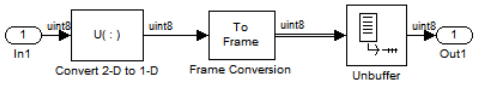


Fig. 3. Bloques que efectúan el pre-procesamiento de la imagen.

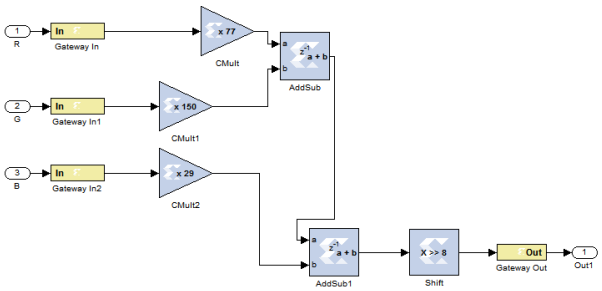


Fig. 4. Esquema en Simulink de la generación del canal Y.

tamaño de la imagen se debe a que en esta etapa de la extracción del descriptor, la misma fue reducida en los pasos previos a la conversión de espacio de color. El funcionamiento del sistema presentado en este trabajo, también fue verificado para una imagen de entrada de 512x512 píxeles en RGB.

Para obtener el valor de luminancia Y, la Fig. 4 muestra como la ecuación (4) es traducida a los bloques de System Generator. Se hizo uso de:

- Gateway In: Este bloque es la entrada a la parte del diseño a ser implementada en System Generator del diseño total. Para este caso, se emplean 3 Gateway In, los cuales se corresponden a los 3 canales de la imagen (R, G, B).
- Cmult: Efectúa las multiplicaciones entre la señal de entrada y el factor correspondiente. En la configuración de este bloque se carga el valor del multiplicador en punto fijo. Este bloque implementa un operador de ganancia, donde la salida es el resultado de la multiplicación entre el valor constante y la señal de entrada.
- AddSub: Realiza la suma o resta entre 2 señales de entrada.
- Shift: Realiza el corrimiento de 8 bits hacia la derecha.
- Gateway Out: Este bloque es la salida de la parte del diseño en system Generator hacia el resto del sistema realizado en Simulink. Se utiliza un solo bloque debido a que solamente se genera la salida correspondiente al canal Y. Como se puede observar que, para la generación de los valores para la luminancia, están implicados los 3 canales de la señal de entrada.

La misma estructura utilizada para la generación del canal de luminancia se emplea para generar los canales de crominancia, con la diferencia que se le

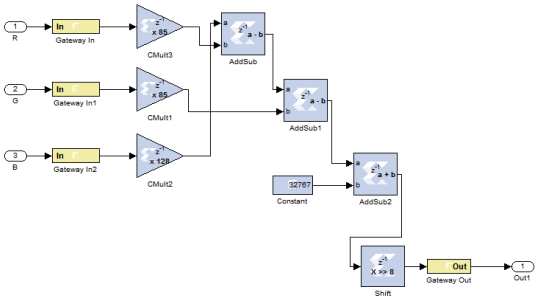


Fig. 5. Esquema en Simulink de la generación del canal Cb.

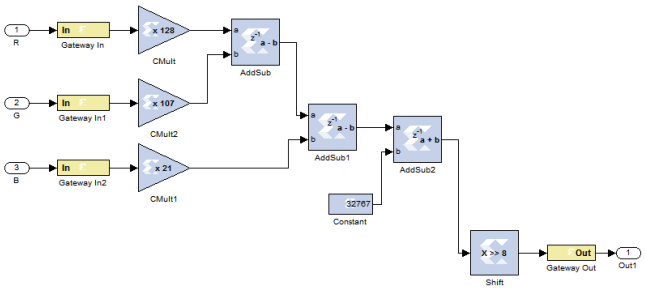


Fig. 6. Esquema en Simulink de la generación del canal Cr.

Fig. 7. Bloques que efectúan la recuperación de la imagen.

incorpora un bloque Constant, debido a que, como se puede observar en las ecuaciones (5) y (6) se suma un valor de 32767.

Los bloques de Simulink encargados de efectuar la recuperación de la imagen a través del arreglo obtenido del procesamiento realizado por los bloques de System Generator son los siguientes:

- Bloque Data type conversion: convierte al formato entero sin signo.
- Bloque Buffer : Convierte las muestras escalares para enmarcar la salida a menor velocidad de muestreo.
- Bloque Convert 1D to 2D: Convierte el arreglo de 1D a la matriz de la imagen correspondiente.

La Fig. 7 muestra la conexión de los bloques mencionados anteriormente. La visualiación de las imágenes obtenidas para cada canal, así como la imagen

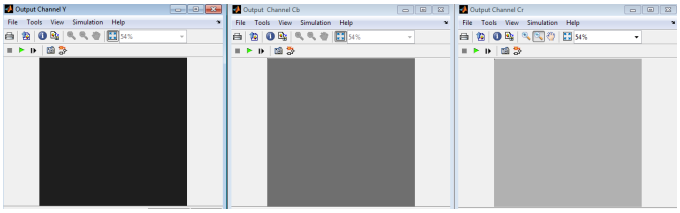


Fig. 8. Salidas: Canal Y, canal Cb y canal Cr respectivamente.

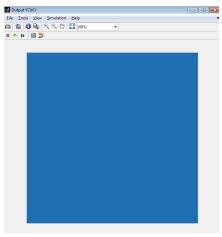


Fig. 9. Salidas: Canal Y, canal Cb y canal Cr respectivamente.

compuesta por los 3 canales se efectúa mediante el bloque Video Viewer. El resultado de la conversión de espacio de color obtenido mediante System Generator se puede observar en la Fig. 8. En la Fig. 9 se presenta la imagen resultante de combinar los 3 canales (Y, Cb y Cr).

3.2 Implementación en Vivado HLS

La implementación en Vivado HLS de la conversión de color se efectúa mediante la codificación en lenguaje C de las ecuaciones en punto fijo (4), (5), (6) mencionadas anteriormente.

La función que efectúa la transformación está compuesta por un bucle, el cual va leyendo un arreglo y efectuando las operaciones necesarias para la conversión. Al estar en punto fijo antes de salir del bucle, se efectúa un shift hacia la derecha sobre el resultado de cada operación, para obtener el resultado definitivo.

Como HLS permite definir directivas para mejorar la performance del diseño, se utilizó la directiva de pipeline para el bucle. En lo que respecta a las directivas relacionados a los recursos a utilizar en la FPGA, se optó por hacer uso de aquellas que permiten seleccionar que las operaciones aritméticas se efectúen en los DSPs disponibles. Sin estas últimas directivas, el hardware generado no hace uso de los DSP e implementa las funciones con lógica combinacional. En la Fig. 10 se muestra el flujo de la función que implementa la conversión del descriptor.

4 Resultados

La plataforma elegida para la implementación fue ZYNQ-7 ZC702 Evaluation Board (xc7z020clg484-1). La misma dispone de 280 BRAM 18K, 220 DSP,

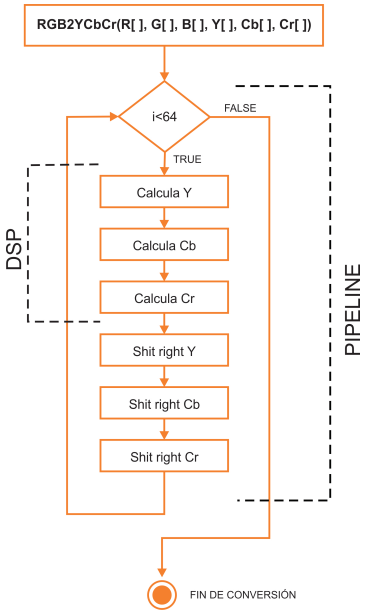


Fig. 10. Salidas: Canal Y, canal Cb y canal Cr respectivamente.

106400 DSP y 53200 LUT. En la tabla 1 se muestra la comparación de recursos entre ambos diseños. La columna Sys Gen muestra los recursos utilizados por System Generator. HLS_nd y HLS presentan los recursos requeridos al realizar la implementación de la conversión del espacio de colores mediante Vivado HLS. El primer caso corresponde a los recursos utilizados sin directivas y el segundo usando las directivas de pipeline y DSP.

Recurso	Sys Gen	HLS_nd	HLS
SLICE	51	109	55
LUT	118	275	105
FF	6	157	126
DSP	8	0	8
BRAM	0	0	0

Table 1. Estimación de recursos.

Considerando las columas Sys_Gen y HLS se puede observar, a diferencia de la cantidad de flip-flops, que existe poca variabilidad entre el resto de los recursos. Sin embargo, al no ustilizar directivas en HLS, la distribución de los recursos cambia notablemente. El tiempo que le insume a la FPGA obtener el resultado, cuando se procesa una imagen de 8x8 píxeles RGB, cuyo diseño se

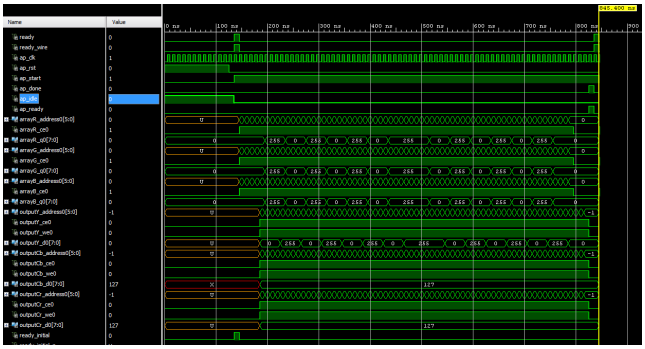


Fig. 11. Resultado de simulación para el diseño con HLS.

Fig. 12. Imagen de entrada para la medición de tiempo de procesamiento del bloque desarrollado en HLS.

efectuó mediante HLS es de 845 ns, mientras que al utilizar System Generator el tiempo es de 640 ns. El mismo se puede observar en la Fig 11, donde el marcador amarillo indica el final de procesamiento. Para esta medición de tiempo, se utilizó la imagen de entrada que se muestra en la Fig. 12.

5 Conclusiones

MPEG-7 es un estándar que permite la representación de contenido multimedia mediante la generación de descriptores. Adicionalmente, su papel en el procesamiento digital de señales en 2D resulta fundamental incluso en aplicaciones específicas de bio-medicina, control de tráfico, telecomunicaciones, entre otras.

En este trabajo, se presentó el diseño e implementación del descriptor denominado Color Layout de MPEG-7 utilizando plataformas paralelas FPGA basadas en SoCs. Tanto las FPGA como los SoC se perfilan como las alternativas de buen rendimiento para la extracción de los descriptores de MPEG-7 en tiempo real, debido a que las diversas extracciones pueden realizarse en forma paralela con el adicional de que cada operación de extracción es altamente paralelizable.

Se utilizaron y compararon en términos de utilización de recursos las herramientas System Generator y Vivado HLS, el cual hace uso de C/C++. Ambas se perfilan como buenas herramientas al momento de diseñar hardware a alto nivel. En el caso particular de cambio de espacio de color, la diferencia de los recursos utilizados no difieren en grandes valores, quedando a una elección subjetiva el uso de una herramienta u otra.

References

1. Yesun Joung, Kyuheon Kim, Kyeongok Kang, *The development of MPEG-7 interface over MPEG-4*, Consumer Electronics, pp.276-277 , 2003.
2. Myungseok Ki, Kyuheon Kim, *MPEG-7 over MPEG-4 systems decoder for using metadata*, Consumer Electronics. ICCE, pp. 245 - 246 , 2006.
3. I. Sezan, P. van Beek, *MPEG-7 standard and its expected role in development of new information appliances*, Consumer Electronics. ICCE. pp. 274 - 275, 2000.
4. Yong Rui, Huang, T.S., Shih-Fu Chang, *Digital image/video library and MPEG-7: standardization and research issues*, Acoustics, Speech and Signal Processing, Vol. 6, pp. 3785 - 3788, 1998.
5. V. V. Vinod, A. Lindsay , *MPEG-7: its impact on research, industry, and the consumer*, Multimedia Computing and Systems, pp. 7-11, 1999.
6. N.H. Eltonsy, G.D. Tourassi, A. Fadeev, A.S.Elmaghraby, *Significance of MPEG-7 Textural Features for Improved Mass Detection in Mammography*, Engineering in Medicine and Biology Society, pp.4779-4782, 2001.
7. N.H. Eltonsy, G.D. Tourassi, A. Fadeev, A.S.Elmaghraby, *Chaotic Synchronization of MPEG-7 Descriptors for Interpretation in Surveillance Video*, Circuits and Systems for Video Technology, Vol. 11, pp.688 - 695, 2001.
8. Zhenni Li, Jingjiao Li, Liang Li, Yue Zhao, Chaoqun Rong, *A SoC design and implementation of dynamic image edge detection based on the LEON3 open source processor*, Natural Computation, pp. 1263 - 1267, 2013
9. M. Borda,B. Belean, R. Terebes, R. Malutan, *FPGA based SoC for automated cDNA microarray image processing*, E-Health and Bioengineering Conference, pp. 1 - 4, 2011
10. Ahirwal, B., Khadtare, M., Mehta, R., *FPGA based system for color space transformation RGB to YIQ and YCbCr*, pp. 1345 - 1349, 2007.
11. Lamjed Touil, Abdessalem Ben Abdelali, Mtibaa Abdellatif, Elbey Bourennane, *R'G'B' to Y'CbCr color space conversion Using FPGA*, Wireless, Mobile and Multimedia Networks, pp. 255 - 258, 2008.
12. Behnam, B., Mansouryar, M., *Modeling and simulation of a DC motor control system with digital PID controller and encoder in FPGA using Xilinx system generator*, Instrumentation Control and Automation, pp. 104-108, 2011.
13. Thangavelu, A., Varghese, M.V., Vaidyan, M.V, *Novel FPGA based controller design platform for DC-DC buck converter using HDL Co-simulator and Xilinx System Generator*, Industrial Electronics and Applications, pp. 270 - 274, 2012.
14. Vidal, M, Cruces, R. ; Zurita, G., *Digital FIR filter design for diagnosing problems in gears and bearings using Xilinx's system generator*, 2014.
15. Vidal, M, Cruces, R. ; Zurita, G., *Implementation and simulation of IIR digital filters in FPGA using MatLab system generator*, Colombian Workshop on Circuits and Systems (CWCAS), pp. 1-5, 2014.
16. Alareqi Mohammed, Elgouri Rachid, Hlou Laamari, *High Level FPGA Modeling for Image Processing Algorithms Using Xilinx System Generator*, Journal of Computer Science and Telecommunications, Vol. 5, Issue 6, pp. 1-8, 2014.
17. Neha. P. Raut, Prof.A.V.Gokhale, *FPGA Implementation for Image Processing Algorithms Using Xilinx System Generator*, IOSR Journal of VLSI and Signal Processing, Vol. 2, Issue 4, pp. 26-36, 2013.
18. Aniket A. Ingle, Vrushali G. Raut, *Hardware software co-simulation of edge detection for image processing system using delay block in XSG*, Journal of Research in Engineering and Technology, Vol. 3, Issue 5, pp. 549-553, 2014.

XIII Workshop Computación Gráfica, Imágenes y Visualización (WCGIV)

7427 | AnArU, a Virtual Reality Framework for Physical Human Interactions
Matías Nicolás Selzer (UNS), Martín Larrea (UNS)

7440 | Detección de bordes basada en imágenes simultáneas de rango e intensidad
Luciano Lorenti (UNLP),
Javier Giacomantone (UNLP),
Armando E. De Giusti (UNLP)

7527 | Corrección de Modelos Digitales de Elevación con Imágenes Multi-Espectrales
Amalia L. Thomas (UNCPBA),
Juan Pablo D'Amato (UNCPBA),
Pablo R. Rinaldi (UNCPBA)

7547 | Visualización 3D de terrenos multirresolución basada en Shader Model 5
Lucas Borrelli (UNLP), María José Abásolo (UNLP)

7614 | A Serious Game about Recycling Rules
Jofré (UNSL), G. Rodríguez (UNSL),
Y Alvarado (UNSL), J. Fernandez (UNSL),
R. Guerrero (UNSL)

AnArU, a Virtual Reality Framework for Physical Human Interactions

Matías Selzer[1,2], Martín Larrea[1,2]
[1]Departamento de Ciencias e Ingeniería de la Computación
[2]Laboratorio de Investigación y Desarrollo en Visualización y Computación Gráfica
Universidad Nacional del Sur
Av. Alem 1253, Bahía Blanca
Buenos Aires, Argentina
{matias.selzer, mll}@cs.uns.edu.ar

Abstract. Virtual Reality has become, once again, a popular and interesting topic, both as a research and commercial field. This trend has its origin in the use of mobile devices as computational core and displays for Virtual Reality. Android is one of the most used platform in this context and Unity3d is a suitable graphic engine for such platform. In order to improve the immersive experience, some electronic devices, Arduino especially, are used to gather information, such as the movement of the user's arms or legs. Although these three elements are often used in Virtual Reality, few studies use all of them in combination. Those who do, do not develop a reusable framework for their implementations. In this work we present AnArU, a framework for physical human interaction in Virtual Reality. The goal of AnArU is to allow an easy, efficient and extensible communication between electronic devices and the Virtual Reality system.

Keywords. Virtual Reality, Arduino, Android, Unity3d, Human Computer Interaction.

1. Introduction

In recent years, there has been a spate of interest in Virtual Reality and its interactions, especially regarding the creation of Head Mounted Displays (HMD) using mobile devices as the computational core ([1, 2, 3, 4, 5]). We have seen mobile phones and small tablets become an ideal platform for Virtual Reality (VR). The current generation of these devices have full color displays, integrated cameras, fast processors and even dedicated 3D graphics chips.

From a Human Computer Interaction perspective, the majority of studies have focused on the visual aspects of a VR experience ([4, 5, 6]), even the interactions with the VR world are solved through visual elements. For instance, in [5] when users want to touch a virtual button, they must first look at it inside the HMD and then click a physical fix positioned one. No matter where users are looking, they always used

the same physical button. This, of course, decreases the immersive experience resulting in unpleasant results.

The use of Game Engines for the creation of VR content has been extensively studied in the past years ([7, 8, 9]). They provide developers a fast way of develop virtual content without the necessity of program directly in low-level languages. They also allow the application to process input from many different sources, including keyboards, cameras and microphones. For this project we decided to use Unity3d, not only for the creation of virtual content, but also because it has a plugin architecture that allows developers to extend the core functionality. Furthermore, Unity3d allows the exportation to many platforms, including Android.

There have been many publications addressing the problem of creating a more natural interaction between the user and the virtual environment, most of them include electronic devices ([15, 16]). However, each of these investigations has solved one particular problem and none of them has considered the creation of a practical and extensible framework for the communication of these electronic devices to the VR system.

The aim of this paper is to present a framework called AnArU (Android, Arduino, Unity3d) which allows an easy, efficient and extensible communication between electronic devices and the VR system by combining Android, Arduino and Unity3d.

In the next section we provide a brief introduction to the background concepts related to the AnArU Framework, and then discuss some relevant work done in this topic. Next we describe the AnArU Framework and its architecture, follow by the case study used to test it. Finally, we conclude with some remarks on the framework and directions for future research.

2. Background

In this section, we provide a brief introduction to the three main component of the AnArU Framework: Android, Arduino and Unity3d.

2.1. Android

Android[1] is a mobile operating system (OS) based on the Linux kernel and currently developed by Google. With a user interface based on direct manipulation,

[1] www.android.com

Android is designed primarily for touchscreen mobile devices such as smartphones and tablet computers.

Its source code is released by Google under open source licenses. This has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven projects, which add new features for advanced users. What is more, that brings the possibility of installing Android to devices with other operating systems.

Taking advantage of these benefits, in a similar approach as [2, 3, 5], we developed a Head Mounted Display by using the power of a smartphone running Android.

2.2 Unity3d

Unity3d[2] is a cross-platform game engine developed by Unity Technologies and it is widely used to develop video games for PC, consoles, mobile devices and websites. Unity3d was developed with an emphasis on portability, thus it is capable of porting to Windows, Xbox 360, Mac, Android and iOS.

We chose Unity3d for AnArU Framework because it is available for free, works very well with Android and it is easy to use.

2.3 Arduino

Arduino is an open-source computer hardware and Software Company[3] that designs and manufactures microcontroller-based kits for building digital devices and interactive objects that can sense and control the physical world. Their products are commonly known as Arduino and they are available commercially in preassembled form. The hardware design specifications are openly available, allowing the Arduino boards to be manufactured by anyone. It is estimated that in mid-2011 more than 300,000 official Arduinos were commercially produced and in 2013 more than 700,000 official boards were in users' hands.

We decided to use Arduino UNO for this project because it is one of the most popular, powerful and cheap microcontrollers nowadays, and also there have been many investigations which use it to create intuitive Human Computer Interaction devices ([11, 15]).

---

[2] www.unity3d.com

[3] www.arduino.cc

3. Previous Work

The combination of Android, Arduino and Unity3d as a platform for VR is not very common. There are very few and recent works published about the three of them associated.

In many investigations, virtual worlds are created in the context of cultural heritage; that is the case of [10], which works with archaeological sites. The virtual constructions are done using Unity3d and the platform for visualization is Android. An Arduino is used to gather information such as orientation of the viewer, physical location, tilt, pan and other movements of the tablet.

Lyons et al. ([11]) developed Loupe, a handheld near-eye display. Although it is not a HMD, it is very similar. In this case Unity3d is used for the Loupe's GUI, Android is the computational core and Arduino provided sensor information.

In contrast to what happens in VR, there are several investigations about Augmented Reality (AR) using Android, Arduino and Unity3d. AR is defined as a live direct or indirect view of a physical, real-world environment whose elements are augmented (or enhanced) by computer-generated sensory input, such as sound, video or graphics ([17]).

We can found several research papers that used this set of elements ([12, 13, 14]) in AR, but all these works, as well as the ones about VR, developed a solution for just a particularly problem. None of them thought about implementing a reusable framework.

4. AnArU Framework

AnArU consists of three main modules: a Unity3d Module running on an Android device; an Arduino Uno Module, responsible for controlling any electronic device attached to it; and a Java Plugin responsible for the communication between the other two. An overview of the framework is shown in Figure 1.

4.1. Arduino Module

The main purpose of the Arduino Module is to obtain information from any connected electronic device and send it to the rest of the system via a Bluetooth communication. Thus, those electronic devices can be used as interactive devices in the Virtual Reality system. The essential components of this module are an Arduino Uno board and a Bluetooth shield, responsible for the communication between the Arduino Module and the rest of the system.

The application running on Arduino is in charge of two simple tasks: waiting for the arrival of any message, and sending new messages whenever necessary. Thus, a new level of abstraction is introduced because any user can communicate by using such interface regardless of the connected devices. The communication protocol will be explained later on this section.

Once users know how to communicate, they can connect their particular devices or peripherals to the Arduino Module and start sending messages to the application running on the other modules of the framework. Furthermore, they can prepare the system to do specific tasks when any special message is received.

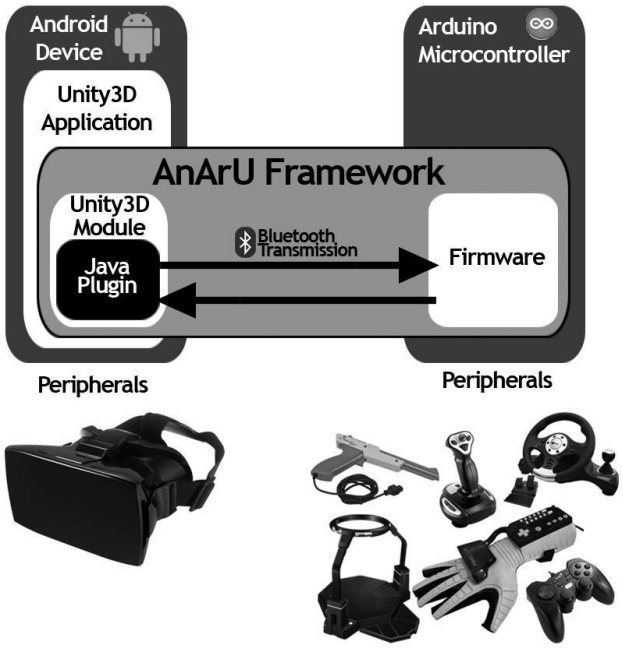


Fig. 1. AnArU Framework overview

4.2 Android Java Plugin

A Java Plugin is another part of AnArU Framework. It establishes a bridge between the Arduino Module and the Unity3d Module. Besides, as this plugin runs on Android devices, it has access to native Android libraries and properties, such as Bluetooth or gyroscope information, which are not available for a normal Java plugin.

The Plugin first task is connecting to the Arduino Module by using the Bluetooth libraries. To accomplish this, the Arduino Bluetooth shield MAC is required. Thus, once the connection is established, a thread is executed in order to wait for any received message. A function is provided to send messages from and to the Arduino Module.

As any electronic device works with many different values, a protocol was implemented in order to get a generic message format. No matter what device is connected to the Arduino, the user has to package the respective values into a single string of text, surrounded by special characters. Then, when needed, this string can be sent. On the other side of the framework, the Java Plugin automatically gets any new message but, in order to save memory, only keeps a backup of the last message received. Note that the protocol is Asynchronous in order to save time. However, a retransmission system was implemented to detect and recover broken or lost messages. On the other hand, when the Java Plugin wants to send a message to the Arduino Module, an analog communication takes place.

4.3 Unity3d Module

The Unity3d Module consists on a collection of scripts that allow the initialization and communication of the Java Plugin. Hence, users can make any application in Unity3d, communicate to the Arduino Module by using the provided methods, and do anything they need with the received values. Note that the values come in form of a string of text. However, as the user knows the specific structure of that string, it can be parsed in an easily manner and the values can be retrieved.

5. Study Case

In recent years, numerous investigations has focused on the ability of walking in Virtual Reality environments, leading to an increment of users' immersion ([18]). Our current investigation involved building an omnidirectional walking platform controlled by Arduino which communicates to a VR application running on an Android mobile device, by using AnArU Framework.

The platform consists of a circular wood base and iron pipes with a ring in the middle in order to hold the users inside of it. In addition, elastic ropes are used to hold users exactly in the middle of the platform. They are also equipped with a pair of rollers so they can walk freely inside the ring while maintaining their position.

In order to sense users' movements, an Arduino UNO Microcontroller and a gyroscope are attached to one of their legs. Thus, by measuring the leg angle respecting to the vertical, the system knows whether users are walking or not, and the corresponding direction. Users are also wearing a Head Mounted Display running a Unity3d application which recreates a complete model of them in a virtual environment. That is, a virtual representation of the user's body and environment in an appropriate scale. By using AnArU Framework, a communication between the Arduino on the leg of the user and the HMD application is performed, giving users the sensation that they are really walking inside the virtual environment, increasing the immersion level. In Figure 2 a user with all mentioned components is shown.

Fig. 2. User on the omnidirectional platform. He is wearing a HMD running a Unity3d application on an Android tablet, and the Arduino Module is attached to his leg.

Baraka et al. ([19]) showed that in most cases the developer has to be very engaged in the designing of the communication protocol between Android and Arduino modules. In our study, the communication between the different modules is transparent. Hence, there is no need to configure a new communication each time a new way of interaction between these technologies is done. Lai et al. ([20]) used a

commercial Plugin to perform a similar connectivity. However, this Plugin is not open source and just a few aspects of the microcontroller are available.

During the first experimentations, we observed some problems when the communication went from Arduino Module to Unity Module and the other way around, at the same time. A retransmission mechanism was implemented to solve this problem, and hence, we have not observed any more similar issues since then.

In an ideal VR system, the minimal delay should exist between users' movements and the corresponding visualization ([21]). In general, a maximum accepted delay is in the order of milliseconds. Hereby, a quantitative speed analysis to measure the communication time, based on the transmission time of a specific size message**,** was applied. As the average speed of a Bluetooth 2.0 communication is 3Mbit/sec, we tested the communication time of a specific message containing three float values corresponding to the values provided by the gyroscope. For this message, with a size of 9 Bytes, after a few tests, we obtained an overall transmission time of 0.04ms, compared to an ideal transmission time of 0.024ms. In case longer messages are needed, we tested the communication time of messages with a size of 23 Bytes. This experiment showed an overall transmission time of 0.07ms. Hence, these results suggests that the communication time of AnArU Framework is fast enough to fulfill its objectives.

6. Conclusions

Prior work has documented some interactivity between Arduino, Android and Unity3d technologies. However, none of them define a transparent and extensible method of communicating these technologies. In this study we created and tested a framework capable of interconnecting these technologies in a straightforward way. Furthermore, we found that the communication is fast enough to satisfy the necessities of VR interactions.

AnArU Framework can be used for any VR interaction, helping developers to save time as they would not need to design any new communication protocol. However, some limitations are worth noting. Although Arduino UNO is a powerful tool for prototyping, it has little memory and computational power, so that, in some cases, it would not be functional.

Future work should therefore consider using a different microcontroller in case that more memory or computational power is needed. Other ways of communication should be considered too. USB or WiFi connectivity may be better in other contexts.

References

1. Amer, A., & Peralez, P. (2014). Affordable altered perspectives: Making augmented and virtual reality technology accessible. In *Global Humanitarian Technology Conference (GHTC), 2014 IEEE* (pp. 603-608). IEEE.
2. Olson, J. L., Krum, D. M., Suma, E., & Bolas, M. (2011, March). A design for a smartphone-based head mounted display. In *Virtual Reality Conference (VR), 2011 IEEE* (pp. 233-234). IEEE.
3. Petry, B., & Huber, J. (2015, March). Towards effective interaction with omnidirectional videos using immersive virtual reality headsets. In *Proceedings of the 6th Augmented Human International Conference* (pp. 217-218). ACM.
4. Hürst, W., & Helder, M. (2011, November). Mobile 3D graphics and virtual reality interaction. In *Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology* (p. 28). ACM.
5. Steed, A., & Julier, S. (2013, March). Design and implementation of an immersive virtual reality system based on a smartphone platform. In *3D User Interfaces (3DUI), 2013 IEEE Symposium on* (pp. 43-46). IEEE.
6. Pujol-Tost, L. (2011). Realism in Virtual Reality applications for Cultural Heritage. *International Journal of Virtual Reality*, *10*(3), 41.
7. Wang, S., Mao, Z., Zeng, C., Gong, H., Li, S., & Chen, B. (2010, June). A new method of virtual reality based on Unity3D. In *Geoinformatics, 2010 18th International Conference on* (pp. 1-5). IEEE.Shiratuddin, M. F., & Thabet, W. (2011). Utilizing a 3D game engine to develop a virtual design review system. *Journal of Information Technology in Construction–ITcon*, *16*, 39-68.
8. Shiratuddin, M. F., & Thabet, W. (2011). Utilizing a 3D game engine to develop a virtual design review system. *Journal of Information Technology in Construction–ITcon*, *16*, 39-68.
9. Jacobson, J., & Lewis, M. (2005). Game engine virtual reality with CaveUT. *Computer*, *38*(4), 79-82.
10. Davies, C. J., Miller, A., & Allison, C. (2012). Virtual Time Windows: Applying cross reality to cultural heritage. In *Proceedings of the Postgraduate Conference on the Convergence of Networking and Telecomunications*.
11. Lyons, K., Kim, S. W., Seko, S., Nguyen, D., Desjardins, A., Vidal, M., ... & Rubin, J. (2014, October). Loupe: a handheld near-eye display. In *Proceedings of the 27th annual ACM symposium on User interface software and technology* (pp. 351-354). ACM.
12. Olmedo, H., & Augusto, J. (2013). Towards the Commodification of Augmented Reality: Tools and Platforms. In *New Trends in Interaction, Virtual Reality and Modeling* (pp. 63-72). Springer London.
13. Lin, C. F., Pa, P. S., & Fuh, C. S. (2013, October). Mobile application of interactive remote toys with augmented reality. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific* (pp. 1-6). IEEE.

14. Lin, C. F., Pa, P. S., & Fuh, C. S. (2014). A MAR Game Design via a Remote Control Module. In *Augmented and Virtual Reality* (pp. 3-18). Springer International Publishing.
15. Schmidt, D., Kovacs, R., Mehta, V., Umapathi, U., Köhler, S., Cheng, L. P., & Baudisch, P. (2015, April). Level-Ups: Motorized Stilts that Simulate Stair Steps in Virtual Reality. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (pp. 2157-2160). ACM.
16. Blake, J., & Gurocak, H. B. (2009). Haptic glove with MR brakes for virtual reality. *Mechatronics, IEEE/ASME Transactions on*, *14*(5), 606-615.
17. Azuma, R. T. (1997). A survey of augmented reality. *Presence*, *6*(4), 355-385.
18. Cakmak, T., & Hager, H. (2014, July). Cyberith virtualizer: a locomotion device for virtual reality. In *ACM SIGGRAPH 2014 Emerging Technologies* (p. 6). ACM.
19. Baraka, K., Ghobril, M., Malek, S., Kanj, R., & Kayssi, A. (2013, June). Low cost arduino/android-based energy-efficient home automation system with smart task scheduling. In *Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on* (pp. 296-301). IEEE.
20. Lai, A. S., & Leung, S. Y. (2013, December). Mobile Bluetooth-Based Game Development Using Arduino on Android Platform. In *Applied Mechanics and Materials* (Vol. 427, pp. 2192-2196).
21. Earnshaw, R. A. (Ed.). (2014). *Virtual reality systems*. Academic press.

Detección de bordes basada en imágenes simultáneas de rango e intensidad

Luciano Lorenti, Javier Giacomantone, Armando De Giusti

Instituto de Investigación en Informática (III-LIDI),
Facultad de Informática - Universidad Nacional de La Plata - Argentina.
La Plata, Buenos Aires, Argentina.
{llorenti,jog,degiusti}@lidi.info.unlp.edu.ar

Resumen. Este artículo presenta un método de detección de bordes que mejora la calidad de los mismos utilizando dos imágenes de la misma escena, una de rango y otra de intensidad. Esto permite abordar problemas de segmentación donde la información de intensidad o de rango separadamente es insuficiente para detectar bordes continuos. El método propuesto opera sobre un grafo pesado en el cual elimina aristas espurias y une fronteras abiertas. Utiliza una función de costo que considera información espacial y topológica. Se presentan resultados experimentales sobre imágenes reales, utilizando cámaras de tiempo de vuelo (TOF), que permiten evaluar el método propuesto.

Palabras clave: Segmentación, Detección de Bordes, Cámaras de Tiempo de Vuelo.

1. Introducción

La detección de bordes o segmentación basada en discontinuidades es una etapa fundamental en los sistemas de visión por computador [1]. Métodos de segmentación utilizados con éxito en ambientes industriales, con colores e iluminación controlada, no obtienen resultados similares en contextos diferentes. Una alternativa para abordar problemas en que las condiciones de contorno no permiten una segmentación adecuada es incorporar información de profundidad, es decir, la distancia a la que se encuentran los objetos que conforman la escena respecto al dispositivo de captura [2] [3]. En este contexto, la segmentación de imágenes consiste en utilizar algoritmos que utilicen ambas fuentes de información y no sólo los niveles de intensidad [4][5]. Distintos métodos de detección de bordes basados en operadores derivativos de primer y segundo orden sobre una imagen dada permiten obtener un mapa de bordes. Los mapas de bordes que éstos métodos generan están compuestos por un conjunto de puntos ubicados sobre las fronteras de los objetos en la imagen. Los mapas de bordes, por lo general, presentan discontinuidades e irregularidades [6]. Dado un mapa de bordes la representación más directa de alto nivel consiste en resolver las discontinuidades y cerrar contornos por proximidad, similitud y simetría. Métodos basados en

teoría de grafos han sido propuestos para enlazar bordes discontinuos y eliminar bordes espurios [6][7]. El método propuesto en [6] obtiene la triangulación de Delaunay de los puntos de bordes y luego calcula el árbol recubridor mínimo de los mismos. La poligonal formada por el árbol recubridor mínimo conforma la información de bordes inicial para el método propuesto. Han sido propuestas recientemente distintas estructuras de grafos que pueden ser definidas sobre un conjunto de puntos en el plano. El objetivo es encontrar el grafo que preserve la relevancia perceptual de los patrones. Estos grafos incluyen el árbol recubridor mínimo, los grafos Gabriel, los grafos de vecindad relativa, los grafos de esferas de influencias y la triangulación de Delaunay. En este trabajo se presenta un método de clausura de contornos obtenidos a partir de imágenes capturadas con una cámara de tiempo de vuelo. En una primera etapa se obtiene el grafo de esferas de influencias generado a partir de los puntos de bordes. Luego se filtra la información de bordes utilizando operaciones morfológicas definidas sobre el grafo. A partir de la información de bordes filtrada se enlazan los bordes abiertos mediante la minimización de una función de costo. Por último se fusiona la información de bordes de las dos imágenes.
El artículo está organizado del siguiente modo, en la sección 2 se describe el método propuesto. En la sección 3 se muestran los resultados experimentales obtenidos. Finalmente, en la sección 4 se presentan las conclusiones.

2. Detección y enlazado de bordes discontinuos

La técnica propuesta consiste en cuatro etapas consecutivas: primero los puntos de bordes representados en un espacio bidimensional son considerados como vértices de un grafo pesado. Las aristas conectan dos vértices si se encuentran dentro de un radio determinado. El peso asociado a cada arista corresponde a una función de distancia definida entre los puntos. Como segunda etapa se realiza un proceso de filtrado basado en operaciones morfológicas. La tercera etapa consiste en enlazar bordes mediante la minimización de una función de costo. Por último se fusiona la información de bordes enlazados de las dos imágenes para obtener el resultado final.

2.1. Construcción del grafo

Sea $I(r, c)$ una imagen de R filas y C columnas y sea $B(r, c)$ la información de bordes de la imagen obtenida previamente mediante un método de detección de bordes. El objetivo de esta etapa consiste en conectar aquellos puntos de bordes próximos, evitando generar aristas inexistentes. Para esto se calcula el grafo de esferas de influencias con los puntos de bordes detectados. Este grafo de proximidad fue introducido por Toussaint [8] para modelar problemas de visión por computador y de reconocimiento de patrones. Una de las características principales que diferencian al grafo de esferas de influencias con respecto a otros grafos de proximidad, es que el grafo de esferas de influencia no es necesariamente conexo. Los grafos de esferas de influencias producen componentes desconectados

que concuerdan con la percepción humana sin la necesidad de la configuración de parámetros [9]. Sea $X = \{x_1, .., x_n\}$ un conjunto de puntos en el plano. Sea r_i la distancia mínima entre x_i y su vecino más próximo. La esfera de influencia de x_i es el círculo con centro en x_i y radio r_i. El grafo de esferas de influencias $G = (V, E)$ tiene $V = X$ con una arista uniendo un par de vértices distintos siempre que las esferas de influencias correspondientes se intersecten, como se muestra en la figura 1. Con el objetivo de evitar la presencia de múltiples aristas en las esquinas de los objetos se impone una restricción adicional. Dos vértices son conectados entre sí, si las esferas de influencias se intersectan y además no existe un camino menor a λ aristas que los una.

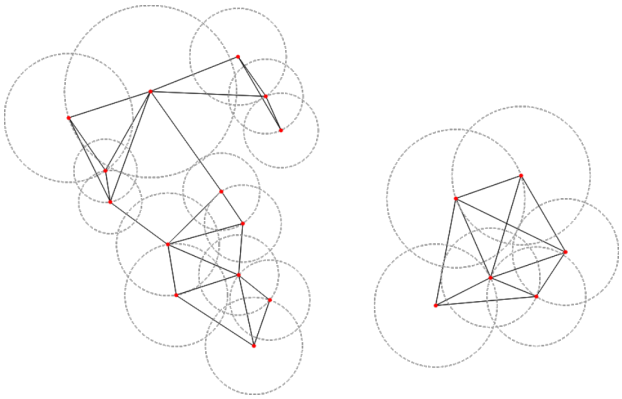


Figura 1. Grafo de esferas de influencias

2.2. Filtrado de grafo

Al construir el grafo de bordes fueron generadas algunas pequeñas ramificaciones conectadas con las aristas de bordes principales. Estas ramificaciones se deben a información redundante o ruidosa. Con el objetivo de realizar el proceso de eliminación de las mismas, se utiliza el método de filtrado morfológico propuesto en [6]. El método ilustrado en la figura 2 utiliza conceptos de morfología matemática para realizar un algoritmo de apertura sobre el grafo. Este algoritmo consiste en realizar un proceso iterativo de erosión seguido de una etapa de dilatación aplicada tantas veces como requirió la de erosión. El algoritmo de apertura asume a cada arista como elemento básico de procesamiento.

En este contexto la erosión es definida como la eliminación de las aristas que conecten a vértices de grado 1. La etapa de erosión es aplicada n veces en cada iteración y todos los segmentos finales son eliminados. El número de iteraciones se determina automáticamente de acuerdo a la diferencia entre los elementos eliminados en cada iteración: $\delta = Re(t) - Re(t-1)$, donde $Re(t)$ representa la cantidad de elementos removidos en la iteración t. El proceso de erosión termina

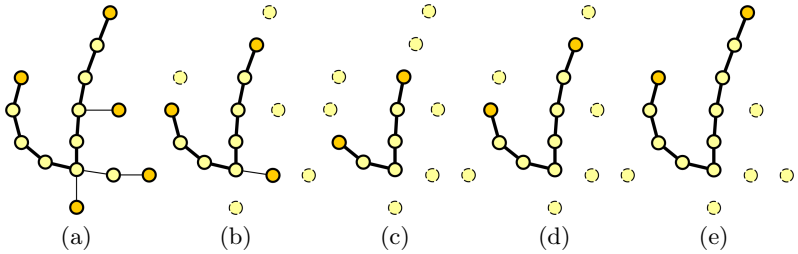


Figura 2. Dos iteraciones de la operación de apertura. (a) Grafo Inicial. (b) Grafo luego de la primer erosión. (c) Grafo luego de la segunda erosión. (d) Grafo luego de la primera dilatación. (e) Grafo luego de la segunda dilatación.

cuando la diferencia δ es cero en al menos r iteraciones consecutivas. Luego de finalizar el proceso de erosión, se realizan n dilataciones. Las dilataciones son llevadas a cabo sobre los segmentos no erosionados. La etapa de dilatación consiste en volver a insertar los segmentos que estaban conectados con cada uno de los vértices de grado 1 presentes en cada iteración. Por lo tanto es necesario almacenar los segmentos no eliminados durante las etapas de erosión. Los puntos que quedaron desconectados al finalizar la etapa de erosión son eliminados de la imagen binaria $\mathbf{B}(r, c)$.

2.3. Algoritmo de enlace de bordes

Luego de remover las pequeñas ramificaciones, el algoritmo se enfoca en detectar y cerrar los contornos abiertos. Estos contornos son caracterizados por aristas que conectan al menos un vértice de grado 1. Primeramente son detectados estos vértices extremos. Luego, por cada punto extremo j se seleccionan un conjunto de puntos de bordes candidatos para unir con el punto j. Los puntos de bordes candidatos son seleccionados por medio de un proceso iterativo sobre un círculo de radio p centrado en el punto j. Luego de extraer el conjunto de puntos candidatos $V_p(i)$ en una iteración, se calcula el costo $E(j, k)$ de conectar cada uno de los puntos posibles $k \in V_p(i)$ con el punto dado.

$$E(j,k) = d(j,k) * (1 + \tfrac{\sigma_i}{\mu_i} * \tfrac{\sigma_r}{\mu_r})$$

$d(j, k)$ es la distancia euclídea 2D entre el punto j y el punto k y $(1 + \frac{\sigma_i}{\mu_i} * \frac{\sigma_r}{\mu_r})$ puede ser considerado como un factor de ponderación: σ_i y σ_r representan la desviación estándar de los niveles de intensidad y de rango asociados con el conjunto de píxeles que define el segmento que une al vértice j con el vértice k, denominado S_{jk}. μ_i y μ_r representan el valor de la media de intensidad y de rango, respectivamente, de ese conjunto de píxeles. Los puntos definidos por S_{jk} pueden ser calculados mediante el algoritmo de Bresenham [10]. La idea principal de este factor de ponderación consiste en evitar unir puntos de borde con un valor de intensidad o rango similar, pero a través de un área con diferentes valores de

intensidad o rango. La varianza de los valores de los puntos que definen a S_{jk} son multiplicados por un factor de escala σ_i y σ_r para obtener un peso relativo a los valores en esa área [6].

2.4. Fusión de aristas

Por lo general las aristas de la imagen de intensidad no coinciden con las aristas de la imagen de rango. En [5] se muestra que hay un desplazamiento horizontal de un píxel hacia la derecha o hacia la izquierda, esto se debe a distorsiones geométricas en el proceso de adquisición de la cámara o a variaciones introducidas por el método de extracción de contornos. Si una arista de la imagen de intensidad se encuentra cerca de una arista de la imagen de rango y ambas tienen la misma dirección, son consideradas como representantes de la misma frontera del objeto. En este caso, la arista de la imagen de intensidad puede ser preservada mientras que la arista de la imagen de rango puede ser descartada [11]. Sea $G_I = (V_I, E_I)$, $G_R = (V_R, E_R)$ los grafos filtrados obtenidos a partir de la imagen de intensidad y de la imagen de rango respectivamente. Primeramente se inicializa el conjunto de bordes final con E_R. A continuación se calcula el punto medio y la dirección de cada arista perteneciente a E_I y E_R. Luego por cada arista de $a_i \in E_I$ con centro en c_{a_i} se buscan las aristas $V(a_i) = \{a_r \in E_R | \ ||c_{a_r} - c_{a_i}||_2 < r\}$ donde c_{a_r} es el centro de la arista a_r. Si $V(a_i) = \emptyset$ la arista a_i es agregada al conjunto de bordes final. Si $V(a_i) \neq \emptyset$ y la arista de intensidad a_i difiere en orientación por mas de g grados con las $a_r \in V(a_i)$ la arista a_i es agregada al conjunto de bordes final.

2.5. Descripción de método propuesto

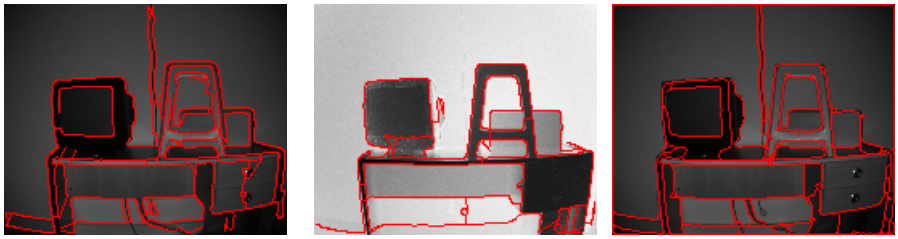
Sea $I(r, c)$ la imagen de intensidad y $B_I(r, c)$ la información de bordes de la imagen de intensidad y sea $R(r, c)$ la imagen de rango y $B_R(r, c)$ la información de bordes de la imagen de rango.

1. *a*) Se obtiene el grafo de esferas de influencias G_I de los puntos $\{(x, y) \in B_I(r, c)\}$.
   b) Se obtiene el grafo de esferas de influencias G_R de los puntos $\{(x, y) \in B_R(r, c)\}$.
2. *a*) Se realiza el proceso de apertura sobre el grafo G_I.
   b) Se realiza el proceso de apertura sobre el grafo G_R.
3. Se fusiona la información de G_I y G_R para formar el grafo G_{IR}
4. Se enlazan los contornos del grafo G_{IR}
5. Se realiza el proceso de apertura sobre el grafo G_{IR}

3. Resultados experimentales

El rendimiento del algoritmo de segmentación propuesto fue evaluado sobre 50 imágenes capturadas utilizando la cámara de tiempo de vuelo MESA

SwissRanger SR4000 [12] y el conjunto de datos completo facilitado por el Laboratorio de Tecnología Multimedia y Telecomunicaciones de la Universidad de Padua [13]. La cámara de tiempo de vuelo MESA SwissRanger SR4000 proporciona dos imágenes: una imagen de amplitud y una imagen de rango ambas de 144×176 píxeles. El conjunto de datos [13] contiene imágenes capturadas con una cámara de tiempo de vuelo y una cámara RGB tradicional.



(a) (b) (c)

Figura 3. Detección de bordes utilizando el método propuesto

La figura 3 presenta resultados experimentales del método propuesto aplicado a una imagen obtenida con la cámara de tiempo de vuelo MESA SwissRanger SR4000. Las figuras 3(a) y 3(b) muestran los resultados obtenidos al aplicar el método propuesto por Sappa [6] sobre las imágenes de intensidad y rango respectivamente. La figura 3(c) muestra los bordes obtenidos aplicando el método propuesto.

(a) (b) (c)

Figura 4. Extracción de bordes utilizando el método propuesto

La figura 4 muestra el resultado de aplicar el algoritmo propuesto sobre una imagen del conjunto de datos de la Universidad de Padua. La figura 4(a) muestra los bordes obtenidos aplicando el método de Canny sobre la imagen de amplitud. La figura 4(b) muestra la imagen de rango con la información

de bordes obtenida mediante el método de Canny. La figura 4(c) muestra el resultado de aplicar el algoritmo propuesto. Se puede observar como el método propuesto cierra correctamente los contornos de los objetos.

Las figura 5 muestran otra captura real generada mediante la cámara de tiempo de vuelo SwissRanger SR4000. La figura 5(a) y 5(b) muestra la información de bordes obtenida mediante el método de Canny aplicado a la imagen de intensidad y de rango respectivamente. La figura 5(c) muestra los bordes utilizando el método propuesto.

(a) (b) (c)

Figura 5. Generación de bordes utilizando el método propuesto

4. Conclusiones

En este artículo presentamos un método de detección de bordes que utiliza imágenes de rango e intensidad. Los resultados experimentales muestran que el método propuesto combina información de ambas imágenes generando bordes mas conectados. Los resultados comparativos presentan, para el método propuesto, una reducción de bordes adicionales generados en el proceso de segmentación. Una etapa de trabajo futuro prevé extender los resultados experimentales e incorporar información de color en la función de costo.

Referencias

1. J. Canny, "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-8, pp. 679–698, Nov 1986.
2. D. Holz, S. Holzer, R. B. Rusu, and S. Behnke, "Real-time plane segmentation using rgb-d cameras," in *RoboCup 2011: Robot Soccer World Cup XV*, pp. 306–317, Springer, 2012.
3. G. M. Hegde and C. Ye, "A recursive planar feature extraction method for 3d range data segmentation," in *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*, pp. 3119–3124, IEEE, 2011.
4. A. Bleiweiss and M. Werman, "Fusing time-of-flight depth and color for real-time segmentation and tracking," in *Dynamic 3D Imaging*, pp. 58–69, Springer, 2009.

5. G. Danciu, M. Ivanovici, and V. Buzuloiu, "Improved contours for tof cameras based on vicinity logic operations," in *Optimization of Electrical and Electronic Equipment (OPTIM), 2010 12th International Conference on*, pp. 989–992, May 2010.
6. A. D. Sappa and B. X. Vintimilla, "Cost-based closed-contour representations," *Journal of Electronic Imaging*, vol. 16, no. 2, pp. 1–9, 2007.
7. A. D. Sappa, "Efficient closed contour extraction from range image's edge points," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 4333–4338, IEEE, 2005.
8. G. T. Toussaint, "The sphere of influence graph: Theory and applications,"
9. T. Michael and T. Quint, "Sphere of influence graphs in general metric spaces," *Mathematical and Computer Modelling*, vol. 29, no. 7, pp. 45–53, 1999.
10. T. Pavlidis, *Algorithms for graphics and image processing*. Computer science press, 1982.
11. H. Yue, W. Chen, J. Wang, and X. Wu, "Combining color and depth data for edge detection," in *Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on*, pp. 928–933, Dec 2013.
12. M. Cazorla, D. Viejo, and C. Pomares, "Study of the sr 4000 camera," in *XI Workshop de Agentes Físicos*, 2004.
13. M. Technology and U. o. P. Telecommunications Laboratory, "Joint color and depth segmentation datasets," July 2014.

Corrección de Modelos Digitales de Elevación con Imágenes Multi-Espectrales

Amalia L. Thomas[1], Juan P. D'Amato[1,2], Pablo R. Rinaldi[1,3]

[1]UNCPBA, [2]CONICET, [3]CICPBA
rinaldipablo@gmail.com

Abstract. Se presenta una metodología extensiva para detectar áreas arboladas en zonas rurales de la provincia de Buenos Aires empleando imágenes satelitales multiespectrales *Landsat.* La máscara generada permite corregir las falencias de los Modelos Digitales de Elevación (MDE) radar como los *Shuttle Radar Topographic Mission* (SRTM). En estos modelos, la presencia de árboles genera errores en las mediciones ya que las ondas del radar no penetran la vegetación, dando valores de altura de terreno similares a las copas de los árboles. Con base en imágenes Landsat se ajustaron funciones de umbralado independientes para cada espectro. Con la aplicación sucesiva de estos umbrales se logra una máscara que se proyecta sobre el MDE. La máscara luego es corregida evaluando los saltos de altura en las fronteras de la misma. Por último, se aplica un algoritmo de corrección al MDE.

Keywords: MDE, Segmentación, Umbralado, Imágenes Multiespectrales.

1 Introducción

Los Modelos Digitales de Elevación (MDE) son representaciones digitales de información cartográfica en forma de raster, y consisten de arreglos cuyo contenido indica la elevación con respecto al nivel del mar de un número de puntos del suelo espaciados a intervalos regulares. Las técnicas de producción de MDEs generalmente implican un *trade-off* entre costo, precisión, cobertura espacial y resolución [1]. Numerosos autores como Hayakawa et *al.*[2], Sampson et *al*. [3] y Baugh et *al*. [4] reconocen al sistema LIDAR como el que provee mejor calidad de mediciones de elevación, aunque presentan costos muy altos para grandes extensiones por lo cual la disponibilidad de cobertura espacial es muy limitada. Por ejemplo, no se dispone de MDEs públicos de Argentina producidos por esta técnica. En cambio, los productos ASTER GDEM y SRTM son de cobertura global y gratuita. Ambos modelos tienen la misma resolución espacial, pero ASTER GDEM presenta un error considerablemente mayor, por esta razón se eligió utilizar MDEs SRTM en el presente trabajo.

En estos modelos, la técnica de medición empleada tiene el inconveniente de no penetrar la vegetación, y entonces los modelos de elevación toman como altura del suelo una aproximación del tope del follaje en la dirección de medición [5]. Este efecto es aún más significativo en zonas de llanura donde la pendiente del suelo es siempre menor a los saltos abruptos de altura desde la base al tope de los árboles. En

consecuencia, la vegetación puede generar una rugosidad artificial en las mediciones de elevación del suelo, afectando la habilidad de los modelos de escurrimiento superficial como AQUA [6] de identificar cauces naturales [7]. Tal es el caso de la mayor parte de las zonas agrícolas de la provincia de Buenos Aires, Argentina, cuya morfología presenta suaves pendientes y arboledas esporádicas que son utilizadas para sombra y protección. En estas zonas altamente productivas los árboles son plantados por el hombre y dispuestos en filas para proteger sembrados y caminos. Este tipo de vegetación es detectado como una lomada por el MDE que se traduce en una barrera al escurrimiento superficial que simulan los modelos hidrológicos.

1.1 Imágenes Multi-Espectrales

La localización de árboles, e incluso la identificación de especies de los mismos, a partir de imágenes satelitales fue ampliamente trabajada en parte gracias a la gran evolución de los satélites Landsat [8] [9]. La técnica más popular consta de un árbol de decisiones a partir de la evaluación de las bandas multiespectrales de Landsat [10] [11]. Esta consiste en establecer umbrales de aceptación para cada banda, y evaluarlos en cada pixel, pero estos umbrales que definen una especie son fuertemente dependientes de la especie y del clima.

La serie de imágenes *Landsat* destaca entre los sistemas proveedores de imágenes satelitales multi-bandas por el amplio rango de información que proporciona, la calidad y la precisión de los datos, su cobertura global, y su fácil y gratuita disponibilidad. Actualmente se encuentran en órbita dos satélites: Landsat-7 y Landsat-8, el último de los cuales proporciona imágenes de gran calidad de la Tierra desde 2013. Landsat-8 provee información en un rango más amplio de bandas para clasificar los tipos de suelos [12].

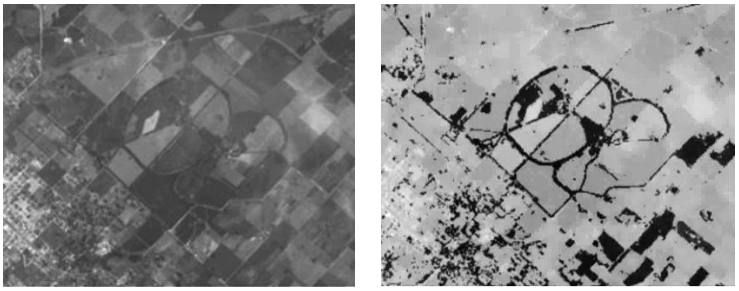
En estas imágenes, cada pixel consta de un Número Digital (DN) en formato de entero sin signo de 8-bits para Landsat-7 y de 16-bits para Landsat-8, que se puede mapear a una medición científicamente útil en unidades de radiancia espectral. El termino radiancia se utiliza para referirse a la cantidad, o más correctamente la densidad de potencia de radiación que recibe un sensor en un determinado tiempo, en tanto que la radiancia espectral se refiere a sólo la radiación de una longitud de onda, o banda de longitudes, en particular. Las unidades de esta magnitud es $Wm^{-2}\Omega^{-1}\mu m^{-1}$ (o sea potencia por unidad de área de la fuente, por unidad de ángulo sólido, por unidad de longitud de onda). Asimismo, la reflectancia espectral se calcula como la porción de la radiación incidente que fue reflejada en una superficie y varía entre 0 y 1. Es la reflectancia *Top Of de Atmosphere* (TOA) la que es característica de un tipo particular de suelo. Los DN se pueden convertir a reflectancia o radiancia planetaria (o TOA) a partir de una transformación de la escala radiométrica [13].

2 Desarrollo

2.1 Segmentación de la vegetación

Una manera muy difundida de identificar la vegetación es mediante el NDVI [14]. Este método tiene la ventaja de ser relativamente simple porque sólo necesita de la información contenida en las bandas roja e infrarroja cercana. Sin embargo, tiene la desventaja de depender de muchos factores ambientales sobre los que no se tiene control, como la fertilidad del suelo y las precipitaciones recientes. En pruebas realizadas se vio que el NDVI selecciona algunas áreas no forestadas, y no selecciona otras que sí lo están. El principal inconveniente radica en que este índice no considera todas las bandas relacionadas a la vegetación que provee Landsat.

En contraste, al realizar la segmentación a partir de las bandas verde, roja, infrarroja cercana e infrarroja lejana, y modificando manualmente los umbrales de cada banda independientemente se obtuvo un filtro mucho más satisfactorio. La Figura 1(a) muestra la imagen visible de una zona elegida para el estudio y la máscara de vegetación correspondiente obtenida determinando manualmente los umbrales para cada capa de la imagen Landsat-8 (b).



(a) (b)

Figura 1. Imagen visible de la zona 1 (a) y máscara resultante de la aplicación de filtros sobre una imagen satelital Landsat-8 (b).

Puede observarse en la esquina inferior izquierda, donde se visualiza un sector de la ciudad, que la máscara detecta incluso la vegetación urbana. Sin embargo, también se incluye en la máscara, zonas de cultivos que en la imagen a color natural aparecen como no forestadas. Este es el caso de los campos cultivados de la esquina inferior derecha. Más adelante, estos píxeles son eliminados de la máscara por el algoritmo de corrección en base a la diferencia de altura en los límites. Es por esto que se prefiere una máscara con falsos positivos en lugar de falsos negativos.

2.2 Determinación de Umbrales.

Para automatizar la obtención de los umbrales es necesario elegir algún criterio para determinar los valores máximos y mínimos de cada banda a partir de los cuales se identifica la vegetación. Como se mencionó anteriormente, para trabajar con una

magnitud que tiene significado físico y que caracteriza las propiedades del suelo, se deben convertir los valores de los pixeles de las imágenes Landsat a unidades de reflectancia planetaria o TOA. Esta transformación a su vez depende de constantes de calibración de los sensores y de propiedades particulares al momento en que se realizaron las mediciones. USGS provee toda esta información en el archivo de texto de metadatos con terminación "*_MTL*" que se descarga junto a las imágenes de cada banda.

En particular, el archivo de metadatos de los productos de Landsat-8 contiene el factor multiplicativo M_ρ y el aditivo A_ρ de escala de la reflectancia. Con estos, se puede obtener fácilmente el valor de la reflectancia TOA correspondiente a cada pixel, ρ_λ'. Por otro lado, el archivo de metadatos de los productos de Landsat-7 no contienen estos parámetros, pero sí al factor multiplicativo M_L y el aditivo A_L de escala de la radiancia con los que se puede calcular la radiancia espectral correspondiente L_λ. A partir del día juliano que puede obtenerse del nombre del archivo, es posible obtener la distancia d entre el sol y la Tierra en unidades astronómicas y, si además se conoce la irradiancia espectral solar exoatmosférica $E_{sol,\lambda}$ se puede calcular la reflectancia espectral ρ_λ'. La distancia d puede calcularse a partir del día juliano, y $E_{sol,\lambda}$ depende de la longitud de onda λ [15]. Aplicando estos cálculos a los pixeles de las imágenes de bandas se obtienen finalmente los valores de reflectancia espectral TOA que son propiedad específica del tipo de suelo. Se decidió no corregir la reflectancia espectral por el ángulo de zenit solar, porque éste involucra una función senoidal que varía con la estación coincidentemente con otros factores que alteran estos valores.

Para realizar un ajuste más preciso, se realizó un análisis de cómo varían los umbrales óptimos determinados manualmente de las bandas relacionadas con la vegetación (verde, roja, infrarroja cercana e infrarroja de onda corta 1) en función del día juliano. La Figura 2 muestra el gráfico resultante de la determinación de umbrales máximos (en rojo) y mínimos (en azul) para la banda infrarroja de onda corta para varias imágenes satelitales tomadas de la zona de estudio 1 entre el año 2000 y el 2003 (por Landsat-7, representados con triángulos), y entre 2013 y 2015 (por Landsat-8, representados con círculos). En primer lugar, resalta que los resultados para Landsat-7 y para Landsat-8 coinciden. Luego, se nota que a gran escala los puntos siguen una función senoidal, y entonces se podrían ajustar a una función del tipo

$$\rho'_\lambda = A \sin\left|\pi \times \frac{(x - x_c)}{\omega}\right| + \rho'_{\lambda,0} \qquad (1)$$

donde A es la amplitud de la función, x_c la ordenada al origen, ω el semiperíodo y $\rho'_{\lambda,0}$ el desplazamiento vertical de la función. Entonces, para determinar una función con la que aproximar los umbrales para cada banda de cualquier imagen, se ajustaron funciones como la representada por la Ecuación 1 a los puntos graficados en la Figura 2 y el valor de desplazamiento vertical $\rho'_{\lambda,0}$ se ajustó de manera tal que el 90% de los puntos quedaran dentro de la región contenida por los umbrales máximos y mínimos. En el gráfico de la Figura 2 también se muestra con líneas continuas los resultados de este ajuste en rojo para los umbrales máximos y en azul para los mínimos. Asimismo,

la Tabla 1 muestra los resultados numéricos de los parámetros de la Ecuación 1 para dichos ajustes.

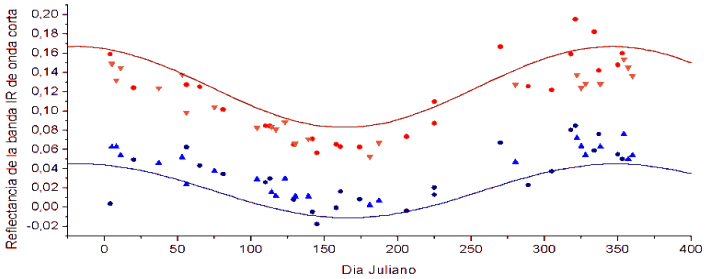


Figura. 2. Valores de umbrales máximo (rojo) y mínimo (azul) de reflectancia de la banda infrarrojo de onda corta para Landsat-7 (triángulos) y Landsat-8 (círculos).

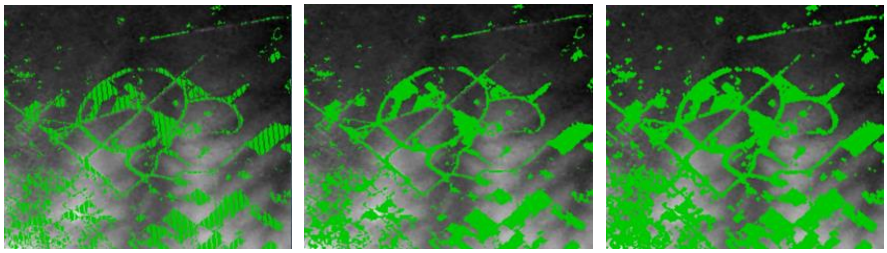
Tabla 1. Parámetros resultados de la ecuación 1 para cada función de umbralado

Umbral	A	xc	$\rho'\lambda_s 0$
Máximo IR onda corta	0.042	254.80	0.125
Mínimo IR onda corta	0.028	257.28	0.017
Máximo IR cercano	0.100	255.89	0.245
Mínimo IR cercano	0.073	263.67	0.075
Máximo Rojo	0.016	240.53	0.055
Mínimo Rojo	0.008	245.54	0.013
Máximo Verde	0.015	254.48	0.064
Mínimo Verde	0.012	255.58	0.023

2.3 Máscara de vegetación en el MDE

Para proyectar la máscara generada con las imágenes Landsat sobre el MDE SRTM es necesario transformar las coordenadas a UTM según la información provista en el *header*, y luego aplicar las ecuaciones de Coticchia Surace [16] para obtener las correspondientes coordenadas geográficas.

Como UTM es un sistema de proyección, las imágenes Landsat están en realidad deformadas por una transformación conforme. El resultado es que al proyectar la máscara sobre el MDE de SRTM se ven líneas verticales omitidas por la máscara debido a la menor resolución de las imágenes satelitales, como se aprecia en la Figura 3(a). La primera corrección que se le aplica consiste en rellenar los pixeles no marcados por la máscara que tienen a su lado pixeles que sí lo están. El resultado de esta corrección se muestra en la Figura 3(b). Además, como los bordes de las zonas arboladas son muy difusos y ocupan varios pixeles en ancho, se expande la máscara por dos pixeles en todas las direcciones. La máscara final de la vegetación en la zona de estudio 1 es la presentada en la Figura 3(c).



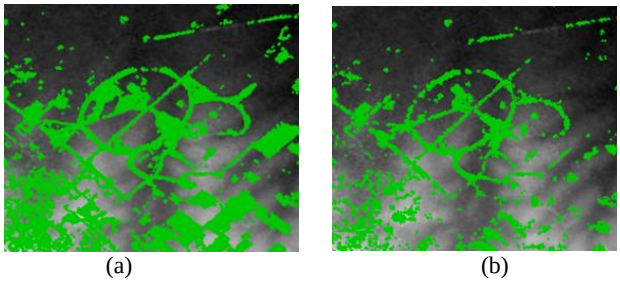
(a) (b) (c)

Figura. 3. Superposición de la máscara de vegetación sobre el MDE de la zona de estudio 1, y las diferentes etapas de corrección.

2.4 Corrección del MDE

En suelos de estas características, las derivadas segundas de la forma del suelo vegetado puede despreciarse, y en este caso es razonable suponer que la elevación del suelo vegetado es una función lineal cuya pendiente puede determinarse evaluando la diferencia de elevaciones a ambos lados de la región cubierta por la máscara. Por ende, el algoritmo de corrección que se propone aquí no depende de la altura de los árboles, que de por sí no es uniforme, sino que las elevaciones de las regiones identificadas por la máscara son descartadas para evaluarlas en función de la pendiente del suelo circundante. En particular, se toman en cuenta las pendientes en los sentidos Norte-Sur y Oeste-Este, definidas a partir de las elevaciones en los límites de las máscaras.

Gallant et *al*. [17] definieron criterios para definir si la altura de los árboles (que en este caso se toma como la diferencia en elevación entre un punto del MDE original y el corregido) es razonable. Entonces, la corrección en altura h provista por el algoritmo es aceptable si: $h > 3$ m (aproximadamente el menor valor asegurable de la varianza), $h < 25$ m, y la pendiente mediante la cual se calcula h es menor a 5 m cada 30 m. De este modo, se corrige la máscara eliminando los falsos positivos. Las Figuras 4 (a) y 4 (b) muestra la mejora en la máscara provista por este procedimiento.



(a) (b)

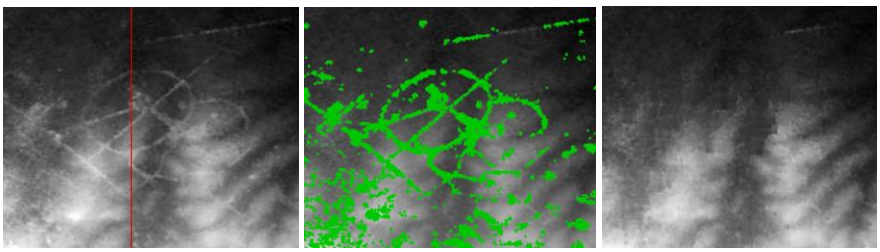
Figura 4. Máscara de vegetación antes (a) y después (b) de la mejora alcanzada aplicando los criterios de aceptación de Gallant et al. [17].

Para la corrección del MDE se desestiman las elevaciones en cada punto arbolado y se calculan a partir de los pixeles que los rodean. Además, se ponderan estas dos

pendientes según las inversas del cuadrado de la distancia al borde más cercano en cada dirección. En este caso, se aclara que el método funciona mejor para filas de árboles como son comunes en los campos argentinos pero no tanto para grandes áreas forestadas. Luego, para solventar las diferencias entre pixeles contiguos de los bordes debido al error del mismo MDE, se suaviza el resultado de cada pixel corregido promediándolo con la media de los pixeles a su alrededor.

3 Resultados

Con el fin de testear la aplicabilidad del algoritmo propuesto, se eligieron tres regiones rurales que adhieran a las características de suelo descritas. La primera, presentada en la Figura 1, está ubicada en las afueras de la ciudad de Tandil, provincia de Buenos Aires. Esta zona cuenta con leves pendientes debido a que se encuentra en la región serrana de la provincia. Presenta superficies arboladas con patrones muy característicos que facilitan la comparación entre imágenes y MDEs. La segunda es un área con relieve típico de llanura, se trata de un campo cercano a General Villegas al noroeste de la Provincia de Buenos Aires. Por último, se escogió un área dentro del estado de Indiana en Estados Unidos de América, de similar relieve y vegetación pero de donde se dispone además de los MDE LIDAR en formato *raster*. A continuación, en las Figuras 5, 7 y 9 se muestran los resultados de aplicar la segmentación de vegetación y la corrección final en cada zona respectivamente. También se comparan los correspondientes perfiles de elevación antes y después de la aplicación del algoritmo de corrección en las Figuras 6, 8 y 10.



(a) (b) (c)

Figura 5. Zona 1. MDE original (a), máscara (b) y MDE corregido (c)

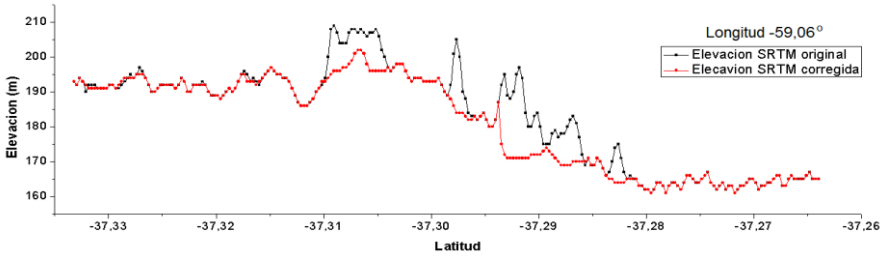
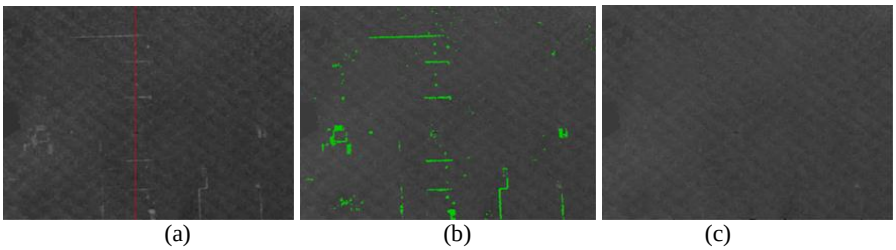


Figura 6. Perfiles de elevación de la Zona 1 para la transecta de la imagen 5ª



(a) (b) (c)

Figura 7. Zona 2. MDE original (a), máscara (b) y MDE corregido (c)

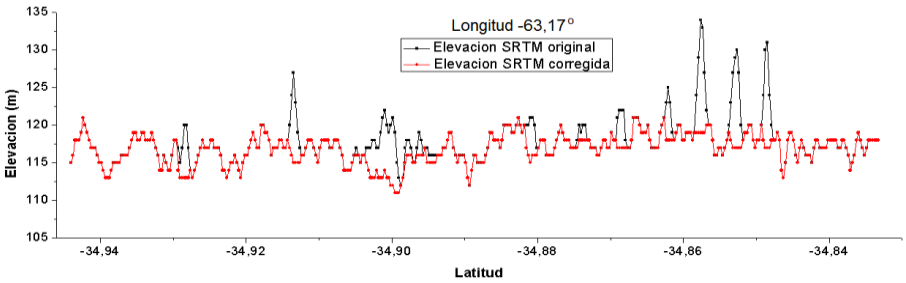
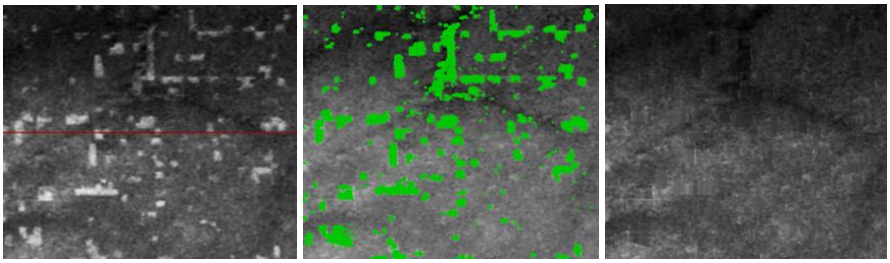


Figura 8. Perfiles de elevación de la Zona 2 para la transecta de la imagen 7a



(a) (b) (c)

Figura 9. Zona 3. MDE original (a), máscara (b) y MDE corregido (c)

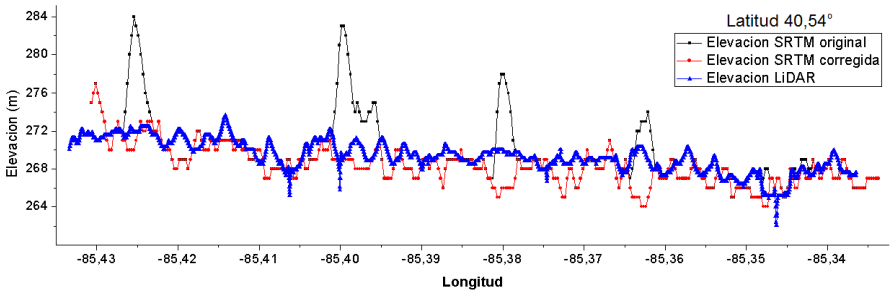


Figura 10. Perfiles de elevación de la Zona 3 para la transecta de la imagen 8a

En esta última zona se observa la similitud lograda entre el perfil del MDE corregido y el MDE de origen LIDAR (en color azul en la Figura 10) de mucha

mayor precisión. Puede verse claramente los picos sobresalientes del SRTM original, causados por la vegetación arbórea y su eliminación exitosa por el algoritmo propuesto en cada uno de los MDE elegidos.

4 Conclusión

En el presente trabajo se describe el desarrollo de una metodología compuesta por varios algoritmos para la corrección extensiva de *Modelos Digitales de Elevación* que reduce el error en las mediciones de SRTM producido por la presencia de árboles. El proceso está enfocado principalmente a la corrección de MDEs de zonas productivas de llanuras como la Provincia de Buenos Aires donde las hileras de árboles son vistas como barreras y generan el efecto de "dique" en los modelos hidrológicos.

En primer lugar, la identificación de la vegetación arbórea se realiza mediante imágenes Landsat definiendo umbrales para las reflectancias punto a punto de la radiación de cuatro bandas: verde, rojo, infrarrojo cercano e infrarrojo de onda corta. Los umbrales de estas bandas se definieron ajustando funciones senoidales a una serie de umbrales elegidos manualmente para una zona en diversos momentos del año.

La máscara luego es proyectada sobre el espacio del MDE, rellenada y expandida para cubrir las zonas forestadas completamente. Finalmente, se evalúa la corrección de altura descartando de la máscara los puntos para los cuales el nuevo valor de elevación no cumple criterios determinado. El algoritmo de corrección definitivo estima la elevación de puntos en un terreno a partir de su entorno suponiendo que el suelo en el punto de elevación desconocida sigue una pendiente constante en las direcciones Norte-Sur y Oeste-Este.

En todos los casos evaluados los valores con los que reemplazan las elevaciones afectadas por la presencia de árboles se vieron contenidos dentro del error propio de las mediciones de SRTM y resultan en MDE de formas visiblemente más naturales, con menor cantidad de picos y puntos sobresalientes. Estos resultados se aprecian en los perfiles de elevación.

5 Referencias

1. Robinson, N., Regetz, J., Guralnick, R.P.: EarthEnv-DEM90: a nearly-global, void-free, multi-scale smoothed, 90 m digital elevation model from fused ASTER and SRTM data. ISPRS Journal of Photogrammetry and Remote Sensing, 87, 57–67 (2014)
2. Hayakawa, Y. S., Oguchi, T., Lin, Z.: Comparison of New and Existing Global Digital Elevation Models: ASTER GDEM and SRTM3. *Geophysical Research Letters*, **35** (17). DOI: 10.1029/2008GL035036 (2008)
3. Sampson, C.C., Fewtrell, T.J., Duncan, A., Shaad, K., Horritt, M.S., Bates, P.D.: Use of terrestrial laser scanning data to drive decimetric resolution urban inundation models. *Journal of Advances in Water Resour*, **41**, 1–17 (2012)
4. Baugh, C.A., Bates, P.D., Schumann, G., Trigg, M.A.: SRTM vegetation removal and hydrodynamic modelling accuracy. *Water Resources Research*, **49** (9), 5276 – 5289 (2013)

5. Austin, A. A., Beacon, P.: Comparación de Modelos Digitales del Terreno realizados con diferentes técnicas: IGM vs. SRTM. *Trabajo Practico Especializado de Agrimensura*, Universidad Nacional de Córdoba, Argentina (2004)
6. Rinaldi, P. R., Dalponte, D. D., Venere, M. J., Clausse, A.: Cellular automata algorithm for simulation of surface flows in large plains. *Simulation Modelling Practice and Theory*, **15**, 315 – 327 (2007)
7. Schumann, G., Matgen, P., Hoffman, L., Hostache, R., Pappenberg, F., Pfister, L.: Deriving distributed roughness values from satellite radar data for flood inundation modelling. *Journal of Hydrology*, **344** (1–2), 96–111 (2007)
8. Ekstrand, S.: Landsat TM-based forest damage assessment: correction for topographic effects. *Photogrammetric Engineering and Remote Sensing*, **62** (2), 151 – 162 (1996)
9. Asner, G. P., Knapp, D. E., Balaji, A., Páez-Acosta, G.: Automated mapping of tropical deforestation and forest degradation. *Journal of Applied Remote Sensing*, **3** (1), 33543 – 33543 (2009)
10. Gavier-Pizarro, G. I., Kuemmerle, T., Hoyos, L. E., Stewart, S. I., Huebner, C. D., Keuler, N. S., Radeloff, V. C.: Monitoring the invasion of an exotic tree (Lingustrum lucidum) from 1983 to 2006 with Landsat TM/ETM+ satellite data and Support Vector Machines in Córdoba, Argentina. *Remote Sensing of Environgment*, **122**, 134 – 145 (2012)
11. Baldeck, C., Asner, G.: Improving remote species identification through efficient training data collection. *Remote Sensing*, **6**, 2682 – 2698 (2014)
12. Espinosa, A. J. M.: Procesamiento de imágenes satelitales multiespectrales. *Proyecto final de carrera, Ingeniería Informática, Universidad del País Vasco* (2014)
13. Conesa Garcia, C., Alvarez Rogel, Y., Martinez Guevara, J. B.: Medio ambiente, recursos y riesgos naturales: Análisis mediante Tecnología SIG y Teledetección. Volumen II. *Universidad de Murcia*, FG Graf, S. L., ISBN: 84-8371-488-4 (2006)
14. Chuvieco, E.: El factor temporal en teledetección: Evolución fenomenológica y análisis de cambios. *Revista de Teledetección*, **10**: 1–9 (1998)
15. Raponi, M., Wolfram, E., Pallottal, J., Piacentini, R. D., Quel, E.: Medición y modelización de la irradiancia espectral solar UV incidente sobre Buenos Aires, Argentina. Determinación de irradiancia eritémica. *Anales AFA* **16**, 283–286 (2004)
16. Coticchia, A, Surace, L.: Risoluzione di problemi geodetici con le minicalcolatrici elettroniche programmabili. *Bollettino di Geodesia e Scienze Affini*, **1**, 37 (1978)
17. Gallant, J. C., Read, A. M., Dowling, T. I.: Removal of tree offsets from SRTM and other digital surface models. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, **39** (4), 275 – 280 (2012)

Visualización 3D de Terrenos Multiresolución basada en Shader Model 5

Lucas Borrelli[1], María J. Abásolo[1,2]

[1] Universidad Nacional de La Plata, Facultad de Informática
calle 50 y 120, 1900 La Plata, Argentina
[2] Comisión de Investigaciones Científicas de la Pcia. de Bs.As.
febolucas@gmail.com, mjabasolo@lidi.info.unlp.edu.ar

Resumen. Este artículo presenta un nuevo algoritmo de visualización de terrenos multirresolución que emplea los últimos avances de la GPU. En su diseño se aplica un criterio de selección de nivel de detalle que considera la percepción que tendrá el usuario de la rugosidad particular de cada zona del terreno. Además, se incorporan las capacidades de teselado por hardware de la GPU correspondientes a *Shader Model 5*, y se implementa un mecanismo de *geomorphing* creando transiciones suavizadas entre distintos niveles de detalle. La técnica realiza una representación multirresolución eficiente y escalable. Los resultados obtenidos posibilitan su utilización en aplicaciones interactivas que requieren de la visualización de terrenos en tiempo real para aplicaciones 3D.

Keywords: Terrenos, Gráficos 3D, Multirresolución, GPU, Teselado.

1 Introducción

Los algoritmos relacionados con la representación y visualización de terrenos se encargan de administrar los datos de modelos de elevación para representar escenarios basados en terrenos, y son importantes en un gran número de aplicaciones, como ser simuladores de vuelo o de entrenamiento, sistemas de información geográfica, realidad virtual, videojuegos, etc.

La principal dificultad a resolver por los algoritmos de visualización de terrenos es obtener una visualización eficiente en tiempo real y lograr terrenos que luzcan realistas. Por esta razón dichos algoritmos utilizan el concepto de multirresolución para lograr una adaptación dinámica de la complejidad del modelo visualizado generando representaciones simplificadas o de menor nivel de detalle o LOD (*Level Of Detail*).

Los últimos avances en las capacidades de las unidades de procesamiento gráfico o GPU (*Graphics Processing Unit*) han introducido la posibilidad de realizar teselados de geometría directamente por hardware y con una mínima intervención por parte de la CPU. Estos avances son muy relevantes respecto de la multirresolución, ya que proponen el desarrollo de sistemas más flexibles y eficientes para la generación de representaciones simplificadas en tiempo real.

Este artículo presenta el diseño de un algoritmo de visualización 3D de terrenos multirresolución basado en la última características de la aceleración de gráficos por hardware, y se muestran los resultados de las pruebas realizadas sobre el mismo. El objetivo ha sido lograr un algoritmo eficiente en el uso de los recursos de procesamiento y con buenos resultados visuales.

El resto del artículo se organiza de la siguiente forma: la sección 2 presenta algunos de los algoritmos multirresolución más relevantes de la bibliografía, la sección 3 detalla los aspectos de diseño del algoritmo desarrollado, y la sección 4 muestra los resultados de las pruebas realizadas sobre el mismo. Finalmente la sección 5 presenta las conclusiones del caso.

2 Trabajos anteriores

Con el advenimiento de la GPU a comienzos de 2000, una serie de nuevos algoritmos de visualización de terrenos han sido desarrollados principalmente para aprovechar al máximo las capacidades de cómputo de ese hardware. A continuación se mencionan las principales características de los algoritmos más relevantes. Una lista exhaustiva incluyendo algoritmos previos a la GPU, puede consultarse en [7].

En primer lugar, el algoritmo *Geometrical Mipmapping* [2] modela el terreno mediante una grilla regular de vértices la cual es dividida en bloques adyacentes para permitir que regiones del terreno sean visualizadas en distinto detalle. Para esto, cada bloque es simplificado realizando un submuestreo regular formando una secuencia de mallas de resolución decreciente denominadas *geomipmaps*. Así un *geomipmap* de nivel 0 corresponde a la mayor resolución, el nivel 1 posee la mitad y así sucesivamente hasta alcanzar el nivel de menor detalle de sólo 2 triángulos.

Otro algoritmo pionero es *Chunked LOD* [9], el cual se basa en una estructura de tipo *quadtree*, es decir un árbol cuaternario utilizado para subdivisión espacial. En este trabajo cada nodo del *quadtree* posee su propia malla denominada *chunk* para representar la zona del terreno que abarca. La malla de cada *chunk* es construida en una etapa de preprocesamiento utilizando un esquema de triangulación no regular, la cual es cargada en memoria de GPU a medida que es necesaria su utilización.

En *Geometry Clipmaps* [6], el terreno es representado mediante un conjunto de grillas regulares anidadas que se centran alrededor de la posición 2D del punto de observación sobre el plano del terreno. Cada grilla posee un nivel de detalle diferente, disminuyendo la resolución hacia las grillas externas. Los datos de elevación que conforman las grillas son almacenados en memoria de la GPU, de modo que cuando el punto de observación se traslada, los vértices deben ser actualizados de manera que las grillas se mantengan siempre centradas respecto del observador.

Por último, el algoritmo *CDLOD* [8] se estructura en un *quadtree* de igual modo que *Chunked LOD*, pero a diferencia de éste utiliza mallados regulares. Este algoritmo renderiza el terreno desde un *heightmap* almacenado en GPU gracias a las características de acceso a texturas presentes a partir del *Shader Model 3* [3]. Además, en el mismo se incorpora una técnica simple para la selección de nivel de detalle basada en un vector de distancias que a su vez posibilita realizar una transición suave entre LODs en la que no es necesario evitar grietas o T-junctions.

3 Descripción del algoritmo

3.1 Representación del terreno multirresolución

Para representar el terreno se conforma una grilla de dimensión 2n+1 de lado. La grilla se compone de vértices separados a intervalos regulares. La información de altura se toma de un *heightmap*, es decir de una imagen en tonos de grises que codifica la intensidad de cada píxel como la altura en ese punto. Un árbol cuaternario o *quadtree* de bloques organiza dicha grilla. El *quadtree* se utiliza como una estructura de subdivisión 2D de la superficie del terreno como se ilustra en la figura 1.

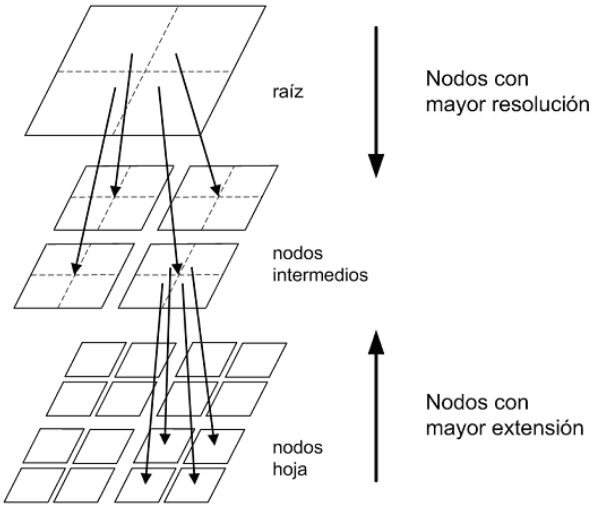


Fig. 1. Representación del terreno mediante *quadtree* de bloques.

La información de altura se modela en todos los nodos del *quadtree*. Así, los nodos del árbol representan una porción cuadrada del terreno a distinto nivel de detalle. El modelo multirresolución se logra aplicando distinto nivel de detalle según la altura de los nodos en el quadtree. La raíz abarca toda la superficie del mismo, mientras que el resto de los nodos abarcan un cuarto de la extensión que abarca su correspondiente nodo padre.

3.2 Selección de nivel de detalle

Algoritmos como [8] y [6] utilizan la distancia al punto de observación como único criterio de selección de LOD. Estos enfoques tienden a maximizar la percepción por parte del usuario de cambios geométricos. Para este trabajo se aplicó una métrica que

emplea el error de aproximación geométrica proyectado en espacio de imagen. Esto implica no sólo emplear la distancia como criterio de selección de detalle, sino que permite considerar la rugosidad propia de cada zona del terreno, asignando además mayor detalle a zonas más rugosas (picos o cimas) y menor detalle a zonas llanas.

El cálculo de la métrica consiste en dos etapas. Primero debe determinarse el error de aproximación geométrica δ para cada nodo del *quadtree*. Luego, el error δ es proyectado en espacio de imagen obteniéndose el error visual ε perceptible por usuario.

Como el error de aproximación geométrica δ es la diferencia de altura (en espacio de objeto) entre los vértices de dos niveles de detalle distintos [5], el cálculo de δ requiere el recorrido y análisis de las muestras del *heightmap*. El costo de este computo no permite ser llevado a cabo durante la visualización de tiempo real, por lo que se realiza en una etapa de preprocesamiento. Los δ_i obtenidos para cada nodo i del árbol son almenados en disco, y cargados posteriormente en memoria para su utilización.

Al momento de la visualización, un proceso de refinamiento recursivo es llevado a cabo para identificar el nivel de detalle para cada zona del terreno. Esto es, el *quadtree* es recorrido de forma descendente comenzando desde la raíz. Descendiendo por el mismo, se recolectan los nodos que satisfacen el nivel de detalle deseado. El error δ_i de cada nodo i visitado es entonces proyectado en espacio de imagen mediante proyección perspectiva como indica la ecuación 1: siendo h_{img} la altura en espacio de imagen del plano de proyección, d la distancia desde el punto de observación al bloque de terreno, y *fov* el ángulo del campo visual con el cual se realiza la proyección mediante la cámara virtual de la escena.

$$\varepsilon = \delta \frac{h_{img}}{2d\ tan\left(\frac{fov}{2}\right)} \quad \textbf{(1)}$$

Esta métrica asume que el punto de observación se encuentra sobre el plano horizontal y la dirección de visualización siempre es paralela al mismo. Esta simplificación del cálculo trae un beneficio adicional. Al ser independiente de la dirección de visualización, el nivel de detalle del terreno se mantiene estable cuando el usuario rota la cámara, variando sólo cuando el usuario traslada su posición, lo que permite minimizar la percepción de posibles cambios en la geometría.

Finalmente, ε es comparado contra un umbral de tolerancia τ configurable. Durante el recorrido del quadtree, si el error visual ε_i de un nodo i supera la tolerancia permitida τ, es necesario mayor nivel de detalle en esa zona del terreno por lo que el nodo debe ser refinado. En caso contrario se ha encontrado un nodo con un nivel de detalle aceptable para ser visualizado.

3.3 Renderización

La renderización se lleva a cabo utilizando las características de teselado de primitivas de *Shader Model 5* [10]. Una misma primitiva tipo *quad-patch* es utilizada

para generar en tiempo real las mallas poligonales correspondientes a la superficie del terreno para cada nodo seleccionado en el proceso de refinamiento y recorrido del *quadtree*.

Los vértices de cada malla son trasladados mediante *shaders*. Un desplazamiento vertical respecto del plano del terreno se realiza mediante el acceso a las muestras de altura almacenadas en un recurso de tipo textura que contiene el *heightmap*. Horizontalmente la posición de los vértices también es modificada según un valor configurable de separación entre muestras, y trasladados según la posición relativa de cada nodo en el *quadtree*. De esta manera puede aprovecharse al máximo todos los recursos de hardware, a la vez que se minimiza la intervención de la CPU en la renderización.

3.4 Prevención de grietas entre bloques

Un inconveniente a resolver por las técnicas basadas en bloques, es la disyunción de vértices entre bloques de distinto nivel de detalle. Estas discontinuidades o "grietas" pueden manifestarse por no emplear los mismos vértices a lo largo del borde que dos bloques comparten entre sí.

En [2] y [4], la solución propuesta ha sido realizar modificaciones a la topología del mallado mediante la reconexión de los vértices en el borde de uno de dos bloques de geometría adyacentes. Por otro lado, con la introducción de la última versión de *Shader Model* (SM5), hoy también es posible realizar la adaptación de bordes por GPU mediante los denominados factores de teselado externos. Sin embargo, esta técnica requiere conocer el nivel de detalle de los bloques vecinos de cada bloque. Esto implica realizar búsquedas adicionales por el *quadtree*, consumiendo ciclos adicionales de CPU y volviendo más complejo y menos performante el algoritmo de visualización.

En este trabajo se ha utilizado geometría adicional alrededor de cada bloque de terreno como en [9]. Esta geometría adicional es denomina “polleras” (*skirts*), ya que consisten en triángulos verticales que comienzan en los bordes de un bloque y se prolongan hacia abajo por una determinada extensión. Las polleras constituyen un método simple y efectivo de resolver la posible aparición de grietas, y son similares a los “triángulos de área cero” utilizados en [6]. Además, pueden utilizarse también primitivas *quad-patch* verticales para generar el mallado de las mismas, permitiendo una implementación limpia y por GPU al ser incluidas en el mismo proceso de renderización de bloques mediante *shaders*.

3.4 Geomorphing

Otro aspecto relacionado a la representación multirresolución de terrenos es el efecto denominado *popping*. Este consiste en la visualización de cambios repentinos de altura o *pops* en la superficie del terreno, que se producen al modificar el mallado de una zona, o al reemplazar un bloque de terreno por otro de distinto nivel de detalle. Con el objetivo de mejorar la performance a la vez de minimizar posibles apariciones de *pops*, se consideró aplicar la técnica denominada *geomorphing*, esto es, una

animación o transición suavizada entre las representaciones geométricas de dos niveles de detalle distintos.

Nuevamente, se han aprovechado las características de *Shader Model 5* para poder implementar esta técnica mediante el uso de los factores de teselado fraccionales. Su utilización ha permitido implementar transiciones suavizadas sin requerir el desarrollo de *shaders* específicos como en modelos de *shader* anteriores [11].

En la figura 2 puede observarse el mallado generado por el algoritmo utilizando distintos colores. La técnica de *geomorphing* es representada mediante la mezcla de los colores rojo y verde. El mallado de los bloques de terreno es coloreado hacia el rojo indicando un nodo a punto de subdividirse, y hacia el verde para indicar su colapso con nodos hermanos y reemplazo por el nodo padre.

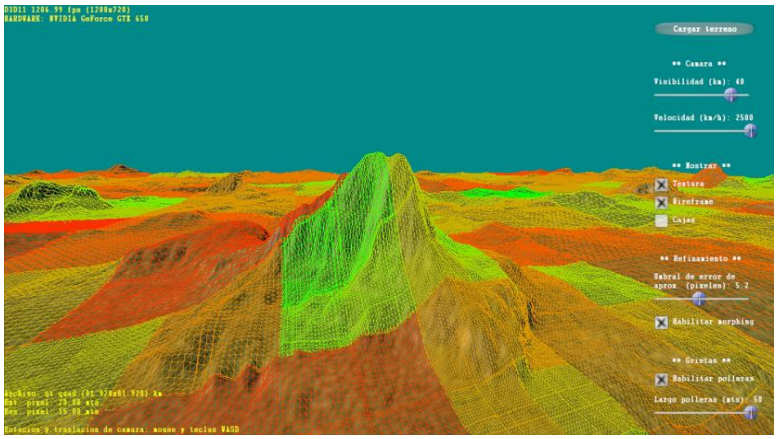


Fig. 2. Técnica de *geomorphing* representada mediante la mezcla de colores rojo y verde.

El *geomorphing* se realiza entre bloques de terreno que poseen niveles de detalle consecutivos, para lo cual se requiere conocer dos distancias para cada nodo del *quadtree*. En primer lugar es necesaria una distancia *s* que indica a qué distancia un nodo debería ser subdividido en sus cuatro nodos hijo de mayor detalle. A su vez, es necesaria la distancia *u* a la que un mismo nodo debe ser unificado junto a sus hermanos y reemplazados por el nodo padre.

$$f = \frac{d - s}{u - s} \tag{2}$$

La ecuación 2 (siendo *d* la distancia entre el centro del bloque y el punto de observación), es utilizada como peso para efectuar una interpolación lineal entre dos factores de teselado enteros. La relación *f* permite aplicar una transición suavizada entre dos niveles de detalle consecutivos, es decir entre un nodo del *quadtree* y sus nodos hijo. La técnica de *geomorphing* se logra aplicando un factor de teselado

fraccional (*Shader Model 5*) resultado de la interpolación. Cuando el punto de observación se traslada (variación de *d*), la transición entre el teselado del bloque actual y el teselado de los bloques de mayor detalle se realiza de forma incremental. Esto brinda una transición suavizada entre teselados para cualquier valor de *d*. Cuando *d* alcanza a *s*, el bloque actual es reemplazado por sus 4 bloques hijo de mayor detalle y el proceso de interpolación vuelve a comenzar sobre otro nivel del *quadtree*.

$$\tau = \varepsilon \quad \Rightarrow \quad \tau = \delta \frac{h_{img}}{2d\ tan\ (\frac{fov}{2})} \quad \Rightarrow \quad d = \delta \frac{h_{img}}{2\tau\ tan\ (\frac{fov}{2})} = s \tag{3}$$

Por último, es necesario calcular las distancias *s* y *u*. En primer lugar, como se ha mencionado la distancia *s* de un nodo es la distancia a la que el nodo es subdividido en sus cuatro hijos, y esa es la distancia a la que ε se equipara con τ, como se indica en la ecuación 3. Por otro lado, dado que la distancia a la que un nodo se divide es la misma a la que sus hijos se unifican, la distancia *u* es igual a la distancia *s* del padre del nodo actual. Así en el recorrido recursivo del quadtree, sólo es necesario enviar por parámetro la distancia *s* hacia los nodos hijo cuando un nodo es subdividido. Por último, la raíz carece de distancia *u*, la cual es considerada como 2*s*.

4 Resultados obtenidos

Para caracterizar el comportamiento del algoritmo, una serie de tests fueron realizados utilizando dos conjuntos de datos de zonas naturales conocidos como *Puget Sound* y *Hawaii*, y un tercer terreno *ad hoc* generado artificialmente, todos de 4097x4097 muestras de lado con separación de 10 metros. Los mapas de altura o *heightmaps* corresponden a imágenes PNG en escala de grises de 8 bits con una resolución vertical de 12.8 metros. Se obtuvieron terrenos de 40.96 x 40.96 km^2 de área, generando cada uno un mallado con resolución total de 33.6 millones de triángulos.

Para comparar performance respecto de la calidad de imagen obtenida, los tests fueron repetidos en cada terreno modificando el umbral de error visual τ = {1.0, 2.5, 5.0, 7.5, 10.0}. Los resultados se resumen en las tabla 1, 2 y 3, para *Puget Sound*, *Hawaii* y el terreno artificial respectivamente. Los valores de duración por cuadro se se estandarizaron en unidades de cuadros por segundo o FPS (*Frames Per Second*).

La implementación fue realizada utilizando la librería gráfica DirectX 11 [12]. El hardware utilizado en los tests fue una PC con procesador Intel® Core™ i7-3770 de 3.40GHz, memoria RAM de 8GB DDR3, y una placa gráfica NVIDIA GeForce GTX 650 con 2GB DDR5 de memoria de video. La resolución de pantalla fue de 1920x1080 píxeles (HD) y en pantalla completa.

Tabla 1. Resultados obtenidos con *Puget Sound*.

Umbral	#bloques (#tris) / seg.	FPS Mínimo	FPS Media	FPS Máximo
1.0	145K (672M)	183 (-29%)	257	363 (+41%)
2.5	108K (523M)	482 (-37%)	762	1106 (+45%)
5.0	61K (315M)	1057 (-21%)	1343	1599 (+19%)
7.5	40K (201M)	1356 (-17%)	1634	1878 (+15%)
10.0	28K (140M)	1598 (-12%)	1810	2000 (+10%)

Tabla 2. Resultados obtenidos con *Hawaii*.

Umbral	#bloques (#tris) / seg.	FPS Mínimo	FPS Media	FPS Máximo
1.0	139K (688M)	269 (-20%)	337	544 (+61%)
2.5	107K (535M)	634 (-16%)	756	1058 (+40%)
5.0	72K (346M)	1047 (-19%)	1296	1618 (+25%)
7.5	44K (225M)	1308 (-19%)	1608	2119 (+32%)
10.0	30K (164M)	1472 (-17%)	1779	2339 (+31%)

Tabla 3. Resultados obtenidos con el terreno artificial.

Umbral	#bloques (#tris) / seg.	FPS Mínimo	FPS Media	FPS Máximo
1.0	140K (694M)	201 (-6%)	214	227 (+6%)
2.5	124K (618M)	457 (-8%)	495	538 (+9%)
5.0	96K (481M)	872 (-7%)	934	1022 (+9%)
7.5	73K (367M)	1130 (-10%)	1249	1308 (+5%)
10.0	56K (284M)	1355 (-6%)	1449	1540 (+6%)

El resultado, es una variación de performance debido a la cantidad de geometría utilizada para mantener la calidad seleccionada. En todos los casos se observa que a medida que se relaja el umbral de error, es necesario un menor número de bloques para aproximar la calidad seleccionada, y por tanto el número de cuadros que es posible generar por segundo aumenta respectivamente. Además la media de FPS no registra grandes variaciones respecto de los extremos mínimo y máximo en cada test. Los tests que registran menor variación corresponden al terreno artificial debido principalmente a la alta frecuencia de zonas rugosas de su topología. Por la misma razón, también registra menores medias respecto de los otros terrenos.

Los mejores resultados son aquellos que logran un balance entre calidad visual y performance. Partiendo de un umbral de 1.0 y hasta 2.5 se registran mejoras en la performance y sin pérdida de calidad visual aparente. A partir de valores de 5.0 y mayores la calidad visual puede no ser muy apropiada. Un valor de umbral en el

rango 2.5 a 5.0 permite lograr el mejor balance entre calidad visual y performance. En este rango se han obtenido performances mínimas entre 500 y 1000 FPS. Sin considerar otros aspectos de las aplicación de visualización de terrenos (trabajo en red, simulaciones físicas, etc.), los resultados obtenidos superan ampliamente los 60 FPS necesarios para obtener una secuencia de imágenes fluida y de buena interactividad [1]. La performance lograda es muy satisfactoria.

5 Conclusión

El algoritmo de visualización 3D de terrenos multirresolución presentado se basa en un *quadtree*. Esta estructura regular ha permitido aprovechar las últimas características de la GPU (SM5), para realizar una representación multirresolución eficiente y escalable. Mediante las funcionalidades de teselado, fue posible controlar la resolución por hardware, facilitando a su vez la implementación de otras características como por ejemplo la técnica de *geomorphing*.

En su diseño se abordaron los inconvenientes provenientes de considerar como único criterio de selección de nivel de detalle la distancia al punto de observación. Se incorporó entonces un criterio más apropiado que considera la percepción que el usuario tendrá de las particularidades de cada zona del terreno. De esta manera es posible tener menos detalle en zonas llanas y tener mayor detalle en las zonas que poseen mayor rugosidad, obteniéndose una mejor distribución de triángulos para aproximar la superficie del terreno, logrando así mayor calidad visual disminuyendo la percepción de cambios de resolución.

Ajustando un parámetro de umbral de error, el algoritmo permite controlar la calidad visual/performance dependiendo de la necesidad o aplicación particular donde se desee implementar el algoritmo. Los tests de performance dieron como resultado un muy buen rendimiento de la técnica de visualización. El bajo consumo de recursos de procesamiento de CPU y GPU posibilita su utilización en aplicaciones interactivas que requieran de la visualización en tiempo real de modelos de terrenos para escenarios virtuales.

Referencias

1. Akenine-Möller, T., Haines, E., Hoffman, N.: Real-Time Rendering. 3ra edición, A.K.Peters Ltd. (2008)
2. De Boer, W.H.: Fast Terrain Rendering Using Geometrical MipMapping. World Wide Web, Octubre de 2000. Disponible en http://www.flipcode.com/archives/ article_geomipmaps.pdf.
3. Gerasimov, P., Fernando, R., Green, S.: Shader Model 3.0: Using Vertex Textures. NVIDIA Corporation Whitepaper (2004). Disponible en ftp://download.nvidia.com/developer/ Papers/2004/Vertex_Textures/Vertex_Textures.pdf.
4. Larsen, B.D., Christensen, N.J.: Real-time Terrain Rendering using Smooth Hardware Optimized Level of Detail. Journal of WSCG, 11(2), pp. 282–9 (2003)
5. Lindstrom, P., Koller, D., Ribarsky, W., Hodges, L.F., Faust, N., Turner, G.A.: Real-Time, Continuous Level of Detail Rendering of Height Fields. Proceedings ACM SIGGRAPH ’96 Conference on Computer Graphics, pp. 109–118 (1996)

6. Losasso, F., Hoppe, H.: Geometry Clipmaps: Terrain Rendering Using Nested Regular Grids. ACM Transactions on Graphics (SIGGRAPH), 23(3), pp. 769-776 (2004)
7. Pajarola, R., Gobbetti, E.: Survey on Semi-Regular Multiresolution Models for Interactive Terrain Rendering. The Visual Computer 23 (2007)
8. Strugar, F.: Continuous Distance-Dependent Level of Detail for Rendering Heightmaps. Journal of Graphics, GPU, and Game Tools, 14(4), pp. 57-74 (2009)
9. Ulrich, T.: Rendering Massive Terrains using Chunked Level of Detail Control. Course Notes of ACM SIGGRAPH '02, ACM Press (2002)
10. Valdetaro, A., Nunes, G., Raposo, A., Feijó, B.: Understanding Shader Model 5.0 with DirectX 11. Proceedings do SBGames (2010)
11. Wagner, D.: Terrain Geomorphing in the Vertex Shader. En Wolfgang E. Shader-X 2: Shader Programming Tips & Tricks With Directx 9. Wordware Publishing (2004)
12. Zink, J., Pettineo, M., Hoxley, J.: Practical Rendering and Computation with Direct3D 11. A.K.Peters Ltd. CRC Press (2011)

A Serious Game about Recycling Rules

N. Jofré, G. Rodriguez, Y. Alvarado, J. Fernández, and R. Guerrero

Laboratorio de Computación Gráfica (LCG) - Universidad Nacional de San Luis,
Ejército de los Andes 950
Tel: 02664 420823, San Luis, Argentina
{npasinetti,gbrodriguez,ymalvarado,jmfer,rag}@unsl.edu.ar

Abstract. Nowadays serious games is one of the biggest existing industries and it is still growing steadily in many sectors. As a major subset of serious games, designing and developing virtual reality applications to support education or promote social behavior has become a promising frontier, because games technology is inexpensive, widely available, fun and entertaining for people of all ages, with several health conditions and different sensory, motor, and cognitive capabilities.

In this paper, we provide an overview about a serious game with a perspective of virtual reality for social behavior. The work uses a serious game in an immersive learning environment for recycling learning. In order to improve the user experience the game was developed to work in a cave-like immersive environment, with natural interaction selective alternative.

The game includes static and dynamic 3D environments, allowing to share the experience of scenario navigation among users, even geographically distributed.

Keywords: Serious Games, Virtual Reality (VR), CAVE, Computer Graphics.

1 Introduction

For some time now, virtual reality has allowed the generation of interaction environments that facilitate new contexts of exchange and communication of information. More specifically, the employment of Virtual Reality is a natural idea to improve the impression of living in a simulated reality, so this tool is largely used in many areas such as medicine, industries, education and entertainment [1].

In entertainment industry, in particular, the creation of computer games using different technologies, rules and goals among others, has grown considerably. Today, playing computer games has become a popular activity for people of different cultures and ages. This habit motivated game developers, educators and domain experts to create other kind of applications for computer game technologies [2, 3].

These new applications which aim to address a specific problem or to teach a certain skill are called *Serious Games* and mainly relates to interactive computer-based game software that intentionally produces games outside of entertainment, i.e games with serious purposes [3].

A serious game is designed based on different educational, training, informational and learning motivations. Specifically, game-based learning involving educational, cognitive and affective aspects induces learners/gamers to higher motivation and enhance their learning success. While there are inherent tensions between contemporary youth culture and traditional education, researches show that new learning game developments promise to help shortcutting the bridge of that growing generational divide. Besides as another positive effect of learning games is to allow the learning of knowledge and problem solving skills with better performance and long-lasting attributes [4]. At the moment, serious games have allowed to solve lot of problems from technological, medical, educational and environmental areas.

One of current problems in the world is the increasing recognition of the need to sustain an ecologically-balanced environment. A helping action to this problem is to reduce and avoid negative impacts of waste on the environment, being the diversion of biodegradable waste from landfills an important contribution to limiting greenhouse as emissions. In this context, serious games of learning using virtual reality are tools that add entertainment to teaching and training for adoption of sustainable development practices [5, 6].

Adopting these practices begins on bringing about behavior change. The designing process to enhance behaviors strongly dependent of intention, i.e the commitment to a certain action. Many times are deliberate acts based on the beliefs of the individual and the norms imposed by society [7]. When an individual is positively predisposed toward a particular behavior, and additionally perceives support for this from people around them, then it will form a positive behavioral intention towards that behavior [8]. Such collective behavior is needed on issues such as recycling, i.e a model of social behavior to enable this kind of "contagion" in social and sustainable problems such as waste recycling. In business world there is a new web-based model that harnesses the creative solutions of a distributed network of individuals through what amounts to an open call for proposals. This model is called *crowdsourcing* and its name is formed from two words, *crowd*, making reference to the people who participate in the initiatives, and the word *sourcing*, which refers to a number of procurement practices aimed at finding, evaluating, and engaging suppliers of goods and services. Literally, *crowdsourcing* means to outsource an activity to the crowd and for that it quickly began to be used in other areas such as entertainment, sociology, psychology and others [9, 10].

Particularly, this property of outsourcing it to an undefined (and generally large) network of people in the form of an open call allow to developers and researchers of serious games to use it in games that allow solve problems either collectively or competitively [11].

This work presents *Recycle Now!*, a serious-game-based virtual reality for enhancing recycling behaviors and environment awareness. Essentially, the idea is to develop a game for motivate and teach the basic principles of recycling and training about different types of recycling using the crowdsourcing concept to create a collective behavior.

2 The developed Serious Game concepts

Currently, even though in world there are many campaigns on recycling (*Wecycle*, *Plastic Hero*, among others) still exists a lack of awareness among people. Some people raise than this activity needs an extra effort since separating the garbage of their homes and putting it on their home's trash container it is a hard work, besides they don't want to dedicate more place to garbage in their homes. Certainly there are major problems like climate change, lack of protection of wildlife, landfills among others, which require the support of the whole society. Therefore people should adopt recycling good habits firstly at home and then apply them in public places [12, 13].

Finding a way to make from recycling a daily activity is not easy but as it was mentioned, serious games are an interesting tool to making recycling a fun and natural activity. Thus, a person could learn at his home the recycling's basic principles and then out into the world and unconsciously apply it. This behavior implies analyze several theoretical aspects which will be used for the development of this game.

2.1 Learning

Some time ago, new technologies have been incorporated to education as learning tools, particularly, people are finding ease increasingly in learning games environments. This game is presented as a game where the user can collect, identify and place trash. At the same time, other players may be doing the same task and also correct each other in order to earn more points. Apparently, the game can be categorized as competitive, but the learning is not; it is expected that at the end of every play each player ends up learning a little more about recycling practices [14].

Therefore, this game is based in a learning method called *Cooperative Learning*; which is an instructional approach in which learners work together in small groups to achieve shared learning goals. This approach invites group members to reach outcomes by setting and working towards a common goal, putting emphasis on cooperative evaluation of these outcomes. While learners are all on equal footing, great emphasis is placed on the responsibility of individuals [15]. Accordingly, the players of this serious game have one goal in common (recycle) and also each is responsible for the moves they perform. Finally, players are somehow cooperating with each other to adopt good practices for recycling.

2.2 Crowdsourcing

Crowdsourcing is evolving as a distributed problem-solving and business production model in recent years. In crowdsourcing paradigm, tasks are distributed to networked people to be completed such that cost and time can be greatly reduced.

Nowadays, many tasks that are trivial for humans continue to challenge even the most sophisticated computer programs, such as image annotation. These tasks cannot be computerized [16]. Current research in crowdsourcing often focuses on micro-tasking, however, participants are people with rich capabilities including learning and collaboration, suggesting the need for more nuanced approaches that place special emphasis on participants. There are no recent studies using learning among these approaches, so crowdsourcing efforts based on learning through a game is a good objective [17, 18].

As it was mentioned the *Cooperative Learning* model allows users to achieve shared learning using interaction between them. Specific situations where the user will be benefited from this concept are related to game dynamics. Some examples where the user learns by interacting crowdsourcing are: when it is a witness of another player's mistake, it corrects another player, or it is corrected by another player on its own mistakes. Clearly, all learning situations depend on the existence of the cross-interaction via network.

Recycle Now! is a serious game that combine crowdsourcing mechanism for learning purposes. In the following subsection we describe a framework that supports the learning mechanism with the mentioned crowdsourcing concept.

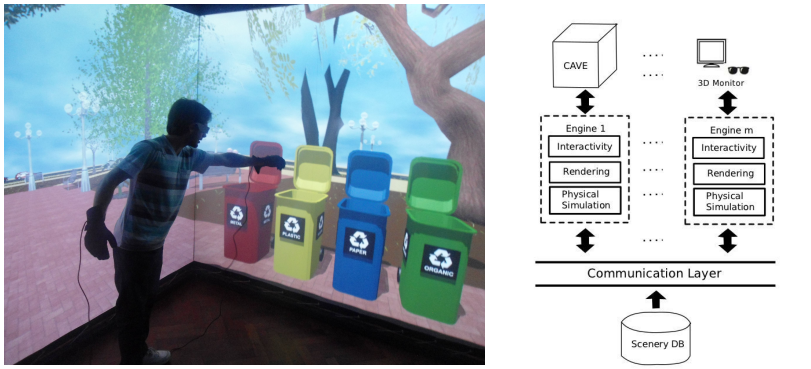
2.3 Platform

The game was developed to work on a computing platform for immersive collaborative 3D virtual world visualization (See Figure 1 (a)). It allows the use of geographically distributed VR media, called a multi-VRmedia. Remote players can interact into a 3D scenario through different multi-VRmedia. During navigation, players can exchange information in order to cooperatively solve the observed problems [19].

Each multi-VRmedia comprises the hardware and software necessary to gather the information obtained during interaction between user and game: via keyboard, mouse, data gloves, sound system, 3D active glasses, screen/projection surfaces, projectors, among others.

A computing platform includes a hardware architecture and a software framework (including application frameworks), where the combination allows software, particularly application software, to run. Typical platforms include a computer's architecture, operating system, programming languages and related user interface (run-time system libraries or graphical user interface). A system to visualize scenarios in a multi-virtual reality media environment has been defined. Such system will provide the necessary structure for attributes definition, rendering and collaborative multi-visualizations, as well as the needed interactive resources. Figure 1 (b) shows an overview of the work.

The system uses a client/server architecture similarly to a traditional network game, this allows user's interaction with others gamers distributed geographically.



(a) Immersive environment (b) Computing Platform

Fig. 1. Virtual Reality System.

3 GamePlay

According to Prensky, *"Gameplay is all the activities and strategies game designers employ to get and keep the player engaged and motivated to complete each level and an entire game"* [20]. This serious game offers a good gameplay combining a familiar and innovative **game design** and an educational and motivating **players experience**.

3.1 Game Design

Serious games must fulfill all the necessary and sufficient conditions to become a game. There are several aspects used in design of traditional games which pretends provide a good gaming experience to players. The most important aspects of the game are explained below.

First-Person game. Player perspective is one of the important design choices made when creating a digital game. Traditional camera options include audience, isometric, birds eye, trailing camera, third person and first person [21]. These views support distinctive experiences of immersion for video game play and different perception of the game space. First person (FP) allows the player to perceive the game through the eyes of the character, observing the world around them up close, giving a clear view of the scenario in front of them. As mentioned, this game is a game-based virtual reality and one of the pillars of VR is immersion. Therefore this game features a FP view to increase the feeling

of being immersed inside game, for example, when the player needs to put waste in the correct place using haptic devices, requires a vision close to the container. Thus this FP feature stablishes a "player-character" relationship to provide the most immersive feel for the player and improving learning abilities stimulating his visual and auditory capabilities.

Multiplayer game. In games world, most players want to share the same experience with other players, i.e seeing and feeling like they are playing the same game (being connected). A game meeting this feature is said to be a multiplayer game and *Recycle Now!* is not the exception [22]. The mainly reason making this multiplayer game is because the game's theme required the use of crowdsourcing' s concept, namely allowing another players to give a solution to an specific problem of the game. The mentioned game platform was built around a client-server architecture where each client connects to a single server resulting in the illusion of a shared experience but really each player is playing a separate game, each with its own game state. This feature allows meet an expected functionality for players and moreover a collective solution.

Affordance. Game engine has increasingly developed to include aspects like emotion, joy of use, user experience, or motivation. Therefore a concept has been sporadically applied to games for several years, this concept is known as "*affordance*" and refers to perception mapping what the external world affords the perceiver [23]. However games researchers try to explain how people discover the functionality of features in game applications. Particulary, virtual reality games developers have focused primarily on what players perceive they can do, as opposed to what players can actually do in an interactive virtual environment [24].

In this case in addition to offering a game with a serious purpose we wanted to give players a way to play to help their perception, granting them different means to traditional desktop game such as mouse and keyboard. It is for this reason that the game platform built allows some players to make use of tools such as 3D glasses, data gloves, body sensors among others, and so to increase its capacity and reduce perceptual cognitive effort. This way affordance is a powerful tool for understanding the relationship between player and system.

Environment. To achieve the proposed goals in the game, a scenario was developed as environment. The game was situated at the central square of a city, where the user can navigate for the square and streets around of it.

Stage was set with ambient sounds, inanimated objects (benches, lights, trees, garbage, trash cans) and animated objects (people) allowing social interaction, making the obtained learning through the game were similar to real world. Particularly, people on stage are avatars (See Figure 2).

Physical realism. Making an object looks real to the user more than visual realism is required. In a realistic game, objects also should behave as in reality

Fig. 2. Scenario.

including their physical characteristics, so it is necessary to simulate aspects like gravity and collision avoiding crossing between solid objects, achieving then a behavior like the impact of a ball on the ground and soft objects deformations, among others.

In addition user's movement basic physics, the game requires to perform collision detection for activities such as grasping and releasing waste. *Collision Shapes* are used for that like envelopes that allow sensing the world surrounding the object and making possible to visualize collisions between objects.

Real time. The virtual reality enables users to simultaneously experience real-time and interactive simulations. Real-time factors considered here are diverse: visualization, realistic audio, media interaction and user response times.

As was described in platform's section, real time is enhanced by haptics devices and others virtual reality devices. Because grasping and releasing waste is a key type interaction, a real-time hand gesture interface to manipulate objects in the game has been implemented. As an example, a user may see a simulated virtual representation of themselves (an Avatar) or a part of themselves (hands) that reflects real-time movements (e.g., lifting a finger, close the hand, etc.).

The developed game has the ability to interact with users in real time and receive feedback. The game allows users from around the world communicating, playing, learning and networking in real time.

3.2 Player's Experience

A good gaming experience involves keeping players motivated. To achieve this motivation both the game and opposing players must be a challenge for all players. Regarding players, an experienced player will not get the expected challenge

if it is playing with beginners. These possible differences in experiences must be balanced before starting to play. Consequently *Recycle Now!* provides different tools and documentation to help players who need it and reports on all relevant regulations about game.

Rules. Before starting the game, players must join to the same play, one player will be the server that created the play and others are guess. Initially each player has a set of different garbage containers labeled: *Organic*, *Plastic*, *Metal* and *Paper*, and they must decide where will place each of them on stage. To achieve a uniform distribution, players are prohibited placing containers near each other (See Figure 3). Once containers of all players are located, they can not be relocated.

Fig. 3: *Garbage Containers Distribution*

After it, game starts. Players must find the trash, collect, analyze and throw it in their containers. Players compete against each other for garbage pick up quickly and increase their scores. After a certain time, the game ends and the player with the highest score wins the game. All players can see the scores of other players while they are playing.

Skills. The game allows to see the score of players, showing who is the winning player so far. It also offers players the opportunity to discredit other players using an ability called "*Let me check!*". All players can execute this skill a certain number of times as they see fit.

When a disbelieving player (player A) wants to discredit a refuted player (player B), player A must execute "*Let me check!*" action. With this ability, player A can adjust player B garbage classification, and consequently player A score, increasing it for each right classification, and decrementing it for every wrong classification. The score of player B will be decremented only for each player A correct classification (See Figure 4).

Fig. 4: *Let me check*

After checking up Player B collected garbage, both players return to play and

collect garbage.

4 Conclusions and Future Works

New technology has great potential to benefit education. From this example, it should be clear how important games can be for stimulating rapid re-mixing for educational examples.

This paper involved the development of a multiplayer serious game with several components: virtual reality, learning, crowdsourcing and affordance, among others. We described a serious game that was designed to motivate players to recycle properly so they can use it on his daily life at real spaces with waste sorting such as squares, public building, etc. The game was designed to be played by any player even without experience with games giving an interactive visualization experience and multi-RV interaction.

In this work we had focused on crowdsourcing learning and how to do it more natural and intuitive to users allowing proficient user-interactivity in real-time, meaningful feedback and learning through an interface.

The effectiveness of the game as a pedagogical vehicle can be highlighted by some influencing factors: social interaction, immersion, ambient noise level, among others [25]. By the moment, some game's aspects are beta version.

Future works will be oriented to improve game engine annoyances. Some environment issues, such as hardware platform, limited the scenario's resolution of the game. This constraint often forced a trade-off in the amount and nature of special effects and the number of textures used to create the game environment.

Acknowledgments. This work is supported by the European Community, Alfa III - GAVIOTA, Contract N: EuropeAid/129-877/C/ACT/RAL-1.

References

1. Williamson Ben. Computer games, schools and young people, 2009.
2. Minhua Ma, Andreas Oikonomou, and Lakhmi C. Jain. *Serious Games and Edutainment Applications*. Springer Publishing Company, Incorporated, 2014.
3. Y. Cai and S.L. Goei. *Simulations, Serious Games and Their Applications*. Gaming Media and Social Effects. Springer, 2013.
4. Ashok Ranchhod, Clin Guru, Euripides Loukis, and Rohit Trivedi. Evaluating the educational effectiveness of simulation games: A value generation model. *Inf. Sci.*, 264:75–90, April 2014.
5. K. Katsaliaki and N. Mustafee. A survey of serious games on sustainable development. In *Simulation Conference (WSC), Proceedings of the 2012 Winter*, pages 1–13, Dec 2012.
6. Pascal Lessel, Maximilian Altmeyer, and Antonio Krüger. Analysis of recycling capabilities of individuals and crowds to encourage and educate people to separate their garbage playfully. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 1095–1104, New York, NY, USA, 2015. ACM.

7. I. Ajzen and M. Fishbein. *Understanding attitudes and predicting social behaviour*. Prentice-Hall, Englewood Cliffs, New Jersey, 1980.
8. Michele Tonglet, Paul S Phillips, and Adam D Read. Using the theory of planned behaviour to investigate the determinants of recycling behaviour: a case study from brixworth, uk. *Resources, Conservation and Recycling*, 41(3):191–214, June 2004.
9. Daren C. Brabham. Crowdsourcing as a Model for Problem Solving. *Convergence: The International Journal of Research into New Media Technologies*, 14(1):75–90, February 2008.
10. Enrique Estellés-Arolas and Fernando González-Ladrón-De-Guevara. Towards an integrated crowdsourcing definition. *J. Inf. Sci.*, 38(2):189–200, April 2012.
11. Sigal Sina, Avi Rosenfeld, and Sarit Kraus. Generating content for scenario-based seriousgames using crowdsourcing. In *In Proceedings of the 28th AAAI Conference on Artificial Intelligence*. AAAI Press, 2014.
12. M. Martin, I.D. Williams, and M. Clark. Social, cultural and structural influences on household waste recycling: A case study. *Resources, Conservation and Recycling*, 48(4):357–395, 2006.
13. W. Kip Viscusi, Joel Huber, Jason Bell, and Caroline Cecot. Discontinuous behavioral responses to recycling laws and plastic water bottle deposits. Working Paper 15585, National Bureau of Economic Research, December 2009.
14. J. McGonigal. *Reality Is Broken: Why Games Make Us Better and How They Can Change the World*. Penguin Publishing Group, 2011.
15. D.F. Salisbury. *Five Technologies for Educational Change: Systems Thinking, Systems Design, Quality Science, Change Management, Instructional Technology*. Educational Technology Publications, 1996.
16. Man-Ching Yuen, I. King, and Kwong-Sak Leung. A survey of crowdsourcing systems. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 766–773, Oct 2011.
17. M.J.-Y. Chung, M. Forbes, M. Cakmak, and R.P.N. Rao. Accelerating imitation learning through crowdsourcing. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 4777–4784, May 2014.
18. Jean-Claude Bradley, RobertJ Lancashire, AndrewSID Lang, and AntonyJ Williams. The spectral game: leveraging open data and crowdsourcing for education. *Journal of Cheminformatics*, 1(1), 2009.
19. Y. Alvarado, N. Moyano, D. Quiroga, J. Fernández, and R. Guerrero. *Augmented Virtual Realities for Social Developments. Experiences between Europe and Latin America*, chapter A Virtual Reality Computing Platform for Real Time 3D Visualization, pages 214–231. Universidad de Belgrano, 2014.
20. Marc Prensky. The Motivation of Gameplay: or, the REAL 21st century learning revolution. *On the Horizon*, 10(1), 2002.
21. Francois Dominic Laramee. *Game Design Perspectives*. Charles River Media, Inc., Rockland, MA, USA, 2002.
22. Yanna Vogiazou. Presence based massively multiplayer games: Exploration of a new concept. 2002.
23. Gibson J J. *"The theory of affordances," in Perceiving, Acting, and Knowing. Towards an Ecological Psychology.* Number eds Shaw R., Bransford J. Hoboken, NJ: John Wiley & Sons Inc., 1977.
24. B. Dalgarno and M. J. W. Lee. What are the learning affordances of 3-d virtual environments? *British Journal of Educational Technology*, 41(1):10–32, 2010.
25. B.P. Bergeron. *Developing Serious Games*. Charles River Media game development series. Charles River Media, 2006.

XIII Workshop Tecnología Informática Aplicada en Educación (WTIAE)

7418 | Using Social Network Analysis in the classroom: a case study applying NodeXL
Antonieta Kuz (UTN FRLP), Mariana Falco (UTN FRLP), Enzo Renzis (UTN FRLP), Nahuel Leopoldo (UTN FRLP), Roxana S. Giandini (UNLP)

7484 | Mejoramiento de la Adaptación usando la Lógica Difusa
Constanza Huapaya (UNMdP), Leonel Guccione (UTN-MDP), Delia Esther Benchoff (UNMdP), Marcela Gonzalez (UNMdP), Francisco A. Lizarralde (UNMdP)

7490 | Pautas para la actualización de la Plataforma Moodle entre versiones distantes. Una experiencia concreta
María Fernanda Guzmán (UNT), Pablo Rodríguez Rey (UNT), Lía Fabiana Torres Auad (UNT), Nicolás Auvieux (UNT), Julio Escalante Figueroa (UNT), Ticiano Torres Peralta (UNT)

7494 | Get Closer to Activate It! An Educational Tool for People with Multiple Disabilities
Filip Bojanovic (UIB), Cristina Manresa-Yee (UIB), María José Abásolo (UNLP), Martín Larrea (UNS), Cecilia Sanz (UNLP)

7428 | Modelo Bayesiano para el Diagnóstico del Aprendizaje en Alumnos de Inteligencia Artificial
Pablo Pytel (UTN-FRBA), Cinthia Vegega (UTN-FRBA), A Deroche (UTN-FRBA), Mariana Acosta (UTN-FRBA), Ma Florencia Pollo Cattaneo (UTN-FRBA)

7519 | Integración de Indicadores Sociométricos de Dinámicas Grupales en Ambientes Virtuales de Trabajo Colaborativo Orientados a Intervenciones Didácticas
Norberto Charczuk (UNLA), Darío Rodríguez (UNLA), Ramon Garcia Martinez (UNLA)

7535 | Propuesta de Virtualización Móvil en la Facultad de Ingeniería de la UNLPam
Jose Luis Filippi (UNLPAM), Carlos Ballesteros (UNLPAM), Rodolfo Bertone (UNLP)

7622 | Analizando el uso de (Dyn)Alloy como Herramienta Educativa
César Cornejo (UNRC), Mariano Politano (UNRC), Fernando Raverta (UNRC), Sonia Permigiani (UNRC), Pablo Ponzio (UNRC), Germán Regis (UNRC), Nazareno Aguirre (UNRC)

7654 | Application of Virtual Reality in a Learning Experience
Victor Udo Obrist Bertrand (UNE), Eustaquio Martinez (UNE)

7657 | Análisis de interacciones y patrones de comportamiento en un curso virtual con Objetos de Aprendizaje
Claudia López de Munain (UNPSJB), Silvina Bramati (UNPSJB), Zulema Beatriz Rosanigo (UNPSJB), Pedro Bramati (UNPSJB), Marcela Torrent (UNPSJB)

7417 | Recomendaciones Personalizadas para Aplicaciones de Soporte al Aprendizaje Ubicuo
Margarita Alvarez (UNSE), Silvina Unzaga (UNSE), Elena Durán (UNSE)

XIII Workshop Tecnología Informática Aplicada en Educación (WTIAE)

7489 | Herramienta de autor para la creación de actividades educativas basadas en Interacción Tangible
Verónica Artola (UNLP), Cecilia Sanz (UNLP), Lucrecia Moralejo (UNLP), Patricia Pesado (UNLP), Sandra Baldasarri (UZ)

7545 | Revisión de Microscopios Virtuales: en busca de características que favorezcan el trabajo colaborativo
Sabrina Lorena Martorelli (UNLP), Sergio R. Martorelli (UNLP), Cecilia Sanz (UNLP), Patricia Pesado (UNLP)

7563 | La Facultad en 3D: un entorno común para dos propuestas educativas
Viviana Harari (UNLP), Claudia M. Banchoff Tzancoff (UNLP)

7568 | Definición de Metadatos Educativos para Repositorios de Instituciones de Educación Superior
Carlos Giorgetti (UNL), Lucila Romero (UNL), Milagros Gutiérrez (UTN-FRSF)

7579 | Uso de un EVA en un contexto de enseñanza presencial
Silvia Susana Escudero (UNLu), José Luis Marazzo (UNLu), Jorge Peri (UNLu)

7582 | Plan de evaluación del material educativo digital EPRA. Propuesta de indagación sobre la motivación intrínseca
Natalí Salazar Mesía (UNLP), Gladys Gorga (UNLP), Cecilia Sanz (UNLP)

7526 | Contextualizando el Conocimiento de Usuarios desde sus Comentarios en Foros
Silvana Aciar (UNSJ), Gabriela Aciar (UNSJ)

7663 | Mundos Virtuales en UNNOBA
Claudia Russo (UNNOBA), Monica C. Sarobe (UNNOBA), Tamara Ahmad (UNNOBA), Michel Lombardo (UNNOBA), Paula Lencina (UNNOBA), Benjamín Ciccerchia (UNNOBA), Hugo Ramón (UNNOBA)

Using Social Network Analysis in the classroom: a case study applying NodeXL

Antonieta Kuz[1], Mariana Falco[1], Enzo de Renzis[1], Leopoldo Nahuel[1], Roxana Giandini[1,2]

[1] LINSI, Universidad Tecnológica Nacional, Facultad Regional La Plata, La Plata, Buenos Aires, Argentina.
{akuz, mfalco, ederenzis, lnahuel} @frlp.utn.edu.ar
[2] LIFIA, Facultad de Informática, Universidad Nacional de La Plata. La Plata, Buenos Aires, Argentina.
giandini@info.unlp.edu.ar

Abstract. The interest in social networks has extended to different disciplines, such as Computer Science. This approach brings Social Network Analysis (SNA) as the study of social structure in different environments, like companies, establishment, and schools, among others. For this reason, this article highlights basic network information like graphs using actors and relations and important concepts related to classroom like structure which constitutes it. Therefore, the main aim of this research is to examine the educational use of SNA. Furthermore, schools in society are a system of actors joined by relationships. Accordingly, the current paper presents a qualitative analysis through a practical approach of SNA, by describing the classroom as a sociometric experiment using NodeXL, to verify the contrast keeping attention in show how students make informal contact and the knowledge that this brings.

Keywords: SNA, Sociometry, Graphs, Classroom, NodeXL

1 Introduction

Information and Communication Technologies (ICTs) can contribute to universal access to education, equity in education, delivery of quality learning and teaching, teachers' professional development and more efficient management in education, governance and administration. As such, the ICTs are becoming an ubiquitous component of classroom learning. They are able to provide additional opportunities to support the learning process and it may be able through the future growing, to transform educational practices [1].

The interest in Social Networks has been increasing and evolving across a wide variety of fields and researches, such as Physics, Psychology and Computer Science [2]. Social Network Analysis (the acronym SNA) was developed in a relatively non-technical manner from the structural concerns of the anthropologist Radcliffe-Brown.

He started to develop a concept of social structure and a web of social life. Social networks have also been studied by Milgram's small world research [3] [4].

Social interactions between students are a major and underexplored part of undergraduate education. Understanding how learning relationships form in undergraduate classrooms, as well as the impacts these relationships have on learning outcomes, can inform educators in unique ways and improve educational reform. Social Network Analysis provides the necessary tool kit for investigating questions involving relational data. We introduce basic concepts within SNA, along with methods for data collection, data processing, and data analysis. In order to make this proposal practical, we present a case study developed in the NodeXL [5] tool to create visualizations of the social networks studied which assist us in our analysis.

We consider the classroom as the main target of this investigation, because this is the environment where professors and students interact every day, involving relational data set to obtain the patterning of relationships among students. Through SNA, it will be possible to deepen the knowledge of social phenomena [6]. This perspective will allow us to point out the intensity of the relationships within the group to study; the degree of cohesion, the structure of a group and each of the positions occupied by each member in the classroom.

This paper is organized as follows: in section 2 we introduce the main Social Network concepts, while section 3 defines SNA. Section 4 presents a Case Study with NodeXL in a school environment and the description of some results obtained through different metrics about the case study. Finally, conclusions and future work are presented in Section 5.

2 Social Networks: Structure and Basic Concepts

With the growing of online social media, everything is connected, people, students, employees, information, events, places, among others. A practical way of making sense of the tangle of connections is through analyzing them as networks. Social networks can be named as a well-defined set of actors such as individuals, groups, organizations, communities, etc. linked to each other, through a relationship or a set of social relations. Another field which formally studies Social Networks is Graph Theory, which is a branch of Mathematics [7].

In social networks, nodes and vertices in a graph represent the actors and relations respectively. Most of the bounds between actors have a purpose. The interpersonal bounds in a network are characterized by the roles and the context of those where they have been developed. The dynamic character of the network appears because sometimes the relations are more or less persistent or also it might be for the existence or not between actors.

A social network is formed by actors. It is very important to try to identify the central individuals in the network. On the one hand, its attributes refer to different aspects, characteristics, and intrinsic properties of the individuals such as opinions, comments, suggestions, and so on. Actors do not act independently; they are influenced in their behavior and attitudes by other actors to whom they are tied. On the other hand, relational data are the contacts, ties and connections, which relate one

actor to another actor. The relations connect pairs and express linkages of actors, they are specific to the context, and the context depends on the interactions among them. Examples of relations are friendship, job relations, flow of information, among others topics [3].

Burt [8] specifies that all social actors involved in a social system that incorporate other actors are significant landmarks in each other's decisions. The relationships that an actor has with others can affect their actions, perceptions and behaviors. So, SNA is focused on uncovering the patterning of how people's interactions will result on data sets. The analysis of the relations between actors allows delimiting the dynamics of flow circulation between actors located in different places in the network.

3 Social Network Analysis: Definition and Tools

Social Network Analysis aims to understand the determinants, structure, and consequences of relationships between actors. SNA is the mapping and measuring of relationships and flows between people, groups, organizations, computers or other information/knowledge processing entities. Social network analysis is the study of structure [9]. It involves relational datasets. That is, structure is derived from the regularities in the patterning of relationships among social entities, which might be people, groups, or organizations.

SNA is a method for visualizing the people and connection power, leading us to identify how we can best interacts to share knowledge. Thus, SNA brings the explanation of behavior of relations that requires an analysis of how the actors are connected to one another considered in a particular environment with contextual factors. In the next section, we provide an analysis of the structure relations between social actors in the school environment [3].

The importance of relationships and emergent structures formed by relationships makes SNA different from other research paradigms, which often focus solely on the attributes of actors. For example, traditional analyses may separate students into groups based on their attributes and search for disproportional outcomes based on those attributes. A social network perspective would focus instead on how individuals may have similar network positions due to shared attributes. These similar network positions may present the same social influences on both individuals, and these social influences may be an important part of the causal chain to the shared outcome. In situations in which a presence or absence of social support is suspected to be important to outcomes of interest, such as formal learning within a classroom, the SNA paradigm is appealing.

NodeXL is a tool for interactive network visualization that leverages the widely available MS Excel application as the platform for representing generic graph data, performing advanced network analysis and visual exploration of networks (see Figure 1) [5]. Likewise, NodeXL is a practical tool, because it uses a highly structured workbook template that includes multiple worksheets to store all the information needed to represent a network graph. NodeXL provides several visualization options availables.

We selected this software tool, because it allowed us to use certain metrics from SNA. Furthermore, we have used this case study to show how the combination of social network and qualitative analysis work together, characterized by communication network defined as a set of "interconnected individuals who are linked by patterned communication flows" [3].

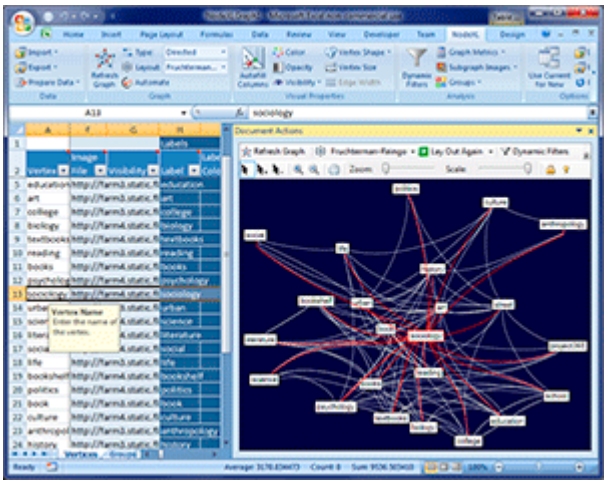


Fig. 1. NodeXL tool

4 Case Study

Burt [8] specifies that all social actors involved in a social system that incorporate other actors are significant landmarks in each other's decisions. The relationships that an actor has with others can affect their actions, perceptions and behaviors. So, SNA is focused on uncovering the patterning of how people's interaction will result on data sets. The analysis of the relations between actors allows delimiting the dynamics of flow circulation between actors located in different places in the network.

Most of the bounds between actors have a purpose. The interpersonal bounds in a network are characterized by the roles and the context of the roles they have developed. The dynamic character of the network appears because sometimes the relations are more or less persistent or also it might be for the existence or not of his actors.

As Hoffman [10] points out, "Sociometry is based on the fact that people make choices in interpersonal relationships. Whenever people gather, they make choices about where to sit or stand; choices about who is perceived as friendly and who is not, who is central to the group, who is rejected, who is isolated". Moreno [11] explained that all criteria have this in common: that the respondents have some actual experience in reference to them, whether ex post facto or present; in sociometric language, they are still "warmed up" to them otherwise the questions would not arouse any significant response.

4.1 Applying Techniques of Collection and Delivery of Social Networks

This section allows us to describe the set of methods and techniques used to characterize a school group specifying the methodological approach of the research, the techniques used to collect the data and the procedures performed for the analysis and interpretation of results.

Sociometric methods became part of SNA and have been developed like a tool that generates an excellent material from the group from specific questionnaires [11]. We take into account and explored a classroom environment composed of a set of eleven (11) students from twelve (12) and thirteen (13) years with a GPA intermediate with individuals highlighted in the positive aspect, i.e. "students with good grades". In order to preserve the identity of the students, we used labels (English, labels), and all the group has been numbered from A1 to A11, while the teacher is A12.

In our study case, we simulate a set of questions to students to obtain a dataset. The sociogram is considered a character sociometric technique, i.e., a method for measuring social relations among members of a group, where its elements are known, have common goals and influence each other. Graphically, a sociogram represents relationships by dots (individuals) appearing together by one or more lines (inter-relationships).

With this information, it is plausible for us to illustrate the most appropriate way in which teachers might make decisions in the classroom. Sociometric data were collected by questions that are formulated online to the students listed in the following list:

1. Who would you choose as a class leader?
2. Who doesn't get on with whom?
3. Who would choose for project teams?
4. Who would you trust in knowing what is really taking place?

4.2 Discovering Group Dynamics through Graphs Topologies

Group dynamics is a system of behaviors and psychological processes occurring within a social group (intragroup dynamics), or between social groups (intergroup dynamics). Kurt Lewin [12] coined the term "group dynamics" to describe the way groups and individuals act and react to changing circumstances. Therefore, the study of group dynamics can be useful in understanding decision-making behavior, tracking the spread of diseases in society, creating effective therapy techniques, and following the emergence and popularity of new ideas and technologies [13].

We use NodeXL to represent graphs from data sets previously obtained from the precedent sociometric questionnaire [5] [14]. Graphs are visual representations of networks, displaying actors as nodes and the relational ties connecting actors as lines. In this case, each node represents a student, considering A1 to A11. Each actor was provided with a list of all actors in the network and asked to indicate those with whom he or she has a particular relation according to the questions previously mentioned. According to the questions set above, it is correct to note that the connections might involve identifying people with whom the student expresses he or she frequently socializes. After that, we display the graph by clicking on the Read Workbook button

directly above the graph pane; a network showing the ties is displayed, as we show in Figure 2.

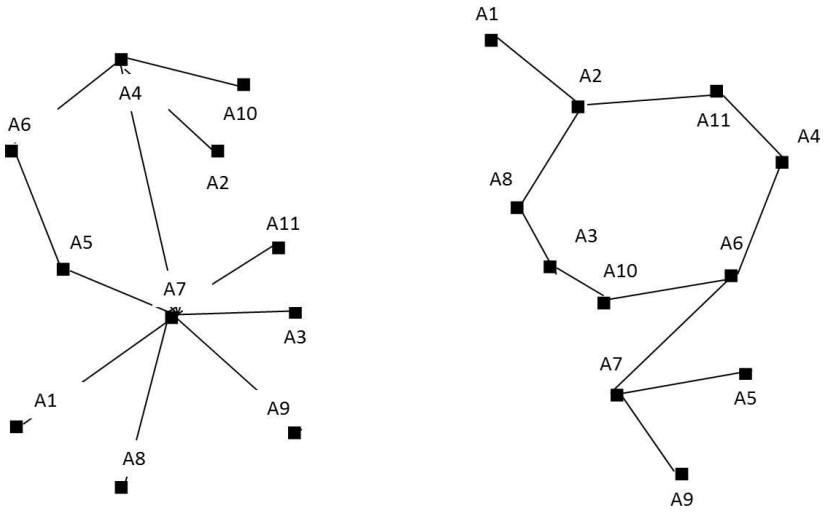


Fig. 2. Social network with a flow of information in the classroom (questions 1 and 3)

Regarding to question 1, we highlight several issues about relationships in the classroom. First, there are no mutual choices because relationships require bilateral content of two involved actors; therefore similarity between members partly increases the likelihood of forming the links within the groups. Then, we emphasize that 5 people elect A7, which makes it a great informal leader (students chosen by both current and by peers).

After that, there are a number of people within the classroom we described as isolates, such as a group of individuals prevented by social barriers from interbreeding with others of their kind and social exclusion. They are: A9, A1, A8, A11, A10 and A2. As in the previous case, there are no mutual choices in response to question 3. It happens that A7 and A2 have been chosen by two people, both present the most choices. Then, we point that between A6, A10, A3, A8, A2, A11 and A4 occurs what graph theory calls a cycle graph which consists of a single cycle with some vertices connected in a closed chain. In this case, we have a directed cycle graph with all the edges being oriented in the same direction. We could say that these relationships suggest a certain bond where friends choose each other (without prior agreement).

Since we obtain the graph of the classroom conferring to the questions 1 and 3, we will continuous to analyze some metrics. It is important to understand that the metrics ultimately became the principal aspect of analysis, and they are obtained with the same software tool.

4.3 Discovering Group Dynamics through Graphs Topologies

Part of the SNA studies all the interaction between individuals and organizations, and flows of information. The analysis of the relations between actors allows delimiting the dynamics of flow circulation between actors located in different places in the network. Most of the bounds between actors have a purpose or may have it because there exist some interest. Therefore, the interpersonal bounds in a network are characterized by the roles and the context of the roles they have developed.

One key direction for education researchers is to study network formation within classrooms, in order to elucidate how the realized networks affect learning outcomes. Network analysis can give a baseline understanding of classroom network norms and illuminate major aspects of students learning. Therefore, network data are collected at the individual level, but the analyses occur at the structural level, with the use of different measures. This allows us to obtain the classroom structural characteristics.

This property refers to the overall pattern of relationships of relationships between the system's actors, by capturing the size and internal connectivity of a network as well as attributes of each node. NodeXL supports a minimal set of the most crucial network measures for individual nodes, such as: density, distance, reachability, degree and betweenness.

4.3.1 Cohesion-density

Perhaps the most basic measurement in network analysis is network density, which points out how many links are observed in a whole network divided by the total number of links that could exist if every actor were connected to every other actor. In Figure 2 it is possible to discriminate that there are 11 ties out of a possible 121 for the organizational network, giving a density of 0.091.

4.3.2 Cohesion-distance

Consider two persons, call them A and B, which each might have five friends. But suppose that none of A's friends have any friends except A. B's friends, in contrast, have five friends each. The information available to B is that B's friends have potential for influence which is far greater than A's; usually known as being a "friend of a friend" may be quite consequential. To capture this aspect of how individuals are embedded in networks, one main approach is to examine the distance that an actor is from others.

If two actors are adjacent, the distance between them is one (that is, it takes one step to go from the source to the receiver). As shown in Figure 2, for question 1 A7 is a distance of 2 from A6, A2, A10, and A9; and for question 3 A7 is a distance of 7 from A4 instead of A11 who can reach A4 within an only tie. This is the notion of ‘‘degrees of separation’’ made familiar to many by a popular play [15].

4.3.3 Cohesion – Reachability

An actor is "reachable" by another if there exists any set of connections by which we can trace from the source to the target actor, regardless of how many others fall between them. In other words, reachability measures whether actors within a network are related, either directly or indirectly, to all other actors [16]. If the data are asymmetric or directed, it is possible that actor A can reach actor B, but that actor B cannot reach actor A.

Regarding the question 1, with the exception of the five isolates (A8, A11, A1, A10, and A9) all of the remaining actors in figure 2 can reach one another. For question 3 with the exception of the three isolates (A5, A9, and A1) all of the remaining actors in figure 2 can reach one another.

4.3.4 Degree Centrality

Centrality measures identify the most prominent actors, those who are extensively involved in relationships with other network members. [17] Centrality indicates one type of ‘‘importance’’ of actors in a network: in lay terms, these are the ‘‘key’’ players. Degree centrality is the sum of all other actors who are directly connected to ego. It signifies activity or popularity. Lots of ties coming in and lots of ties coming out of an actor would increase degree centrality.

For question 1, A7 has the highest degree centrality with five direct ties and A4 is the next most central with three direct ties. For question 3, A7 and A2 have the highest degree centrality with two direct ties each.

4.3.5 Betweenness Centrality

Betweenness centrality is the number of times an actor connects pairs of other actors, who otherwise would not be able to reach one another. It is a measure of the potential for control as an actor who is high in ‘‘betweenness’’ is able to act as a gatekeeper controlling the flow of resources between the alters that he or she connects [17]. A7 is by far the most powerful actor depicted in Figure 2.

All actors in the network must go through A7 to reach A1, A11, A8, A3 and A9. The next most powerful is A4, because all actors must get through him to reach A2 and A10. Also, in question 3, all actors in the network must go through A6 to reach A10, A3, A8, A2, A11 and A4. By going through A2, we can reach A1. In the other way, through A7 we can reach A5 and A9.

These measures of centrality are purely structural measures of popularity, efficiency, and power in a network, namely that the more connected or central an actor is the more popular, efficient, or powerful.

5 Conclusions and future work

Beyond considering a group of students and determining the dynamics, this research focused on deepening the search for a simple way in which it is plausible to help not only the teacher, but mainly the students, paying attention to their views and needs [11]. We have used NodeXL, a software tool through which we obtained different results on relations in the classroom, and by the metrics we appreciate and support the textual analysis of the graphs. Therefore, this study not only was a description of the graph, it was also an objective assessment of the hidden reality on students, their feelings, their relationships with classmates and teachers, as well as their perception of the organizational hierarchy.

As we mentioned at the beginning, SNA not only can be applied to an organization like schools, but also to different environments. In the school, we see that SNA is very important, because in this environment children and teenagers improve their character through emotions, feelings but especially, by fulfilling the meaning of relationships. They learn how to interact with the equals, how to build what we know as social consciousness. The connection and exchange between students are the most important sources of information and knowledge, and this is it because students trust more those they know than those they don't.

As regards future work, we will try to analyze the network formed by students, and after that, we are going to study their choices, which will determine what others will see and how they will be connected to others. So we will intend to consider the concept of trust and the different concepts related to it, like confidence and their differences. In this way, we will introduce trust to Computer Science, and we will also place it in the center of our attention in SNA. Moreover, we are going to use NodeXL to represent that. In this manner, we will take into account the properties and the most important components: the trustor and the trustee, and how they interact.

Acknowledgments. The authors wish to thank the authorities of the National Technological University - Regional La Plata (UTN - FRLP) for their support, as well as the Innovation Laboratory Information Systems (LINSI).

References

1. Greenhow, C. and Robelia, B.: Old Communication, New Literacies: Social Network Sites as Social Learning Resources. Journal of Computer-Mediated Communication, 14:130–1161. doi: 10.1111/j.1083-6101.2009.01484.x (2009)
2. Hanneman, R. A. & Riddle, M.: Introduction to social network methods, University of California, Riverside, Riverside, CA, (2005)
3. Scott, J.: Social Network Analysis: A Handbook, Sage Publications (2000)
4. Watts, D.: Six Degrees: The Science of a Connected Age, W. W. Norton (2004)
5. Matei, S. A.: 'Analyzing Social Media Networks with NodeXL: Insights from a Connected World by Derek Hansen, Ben Shneiderman, and Marc A. Smith', Int. J. Hum. Comput. Interaction 27(4), 405–408 (2011)
6. Kuz, A., Falco, M.: Herramientas sociométricas aplicadas al ambiente áulico, in

Congreso Nacional de Ingeniería Informática/Sistemas de Información (CoNaIISI 2013) (En línea) Córdoba, Argentina. ISSN: 2347-0372 (2013)
7. Tang, L., Liu, H.: Graph mining applications to social network analysis. In C. Aggarwal and H. Wang, editors, Managing and Mining Graph Data, chapter 16, pages 487-513. Springer (2010)
8. Burt, R.S.: Models of Network Structure. Annual Review of Sociology, vol. 6, pp. 79-114. JSTOR publishers (1980)
9. Wellman, B., Berkowitz, S.D.: Social structures: a network approach. Greenwich: JAI Press, (1997)
10. Hoffman, C., Wilcox, L., Gomez, E., & Hollander, C.: Sociometric applications in a corporate environment. Journal of Group Psychotherapy, Psychodrama & Sociometry, 45(1), Spring, 3-16.(ISSN: 0731-1273 (1992)
11. Moreno, J.L.: Who Shall Survive? 2nd edition, New York. Beacon House (1953)
12. Lewin, K.:Dinámica de la personalidad: selección de artículos, Filosofía, Psicología, Pedagogía, Morata (1973)
13. Backstrom, L.; Huttenlocher, D.; Kleinberg, J.; Lan, X.: "Group formation in large social networks". Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '06. p. 44. doi:10.1145/1150402.1150412. ISBN 1595933395 (2006)
14. Dunne, C., Shneiderman, B.: Improving graph drawing readability by incorporating readability metrics: A software tool for network analysts , University of Maryland, Human-Computer Interaction Lab Tech Report HCIL-2009-13 (2009)
15. Guare, J.: Six degrees of separation: a play. New York: Vintage Books (1990)
16. Doreian, P.: On the connectivity of social networks. Journal of Mathematical Sociology 1974;3:245–58.
17. Freeman, L. C.: 'Centrality in social networks conceptual clarification', Social Networks p. 215 (1978)

Mejoramiento de la Adaptación usando la Lógica Difusa

Constanza Huapaya[1], Leonel Guccione[1], Delia Benchoff[1], Marcela Gonzalez[2]
Francisco Lizarralde[1]
[1] Departamento de Matemática, Facultad de Ingeniería, [2]Facultad de Psicología
Universidad Nacional de Mar del Plata,Juan B. Justo 4302
7600 Mar del Plata, Argentina
{constanza.huapaya, leonel.guccione,ebenchoff.sead,francisco.lizarralde}@gmail.com,
mpgonza@mdp.edu.ar

Resumen. Se presenta un módulo de un Modelo del Estudiante de un Ambiente Virtual de Aprendizaje a fin de promocionar la personalización de los materiales instruccionales basada en los niveles de conocimiento dinámicos y estilos de aprendizaje. El mejoramiento está dado por la inclusión de la experiencia de expertos en el dominio que se enseña cuyas opiniones fueron expresadas en reglas difusas utilizando dos variables lingüísticas de entrada (nivel de conocimiento y estilo de aprendizaje) y dos variable lingüísticas de salida (dificultad y complejidad). Estas dos últimas variables categorizan los materiales instruccionales.

Palabras clave: adaptación, personalización, estilos de aprendizaje, Lógica Difusa, estudiante de ingeniería

1 Introducción

Los avances en el desarrollo de las tecnologías de la información producen una notoria influencia en la educación basada en computadoras. En la búsqueda del mejoramiento de los sistemas informáticos educativos, las características más examinadas y utilizadas son la adaptabilidad y la personalización de los sistemas de aprendizaje. Se pretende sustentar tanto la diversidad como las necesidades individuales de los alumnos. Este es un objetivo perseguido por muchos investigadores mediante el uso del modelo del estudiante, donde se describe una combinación de características personales, comportamiento y niveles de conocimiento [1] [2]. Dentro de los rasgos personales, los estudiantes poseen diferentes estilos de aprendizaje, esto es, aprenden de modos diferentes. En esta dirección, Felder y Silverman [3] sostienen que los alumnos que estudian con material instruccional ajustado a sus estilos de aprendizaje aprenden más efectivamente y progresan mejor. En este artículo se presenta un modelo para adecuar los materiales a estilos de aprendizaje y nivel de dificultad.

1.1 Aprendizaje personalizado, aprendizaje adaptativo

En el aprendizaje personalizado se pretende que el estudiante recorra su propio camino hacia el conocimiento. Este tipo de aprendizaje comprende cualquier acción que tanto un profesor como un software ejecutan para enseñar un tema a un estudiante entendido como una individualidad. La personalización puede ser aplicada al contenido del currículo, métodos, ritmo del aprendizaje, etc. Es una opción que toma en cuenta las necesidades individuales de los alumnos y respeta sus diferencias. Así, identificando las características de los estudiantes, el docente puede proponer un material gráfico para un alumno, mientras que para otro entregará material principalmente en texto, porque así lo prefieren. La idea común de un enfoque personalizado es brindar una experiencia de aprendizaje al alumno, valorando a cada individuo tanto en el salón de clases como mediante el empleo de un software educativo.

El aprendizaje adaptativo puede entenderse como la adaptación al estudiante siguiendo los datos que se desprenden durante las interacciones estudiante/computadora, generalmente, en línea. Entre otras acciones, el sistema computacional modifica la presentación del material instruccional como respuesta a las acciones del estudiante. El proceso adaptativo se "ajusta" al nivel de desempeño mostrado e intenta predecir qué tipo de material necesitará el alumno en un punto determinado de su camino al aprendizaje. Los sistemas del aprendizaje adaptativo intentan todo el tiempo sostener la evaluación formativa, en un esfuerzo por proveer el siguiente paso tutorial apropiado en el proceso de aprendizaje de los estudiantes.

Los sistemas computacionales educativos, entonces, deben considerar diferentes necesidades y la diversidad en cuanto a características de aprendizaje: dispares niveles de conocimiento, habilidades cognitivas, preferencias, estilos de aprendizajes, emociones, reacciones, etc. Un sistema que intente poseer rasgos adaptativos debería considerar cómo identificar dicho conjunto de características, a fin de inferir sus necesidades y preferencias, para entregar el material de aprendizaje adecuado según las necesidades del estudiante, y, de este modo, lograr aconsejar y proveer retroalimentación efectiva. [4].

1.2 Técnicas adaptativas

Un ambiente de aprendizaje puede ser considerado adaptativo si es capaz de monitorear las actividades del usuario, interpretarlas siguiendo modelos del dominio específicos, inferir requerimientos del usuario y preferencias más allá de la interpretación de las actividades, para finalmente actuar sobre todo el conocimiento adquirido de sus usuarios y facilitar dinámicamente el proceso de aprendizaje. El comportamiento adaptativo de los ambientes presenta diversas manifestaciones: interacciones adaptativas, la entrega de cursos adaptativos, articulación de material instruccional y ayuda en la colaboración adaptativa. Las *interacciones adaptativas* son las modificaciones que tienen lugar en las interfaces del sistema a fin de facilitar las interacciones pero sin modificar los contenidos. Como ejemplos se puede mencionar el uso de esquemas o gráficos, tamaño de las letras del texto, y la reorganización de las tareas utilizando metáforas a nivel semántico. Las modalidades de entrega de *cursos adaptativos* son las técnicas más usadas actualmente. En particular, este concepto se refiere a cursos hechos a medida del estudiante individual.

El objetivo es optimizar el ajuste entre los contenidos y las características del usuario/estudiante. Ejemplos de estas técnicas son la re-estructuración dinámica del curso, la ayuda en la navegación adaptativa y la selección adaptativa de los materiales del curso [5]. El *descubrimiento y articulación del material instruccional* se realiza desde diversas fuentes como repositorios o bases de conocimiento. La adaptación, en esta perspectiva, tiene como base modelos de adaptación y conocimiento derivado del monitoreo del estudiante. Pueden proponerse dos visiones sobre la búsqueda del material relevante: el estudiante navega dentro del material o el autor/profesor prepara el material orientado a grupos específicos de alumnos. La ayuda en la *colaboración adaptativa* consiste en capturar el soporte adaptativo que involucra la comunicación entre múltiples estudiantes (i.e. interacción social).

2 Modelo propuesto

La propuesta que se muestra usa como técnicas adaptativas, los *cursos adaptativos* y principalmente la *preparación del material instruccional* planificado por el autor/profesor dirigido a grupos específicos de alumnos. Para la aplicación de estas técnicas se han utilizado dos fuentes de información para la personalización, *estilos de aprendizaje* y *nivel de conocimiento*. El nivel de conocimiento se traduce en perfiles dinámicos de los estudiantes que serán utilizados para reflejar su avance. Para determinar los estilos de aprendizaje se ha usado los resultados del cuestionario de Felder y Silverman.

2.1 Estilos de aprendizaje

Un estilo de aprendizaje es el modo que caracteriza a un individuo cuando adquiere, retiene y recupera información. Los estudiantes, en particular, muestran diferentes fortalezas y preferencias cuando adquieren información. Esto es, poseen distintos estilos de aprendizaje. En el presente trabajo se ha adoptado el modelo de estilos de aprendizaje formulado por Richard Felder y Linda Silverman [6] el cual fue diseñado para estudiantes de ingeniería, con el fin de identificar las diferencias de estilos de aprendizaje. A partir de dicha identificación, los autores indican que puede formularse un enfoque de la enseñanza que responda a las necesidades de aprendizaje de todos los estudiantes. El modelo de Felder y Silverman clasifica a los estudiantes de acuerdo con sus preferencias por una u otra categoría en cada una de las siguientes cuatro dimensiones:

Sensitivo / Intuitivo. El estudiante con estilo preferencialmente sensitivo, tiende a ser concreto, y orientarse por hechos o procedimientos bien establecidos; el alumno con estilo intuitivo es innovador, se caracteriza por preferir el pensamiento abstracto y guiarse por teorías y significados subyacentes, con principios generales más que datos concretos.

Visual / Verbal. El estilo de aprendizaje visual indica la preferencia por representaciones visuales de los materiales de estudio tales como imágenes, diagramas, mapas conceptuales, gráficos. En el otro extremo de la dimensión, el

estudiante con un estilo verbal, se sentirá más a gusto con explicaciones tanto escritas como habladas.

Activo / Reflexivo. El alumno activo aprende mejor al trabajar dinámicamente con el material, prueba cosas, y prefiere el trabajo en grupo. En cambio, el alumno reflexivo opta por pensar más que realizar pruebas, y se inclina por trabajar solo o con un único compañero.

Secuencial / Global. El estilo secuencial define a un estudiante que aprende mejor en pequeños pasos incrementales, detallista; mientras que un alumno con un estilo de aprendizaje global, aprenderá de manera holística, estableciendo relaciones con su experiencia, más interesado por el conocimiento general.

A partir del modelo, Soloman y Felder [7], crearon el Cuestionario de Estilos de Aprendizaje (ILS, Index of Learning Styles), un instrumento con 44 ítems de respuesta dicotómica, en línea, y considerando las cuatro dimensiones propuestas. A los fines del presente trabajo, se ha aplicado el cuestionario a los alumnos de asignaturas pertenecientes a las carreras de Ingeniería, siendo procesados los resultados en la página web de la Universidad del Estado de North Carolina, provista por los autores referenciados. Cada dimensión posee un rango que varía de 0 a 11. Si el resultado del cuestionario toma un valor entre 0 a 3 (en cualquiera de las dos direcciones), el estudiante se encuentra bien *equilibrado* entre las dos dimensiones que figuran en los extremos de la escala; si el puntaje en la escala es 5 o 7, tiene una *equilibrada preferencia* por el extremo al que se acerca y si el puntaje en la escala es 9 u 11, tiene una *fuerte preferencia* por el extremo donde se encuentra. Se ha elaborado la dimensión activo/reflexivo sobre 47 estudiantes de ingeniería (ver figura 1).

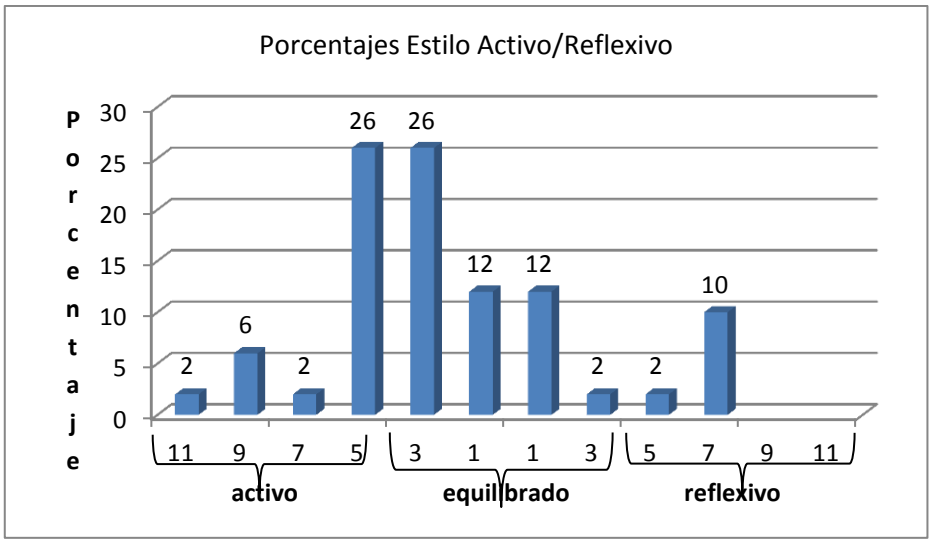


Fig. 1. Resultados del cuestionario de Felder Soloman

Para *el nivel de conocimiento* se han usado los estereotipos *aprendiz, nivel intermedio de conocimiento* y *experto* desarrollados en [8].

2.2 Breve introducción a la Lógica Difusa

La Lógica Difusa es usada para tratar la incertidumbre presente en problemas reales causada por la información imprecisa como la generada por la subjetividad humana [9]. En el modelado de los sistemas suelen utilizarse variables con valor incierto y esta problemática es resuelta con el uso de los conjuntos difusos. Los conjuntos difusos describen a las variables con valores como "malo", "regular" y "bueno" en lugar de valores booleanos "verdadero/falso" o "si/no" [10]. Los conjuntos difusos se determinan por una función de pertenencia al conjunto difuso A expresada con $\mu A(x)$:

$$\mu A(x):X\rightarrow[0,1],\text{donde} \quad \mu A(x)=\begin{cases} 1 \text{ si } x \text{ está en } A \\ (0,1) \text{ si } x \text{ está parcialmente en } A \\ 0 \text{ si } x \text{ no está en } A \end{cases} \qquad (1)$$

El valor de $\mu A(x)$ se denomina grado de pertenencia y posee un valor real entre 0 y 1. Cuando x pertenece completamente al conjunto difuso A, $\mu A(x)$ vale 1 y cuando x no está en A, $\mu A(x)$ vale 0. Cuanto más alto sea el valor de la función de pertenencia, más fuerte es el grado de pertenencia de x al conjunto A.
Un sistema de inferencia difusa (FIS) es un sistema que usa la teoría de los conjuntos difusos a fin de mapear las variables lingüísticas de entrada (características en caso de clasificación difusa) sobre variables lingüísticas de salida (clases en el caso de clasificación difusa). Las reglas difusas son una colección de sentencias lingüísticas que describen como el FIS debe tomar las decisiones sobre una entrada o controlando una salida. Las reglas difusas IF-THEN poseen un antecedente y un consecuente. Veamos un ejemplo:

Si el *estudiante es excelente* entonces *tendrá muy buenas notas en los exámenes*

2.3 Inferencia difusa en el modelo

A fin de adaptar el material instruccional se han definido dos variables lingüísticas de entrada. La primera es el *estilo de aprendizaje* la cual toma tres términos (conjuntos difusos): activo, equilibrado y reflexivo. La segunda variable, *nivel de conocimiento,* toma los valores aprendiz, intermedio y casi_experto. En la figura 2 se aprecia la definición de las funciones de pertenencia. Para este desarrollo se ha utilizado la herramienta de código abierto FisPro (Fuzzy Inference System Professional).

Las variables lingüísticas de salida caracterizan el material instruccional que será mostrado a los estudiantes. El parámetro, en la dimensión activo/reflexivo, que se ha considerado es la preferencia por estudiar en grupos. Se han definido las siguientes variables: *nivel de dificultad* (con los términos baja, media y alta) y *para grupos* (con los términos individual, equilibrado y grupal) cuya definición se ve en la figura 3.

Se definieron nueve reglas del sistema de inferencia difusa siguiendo la recomendación de docentes experimentados. Estas se encuentran ilustradas en la figura 4.

En la figura 5 se muestra un caso particular donde se aprecia que a un alumno con nivel de conocimiento un poco más que intermedio y estilo de aprendizaje

ligeramente activo se le recomienda material de estudio con dificultad media/alta y trabajos en grupo.

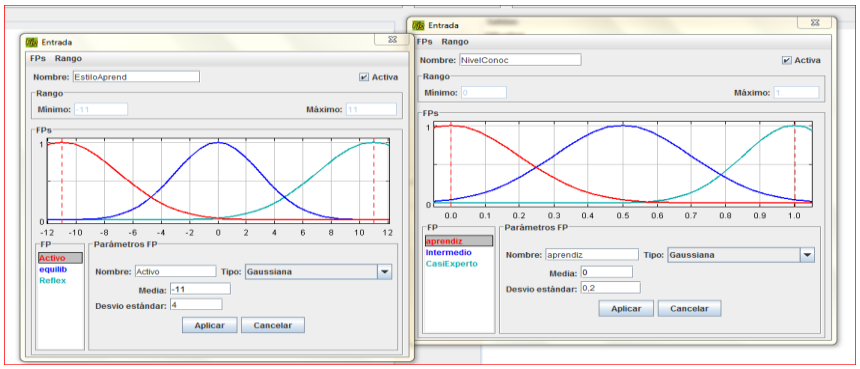


Fig. 2. Variables lingüísticas de entrada y sus correspondientes términos.

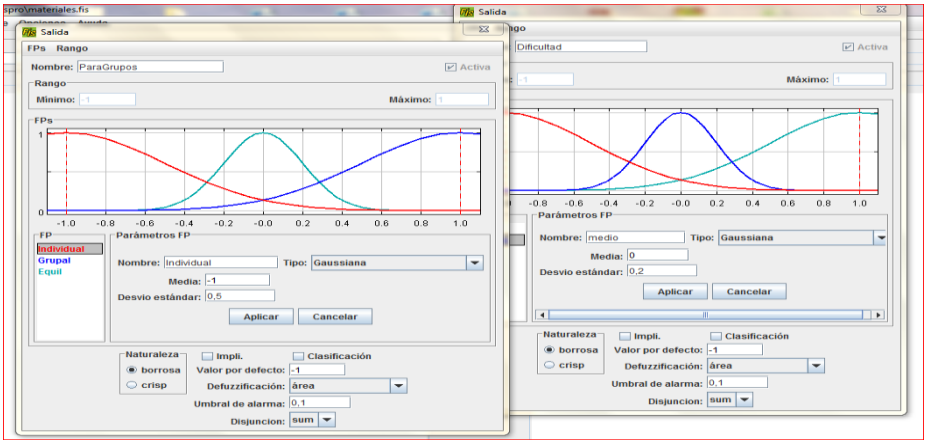
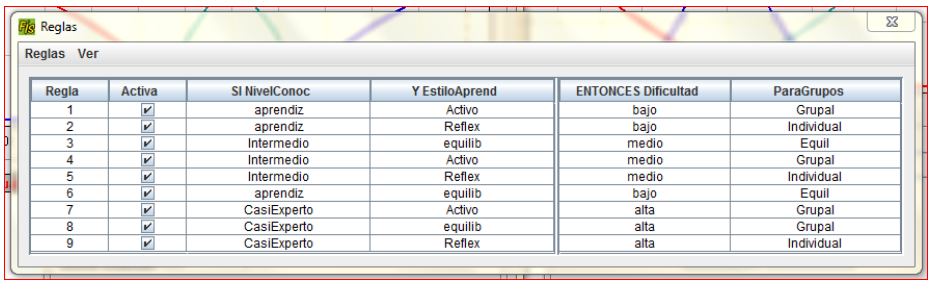


Fig. 3. Variables lingüísticas de salida y sus correspondientes términos.



Regla	Activa	SI NivelConoc	Y EstiloAprend	ENTONCES Dificultad	ParaGrupos
1	✔	aprendiz	Activo	bajo	Grupal
2	✔	aprendiz	Reflex	bajo	Individual
3	✔	Intermedio	equilib	medio	Equil
4	✔	Intermedio	Activo	medio	Grupal
5	✔	Intermedio	Reflex	medio	Individual
6	✔	aprendiz	equilib	bajo	Equil
7	✔	CasiExperto	Activo	alta	Grupal
8	✔	CasiExperto	equilib	alta	Grupal
9	✔	CasiExperto	Reflex	alta	Individual

Fig. 4: reglas de inferencia difusa

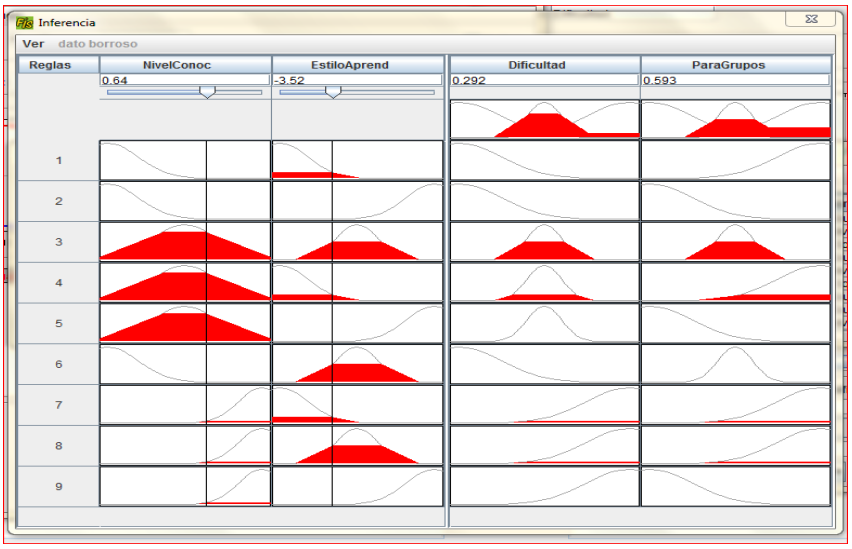
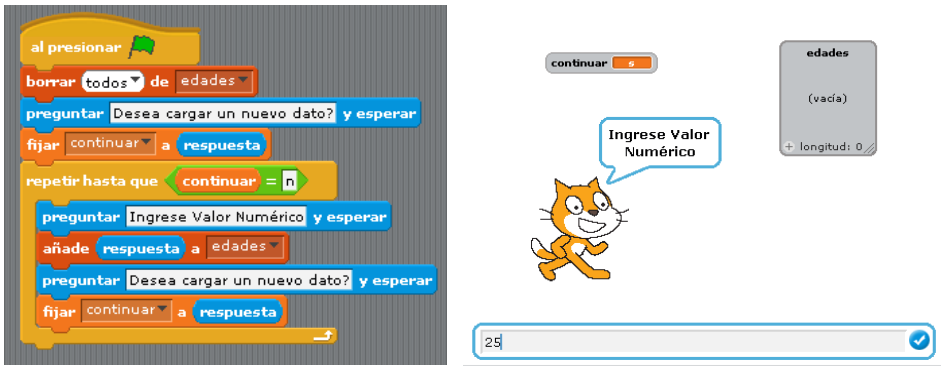


Fig. 5: ejemplo de inferencia

3 Resultados de la inferencia difusa: ejemplos de material instruccional

Como resultado de la aplicación del sistema de inferencia difusa recién mostrado, se ha desarrollado material instruccional en el dominio de la programación básica correspondiente a asignaturas del ciclo inicial de la carrera de Ingeniería en Informática de la UNMDP. Se muestran tres tipos de ejercicios desarrollados en Scratch y utilizados en Moodle. Scratch es una herramienta propicia para la enseñanza de la programación a nivel inicial, ya que brinda un entorno gráfico muy amigable, desarrolla el pensamiento algorítmico, fomenta la utilización y exploración de los recursos (objetos, gráficos, concurrencia, mensajes, estructuras de control, variables, listas, etc.) de una manera intuitiva, debido a la forma de construir el código (bloques encastrables que representan su funcionalidad de manera visual) y también debido a que los resultados de los avances se pueden validar fácilmente. En cada ejemplo se ha agregado entre paréntesis, nivel de dificultad y si se ha desarrollado para grupos, esto es los términos de las variables lingüísticas de salida..

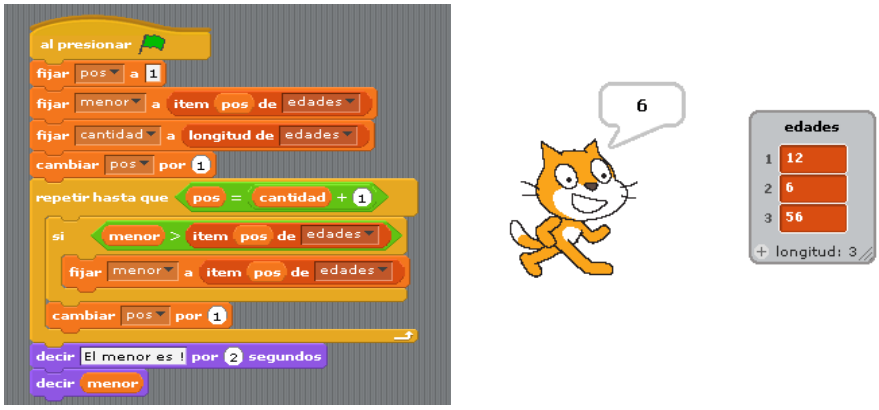
Ejemplo 1 (*Dificultad baja, individual*): Hacer un programa que almacene una lista con valores numéricos. La cantidad es determinada por el usuario.



(a) **(b)**

Fig. 6: ejemplo simple donde en (a) se ve código y en (b) pantalla de resultados

Ejemplo 2 (*Medio, equilibrado*). Hacer un programa que encuentre el menor valor de una lista de números.



(a) **(b)**

Fig. 7: ejemplo intermedio donde en (a) se ve código y en (b) pantalla de resultados

Ejemplo 3 (*Trabajo en grupo*): programar el juego de Los 5 Escalones. Es un juego de preguntas y respuestas con las siguientes características: el jugador responde preguntas realizadas por el juego y a medida que acierta, sube un escalón. El participante gana cuando llega a la cima de la escalera (responder bien 5 preguntas de 8). Los personajes son: Preparador, Animador y el Participante. El Preparador del Juego pide que se ingresen 8 preguntas con sus respectivas respuestas y da inicio al Juego. El Animador realiza, una por una todas las preguntas al Participante, hasta llegar a 5 correctas o hasta completar las 8. Al completar el juego, el Animador

informa al Participante el resultado del juego con un mensaje, si el mensaje es de "ganador" el Participante festeja, caso contrario se va del juego.

Fig. 8: Interfaces correspondientes al ejemplo 3

En la figura 8 se observan dos pantallas. La primera es una presentación. La otra permite el ingreso de datos.

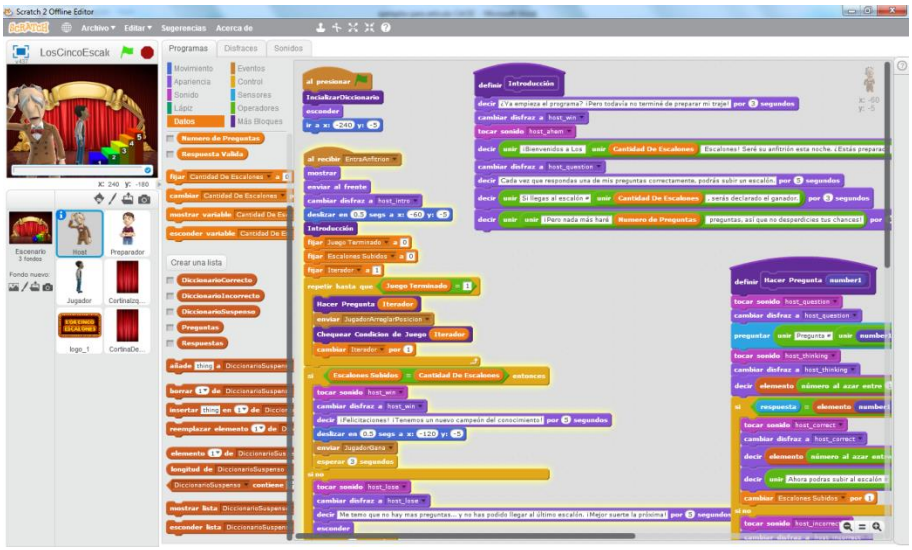


Fig. 9: pantalla de desarrollo

En la figura 9 se aprecia una pantalla de creación del juego extraída de una animación y explicaciones auditivas sobre el desarrollo donde se distinguen el escenario, objetos, variables, listas y parte del código construido.

4 Conclusión

Este artículo tiene por objetivo promocionar la adaptación en Ambientes Virtuales de Aprendizaje. En tal sentido, adaptando los materiales instruccionales a las características personales de los estudiantes, se optimiza el tiempo y el esfuerzo de éstos para alcanzar una apropiada adquisición del conocimiento. En pos de lograrlo, se ha consultado a docentes experimentados quienes volcaron su pericia en reglas difusas a fin de categorizar los materiales instruccionales. Se ha utilizado la Lógica Difusa porque representa de modo realista la experticia humana. Se tiene previsto agregar pruebas adaptativas al material de estudio a fin de mejorar la adaptación al estudiante en un Ambiente Virtual de Aprendizaje.

Referencias

1. Gu Q., Sumner T.: Support Personalization in Distributed E-Learning Systems through Learner Modeling . In: 2nd Information and Communication Technologies, ICTTA 2006, vol. 1, pp. 610–615., (2006).
2. 2. Tian, F., Zheng, Q., Gong, Z., Du, J., & Li, R.. Personalized learning strategies in an intelligent e-learning environment. In Proceedings of the 11th International Conference on Computer Supported Cooperative Work in Design (pp. 973–978). (2007).
3. Felder R. & Silverman, L.K. . Learning and teaching styles in engineering education Engr. Education, 78(7), 674-681 (1988)
4. Chrysafiadi K.; Virvou M.. Advances in Personalized Web-Based Education. Springer Cham Heidelberg. (2014)
5. Brusilovsky, P. :Adaptive hypermedia. User Modeling and User Adapted Interaction, 11(1/2), pp87-110. (2001).
6. Felder R. Are Learning Styles Invalid? (Hint: No!). On-Course Newsletter, September 27, recuperado de : http://www.oncourseworkshop.com/Learning046.htm. (2010).
7. Soloman B. y Felder R. (n.d.). Index of learning styles questionnaire. Recuperado el 10 de junio de 2014 de: http://www.engr.ncsu.edu/learningstyles/ilsweb.html
8. Huapaya C., Gonzalez M., Benchoff E., Guccione L., Lizarralde F. Estimación del Diagnóstico Cognitivo del Estudiante de Ingeniería y su mejora con pruebas adaptativas. X Congreso de Tecnología en Educación y Educación en Tecnología. Pp. 480-489. (2015).
9. Zadeh L. Computing with Words. Principal Concepts and Ideas. Springer. (2012).
10. Gokmen G. , Akincib T., Tektas M., Onat N. y Kocyigita G. Evaluation of student performance in laboratory applications using fuzzy logic. Procedia Social and Behavioral Sciences, pp 902-909. (2010).

Pautas para la actualización de la Plataforma Moodle entre versiones distantes. Una experiencia concreta.

María Fernanda Guzmán, Pablo J. Rodríguez Rey, Lía F. Torres Auad, Nicolás G. Auvieux, Julio A. Escalante Figueroa, Ticiano Torres Peralta

Facultad de Cs. Exactas y Tecnología, UNT, Argentina
mferguzman@herrera.unt.edu.ar, pablojrrey@yahoo.com.ar,
ltorresauad@herrera.unt.edu.ar, nicoauvieux@gmail.com,
jescalante@herrera.unt.edu.ar, tjt132@gmail.com

Resumen: El uso de plataformas virtuales de aprendizaje para la implementación de Entornos Virtuales Institucionales de Educación Superior es una realidad global. La Facultad de Ciencias Exactas y Tecnología de la UNT, a través del CEDITE (Centro de Educación a Distancia e Investigación en Tecnologías Educativas) inició su propio camino en el año 2011 para la implementación de la modalidad virtual, con la creación del Entorno FACETVirtual. El equipo responsable encaró la tarea de actualización de la plataforma utilizada hacia una versión más reciente, atendiendo la necesidad de garantizar la mejora continua de la calidad de las prestaciones. El objetivo es presentar las consideraciones previas, la secuencia de actividades y decisiones llevadas a cabo en el proceso de actualización, presentando además las dificultades que tuvieron que sortearse. Finalmente se proponen algunas claves a tener en cuenta como administradores de Entornos Virtuales de Aprendizaje a la hora de la actualización de versiones.

Palabras clave: plataforma virtual, actualización, rendimiento, MOODLE

1 Introducción

La flexibilización de las instituciones de educación superior para adaptarse a las necesidades de la sociedad actual pasa por la explotación de las tecnologías de la información y la comunicación en los procesos de formación. Lograr que esos procesos sean de calidad implica cambios en la concepción de los alumnos–usuarios, cambios en los profesores y cambios administrativos en relación con el diseño y distribución de la enseñanza y con los sistemas de comunicación que la institución establece.

Es importante destacar que la calidad está relacionada, en parte, con la estructura del entorno y la tecnología seleccionada para soportarlo, pero la enseñanza y el aprendizaje no se optimizan sólo como resultado de mejores entornos y con el uso de la tecnología. Estructura y tecnología, aun siendo importantes, no constituyen el verdadero fundamento de un entorno de aprendizaje –sea presencial, sea virtual– si se ignora o soslaya la función pedagógica que deben asumir los mismos.

Por lo tanto, cualquier proceso de actualización de la plataforma de software que soporta un Entorno se enmarca, no sólo en la necesidad de acompañar las innovaciones tecnológicas, que se suceden a través del tiempo, sino, y especialmente, en las posibilidades metodológicas que esta actualización depara en orden a optimizar los aprendizajes.

Los sistemas de gestión de aprendizajes y contenidos LCMS[1] (Learning Content Management Systems), nombre con el que se conocen las plataformas de software destinadas a la creación de Entornos Virtuales, suponen la integración de dos vías tradicionalmente separadas como los CMS (Content Management Systems) y los LMS (Learning Management Systems). Se convierten en una propia entidad editora, con autosuficiencia en la publicación de contenido de una forma sencilla, rápida y eficiente, resolviendo los inconvenientes de las anteriores plataformas. Ofrecen facilidad en la generación de los materiales, flexibilidad, adaptabilidad a los cambios, control del aprendizaje, mantenimiento actualizado del conocimiento. Entre los ejemplos de LCMS comerciales se encuentran: Connect Learning, Vertice, WebCT, EduStance, Blackborad, E-ducativa, eCollege, Educallao, Eminus, Fronter. Mientras que las de tipo Opensource: ILI, WHiteborad, Moodle, Dokeos, Claroline, ATutor, DOcebo, .LRN, Gabesh, eFront, ILIAS, OLAT.

Moodle es la plataforma LCMS más utilizada en el mundo. Más de 26 millones de usuarios distribuidos en 47 mil sitios, en 196 países y traducido en 75 idiomas. Sólo como ejemplos: La británica Open University, con 180.000 alumnos, usa Moodle desde hace años y la Universitat Oberta de Catalunya empieza a implantarlo, así como también la Universidad de Buenos Aires[2].

Uno de sus aspectos más trascendentes, consiste en que se retroalimenta del trabajo realizado por múltiples instituciones y participantes que colaboran en red, lo cual nos permite acceder libremente e incorporar a nuestra asignatura múltiples módulos y recursos creados por otros usuarios[3].

Precisamente la metodología de desarrollo del open source, y de los métodos de trabajo colaborativo para el desarrollo de estas aplicaciones, han permitido, a Moodle en particular, liderar globalmente. Antes de introducirnos a la descripción de la experiencia presentamos una breve descripción del Entorno Virtual y sus alcances.

2 Creación y Evolución de FACETVirtual

La Facultad de Ciencias Exactas y Tecnología, una de las pioneras en la UNT en la incorporación de las TIC en procesos de enseñanza–aprendizaje, delegó en el CEDITE la creación de un Entorno Virtual, que se concretó en el mes de agosto de 2011. Dicha gestión, enmarcada en el "Plan Estratégico para la inserción de TIC en procesos educativos y la implementación de la Educación a Distancia en la FACET", implicó una serie de pasos y decisiones, entre ellas, la elección de la plataforma educativa a implementar, tarea que tuvo en cuenta todas las consideraciones previamente detalladas.

"En la actualidad existen numerosas plataformas e-learning que permiten a instituciones educativas implementar la educación a distancia. La elección de una de ellas a la hora de implementar un proyecto e-learning, en el ámbito educativo

universitario no es una tarea menor. Los criterios de selección pueden tener en cuenta varios aspectos: presupuesto con el que se cuenta, contexto donde se llevará a cabo la propuesta educativa y objetivos que se pretende alcanzar a través de este tipo de formación, entre otros. Después de un análisis en el que evaluamos las distintas plataformas, se optó por elegir la plataforma Moodle"[4].

Si bien, la plataforma Moodle hacia fines de 2011 ya había presentado oficialmente la versión 2.0, se efectivizó la instalación de la versión 1.9 dentro de nuestra unidad académica, ya que su performance y estabilidad estaban probadas hasta ése momento.

Hasta la fecha, con casi 4 (cuatro) años de funcionamiento ininterrumpidos, "El Entorno FACETVirtual cuenta actualmente con espacios virtuales de más de 100 cátedras distribuidas en las 20 carreras de esta unidad académica con un total de 2300 alumnos activos en el último año y más de 120 docentes. También hay grupos de investigación que utilizan las potencialidades del Entorno"[5].

Durante todos estos años la prioridad fue ofrecer soporte permanente y asegurar el correcto desempeño y la funcionalidad de las prestaciones. En este sentido, los recursos de hardware insuficientes dentro de la institución, cierto grado de inestabilidad en el servicio, la creciente demanda de aulas y el aumento constante del número de usuarios generó la necesidad de optar por un servidor dedicado externo a la institución. El proceso de Migración hacia un servidor con estas características se produjo el 25 de abril de 2012.

3 El proceso de Actualización de versión de Moodle

A finales del año 2013 el equipo responsable de Conectividad y Plataforma resolvió comenzar el camino que permitiera la Actualización de la Plataforma de la versión 1.9 a la versión 2.7. El hecho que hubiera varias versiones intermedias, no fue un tema menor y dio lugar a un proceso de estudio y planificación que continuó hasta mediados de 2014.

Al emprender esta tarea de planificación para llevar adelante la tarea del proceso de actualización de versión en el entorno FACETVirtual buscamos denodadamente información de referencia que orientara nuestro trabajo, ya que la distancia entre una versión y otra nos permitía esperar algunas dificultades. La tarea resultó infructuosa ya que no encontramos dato alguno en la Web que sirviera de modelo, solamente algunas explicaciones muy escuetas del proceso. Esto nos motivó a pensar que el tema tratado sería de interés, utilidad, y por supuesto, de aplicabilidad para todos aquellos administradores de plataforma que deban afrontar el mismo proceso; en especial conociendo la multiplicidad de sitios basados en Moodle en nuestro país[6] (953 sitios registrados en la página oficial) y en otras latitudes (52905 sitios registrados en el mundo). Lo que constituye un número no menor dado que se puede observar de las mismas estadísticas del sitio oficial de Moodle que en la actualidad más de un 30% de plataformas se encuentran corriendo en la versión 1.9[7]

Habiendo realizado el análisis correspondiente a los aspectos o que pudieran presentarse durante el proceso, una de las consideraciones más importantes para realizar la actualización era la definición de una fecha que permitiera cierta amplitud temporal para este proceso y resultara lo menos dificultosa para la tarea docente. Es

así que se acordó la actualización en el mes de enero de 2015, entendiéndola como la más apropiada dada la inactividad propia del período de vacaciones de verano. Así es como, durante el segundo semestre del 2014, el equipo se abocó a la elaboración de una serie de pruebas que aseguraran la implementación de la nueva versión sin ningún contratiempo (control de riesgos).
De los aspectos evaluados durante este proceso de prueba, surgió la necesidad de implementar una actualización del hardware del servidor que permitiera el soporte adecuado de la nueva versión de la plataforma.

3.1 Primera etapa

3.1.1 ***Actualización del hardware del Servidor*****:** En un primer momento, el hardware con que se contó para la implementación del LCMS era el siguiente:

- Intel Celeron Dual Core; 2 GB RAM / DDR2 / 667 Mhz; Linux CentOS 5; HD 1: 160 Gb SATA; HD 2: 160 Gb SATA

Consideramos importante presentar esta información para los administradores de sitios, de modo que puedan realizar las comparaciones y adecuaciones que crean oportunas.
En enero de 2015, y teniendo en cuenta las necesidades planteadas en la etapa de análisis, se cambió a un servidor con las siguientes características:

- CPU Intel® Core i3; 4 GB RAM / DDR 3 / 1333 Mhz; HD 1: 1 TB SATA; HD 2: 1 TB SATA [BACKUP]; Linux CentOS 6

3.1.2 ***Relevamiento de plataforma:*** Paralelamente se llevó a cabo el relevamiento de las aulas virtuales en uso efectivo, como también de las herramientas específicas de la plataforma Moodle que cada una de éstas utilizaba.

PRIMERA PAUTA: realizar el relevamiento previo de las aulas en uso efectivo, y herramientas específicas.

Esta tarea fue muy importante ya que permitió conocer cabalmente cuáles son los requerimientos a los que se debe poner mayor atención durante de la actualización de la misma.

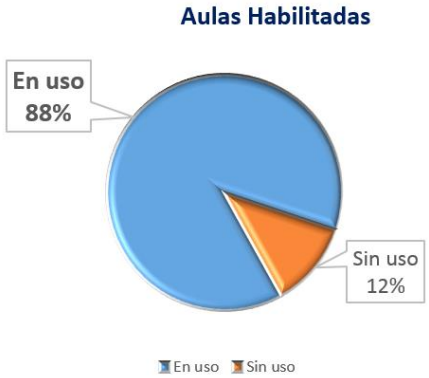


Gráfico 1: Aulas Habilitadas

Nuestro análisis concluyó en que todas las aulas evaluadas utilizaban la totalidad de los recursos de que posee la plataforma Moodle y que la mayor parte de ellas utilizaban sólo tres de las herramientas denominadas actividades.

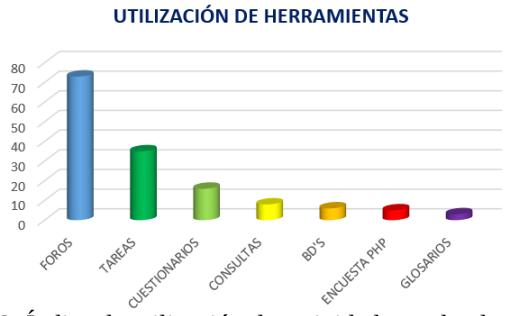


Grafico 2: Índice de utilización de actividades en la plataforma

En el gráfico 2 se presenta un resumen del uso de las actividades del Moodle por parte los docentes a cargo de las aulas.

3.2 Segunda Etapa: Pasos para la actualización

3.2.1 ***Respaldo del Sitio:*** el primer paso a llevar a cabo antes de realizar la actualización de la plataforma Moodle (como las buenas prácticas de administración lo aconsejan) consistió en realizar un *respaldo total* de lo que estaba en funcionamiento. Dicho respaldo incluyó la base de datos de Moodle, implementada en MySQL, mediante el uso de la herramienta mysqldump, el código ejecutable del software de plataforma (scripts en php) y los archivos de usuario que se encontraban en el directorio MoodleData.

SEGUNDA PAUTA: realizar el respaldo total del sitio en funcionamiento. Dicho respaldo incluyó la base de datos de Moodle, implementada en MySQL.

Es importante destacar en este punto que el proceso que vamos a describir se realizó primero en una instancia de Moodle instalada en un entorno de laboratorio, luego en un sitio de prueba habilitado para tal fin en un servidor remoto, a los fines de evaluar la incidencia del tráfico de Internet en la performance. A partir del 01 de enero de 2015 (fecha acordada e informada a los usuarios) se realizó la actualización en el sitio de producción.

El pasaje desde la versión 1.9.14 directa hacia la 2.7.4 no era una opción viable, conclusión a la que se llegó a partir del análisis de la documentación oficial y de la información obtenida por foristas en la web. La secuencia de versiones que dio resultados exitosos fue la siguiente:

- v1.9.14 (versión de partida)
- v1.9.19 (última versión estable de Moodle y recomendada para iniciar la actualización del sistema)
- v2.0.10
- v2.1.10
- v2.2.11
- v.2.7.4 (versión actual)

3.2.2 ***Preparación previa del sitio****:* Como es sabido (la documentación oficial de Moodle lo dice), se automatiza la ejecución de algunos trabajos mediante el uso del programadores de tareas del sistema operativo (en este caso implementado con cron), por lo tanto, en primer lugar, se debe suspender la ejecución de las mismas. Esto debe realizarse para evitar que durante el proceso se intente ejecutar las notificaciones de las actividades y tareas de mantenimiento propias de Moodle.
En segundo lugar, a partir de la plantilla standard de Moodle para su entorno gráfico, se impone la misma a todos los cursos, deshabilitando la opción que permite la elección de plantillas en las aulas desde: ***Administración/ Apariencia/ Ajustes de temas/ "Permitir estilos en los cursos"***.
Al realizar esta acción se ve afectada la estética del sitio, luego mejora al cargar la versión 1.9.19, esto no debe causar preocupación ya que se trata sólo de un paso intermedio.

TERCERA PAUTA: suspender la ejecución del CRON y deshabilitar la opción que permite elección de plantillas.

3.2.3 ***Eliminación y carga de carpetas***: Se procedió a eliminar todas las carpetas excepto MoodleData, y el archivo config.php – ya que éstas contienen toda la información para el funcionamiento de la plataforma. A continuación se realizó la carga de todas las carpetas y archivos de la versión de Moodle 1.9.19. Luego se accedió al sitio, vía web, para que comenzaran a correr la instalación de la versión cargada.
Una vez realizada la actualización correspondiente, se procedió a estandarizar la base de datos dejando de lado los módulos que incluían recursos, actividades y/o bloques que no estaban siendo utilizados en ése momento, por ejemplo wiki, lección, taller. Además se quitaron los módulos de Encuesta phpESP, Geogebra, LAMS, Journal y todos aquellos módulos, recursos y/o actividades cargados específicamente para la versión 1.9, esto se debe a que mucho de ellos no poseen soporte en la versiones superiores o en su defecto tienen versiones ya integradas que se encuentran bajo otra nomenclatura en las versiones sucesivas de Moodle (por ejemplo, Encuestaphp en la versión 1.9 es una actividad externa, en la versión 2.7 se encuentra como Encuesta pero no son compatibles las mismas). Esta tarea se llevó a cabo desde el bloque ***Administración/ Módulos/ Actividades/ Gestionar actividades***.
Finalmente se eliminaron todas las carpetas correspondientes a los módulos mencionados:

```
<raíz del sitio>/mod/questionnaire,
<raíz del sitio>/mod/sourvey,
<raíz del sitio>/mod/geogebra,… etc.
```

CUARTA PAUTA: actualizar desde una versión a la siguiente por vez y analizar correctamente los resultados.

Para la carga del Moodle versión 2.0.10, se procedió exactamente igual respecto de la eliminación y carga de carpetas, y se ingresó a la página para comenzar a correr todo el proceso de actualización.
3.2.4 ***Actualizar Cuestionarios***: al realizar la misma tarea para la carga de la versión 2.1.10 –es decir, el proceso de eliminación de carpetas y archivos exceptuando MoodleData y config.php–, fue necesario corregir las preguntas de los Cuestionarios, ya creados por los usuarios, puesto que en ésta versión se presentan

nuevas opciones en los tipos de preguntas ofrecidos, esto se realizó desde el Bloque Administración, opción ***Motor de ayuda actualización de preguntas*.**

QUINTA PAUTA: prestar especial atención a los cuestionarios y a su correcta actualización.

Se utilizó la opción "Reiniciar" los cuestionarios, que NO elimina las respuestas, sino que permite reconfigurarlas. Luego, desde la opción ***Listado de cuestionarios ya actualizados que pueden ser reiniciados,*** y operamos uno por uno los cuestionario; operación que efectivamente toma varios minutos. Una vez realizado esto debemos actualizar los cuestionarios, esto podemos hacerlo desde la opción "***Listado de cuestionarios aún por actualizar***".

Nota: los cuestionarios que quedaron abiertos, es decir con tentativas no finalizadas por los alumnos en cualquier curso o aula, NO se actualizan. Por lo que el proceso en cada uno de los casos es manual y consiste en reiniciar cada uno de los mismos. Cabe aclarar que antes de llevar adelante el reinicio de los mismos, procedimos a descargar una planilla con los resultados de estos cuestionarios, hacer un respaldo de estas respuestas y enviarlas a los docentes–administradores de cada curso.

3.2.5 ***Penúltima carga***: Actualización a la versión 2.2.11, se realizó mediante el mismo procedimiento ya indicado.

Cabe aclarar, que en este punto se planteó la posibilidad de pasar a una versión 2.2.2 (que es una de las versiones estables y recomendadas para llevar adelante este proceso de actualización) pero no pudo realizarse ya que esta versión presenta un inconveniente con respecto al editor de textos TINYMCE. Sucede que la versión del editor es anterior al que se encuentra corriendo en la versión de Moodle 2.1.10 previamente instalada, por lo que genera un conflicto al momento de actualizar y no permite llevar adelante el proceso.

SEXTA PAUTA: analizar el correcto funcionamiento de la versión instalada, especialmente en lo referido a los editores de texto.

Una vez verificado el normal funcionamiento de la plataforma, se continuó con la etapa de actualización a la versión 2.7.4 que sería la última versión a actualizar. Se permite hacer este salto entre la versión 2.2.11 y la 2.7.4 ya que la base de datos entre estas versiones no presenta mayores modificaciones como sí lo presenta en etapas anteriores a estas. Por lo que continuar escalando versión por versión sería una tarea que enlentecería el proceso de actualización.

3.2.6 ***Problemas presentados en la nueva versión***: Una vez llevado adelante nuevamente el proceso de eliminación y carga de carpetas (salvo MoodleData y config.php), y la actualización correspondiente para llegar a la última versión a actualizar – Moodle 2.7.4 –, nos encontramos con una nueva dificultad*:* ***la actividad Tarea se presentó con variantes***. Sin embargo, al revisar las actualizaciones de las 157 tareas creadas en la plataforma desde ***Administración de sitio Motor de ayuda para actualizar Tareas/Lista de tareas que no se han actualizado***, comprobamos que, se actualizan todas sin inconvenientes ya que ninguna tiene configuraciones no estándares.

El problema que se plantea es que dicha actividad en las versiones de Moodle anteriores a la 2.2 se denomina TAREA, en la versiones 2.2 a la 2.4 dicha actividad fue renombrada como TAREA 2.2 y se creó una nueva actividad TAREA que permite la realización de la misma de manera grupal, y ofrece a los docentes la posibilidad de

una única subida de archivos por parte de los integrantes de los grupos así como también admite que al calificar a un integrante esta misma calificación se replique en los demás participantes del grupo y no tener que replicar manualmente las mismas.
En la versión 2.7.4 el concepto de TAREAS se unifica, por lo que la actividad TAREA 2.2 aparece automáticamente deshabilitada, y por ende NO VISIBLE para los docentes y alumnos usuarios de la plataforma.

SEPTIMA PAUTA: tener en cuenta que a partir de la versión 2.2 la actividad TAREA experimenta cambios que son sustanciales a la hora de la actualización, pueden confundir u ocasionar pérdida de información importante.

Para solucionar este inconveniente, lo que hicimos a nivel administrador de Moodle es acceder desde el bloque **Administración del sitio/ Extensiones/ Módulos de actividades** y hacer visible el módulo de la actividad Tarea 2.2.
Una vez hecho esto fuimos nuevamente al bloque **Administración de sitio/ Motor de ayuda para actualizar tareas** e hicimos correr este motor para que actualizara las tareas que quedaron almacenadas, como del tipo Tarea 2.2.
Por último y para evitar inconvenientes a futuro, volvemos a deshabilitar la actividad TAREA 2.2 desde el bloque **Administración del sitio/ Extensiones/ Módulos de actividades.**

3.2.7 ***Finalmente,*** se procedió a la instalación de plantillas nuevas, y se realizó la configuración de la plantilla final a usar.
Luego continuamos con la instalación de un nuevo formato de curso denominado **one_topic**[8] – que permite visualizar los temas de un curso en formato de solapas –, para llevar adelante esto se cargó vía FTP en la carpeta **course/ format** la carpeta onetopic y luego debemos ir a **Administración de sitio/ Notificaciones** para que se cargue efectivamente en la plataforma.
Para completar las habilitaciones necesarias para la integración en el uso de repositorios externos, como Google Drive, se realiza desde **Administración de sitios/ Extensiones/ Repositorios/ Gestionar Repositorios,** desde aquí ponemos como visible Google drive**.** Tenemos 3 opciones *Deshabilitado*, *Activo y visible* y *Activo pero oculto***.** Seleccionamos Activo y visible. Esto nos llevó a una ventana de configuración para establecer ciertos parámetros[9]**.**

OCTAVA PAUTA: gestionar todas las habilitaciones necesarias para permitir el correcto funciona-miento de la última versión instalada.

3.3 Inconveniente que se planteó una vez puesta en producción:

La Mensajería interna presentó problemas a los docentes y estudiantes ya que se podían observar las fechas de los mensajes enviados y recibidos, pero no el contenido de los mismos. Se procedió a analizar la situación para concluir que esto sucedía debido a que la tabla *message_read* presenta diferencias de estructura entre la versión 1.9 y la 2.7.
Tras análisis y pruebas realizados en laboratorio sobre las bases de datos, se observa la existencia de un nuevo campo llamado *smallmessage* que es el que se hace visible desde la plataforma y que contiene un resumen del mensaje original que se encuentra en el campo *fullmessage*, el mismo se rellena con NULL en la conversión de una estructura a otra. Por ello se procede a copiar el texto cargado en *fullmessage* al

campo *smallmessage*. Cabe destacar que esta operación eventualmente puede truncar el texto a la longitud del campo *smallmessage.*
Para poder ejecutar la instrucción que llevará adelante esta tarea, nos debemos ubicar en el gestor de bases de datos (en nuestro caso phpmyadmin) y ejecutar la siguiente instrucción desde consola: `update message_read set small-message=fullmessage`
Con esto quedan nuevamente visibles los mensajes.

NOVENA PAUTA: poner a punto el servicio de mensajería interna, de modo que puedan visualizarse correctamente.

4 Perspectivas y mejoras

Este cambio hacia la versión 2.7 de Moodle implicó mejoras en el ámbito pedagógico a partir de nuevas herramientas que favorecen el proceso de enseñanza y aprendizaje con uso de TIC, como también aplicaciones y funcionalidades que propician la comunicación, interacción y colaboración (ej. multimedia, sincronización a repositorios externos, acceso más sencillo a Youtube, etc.). Desde el ámbito Tecnológico, podemos mencionar: mejor navegabilidad, mejor integración con herramientas Web 2.0, fácil carga de archivos, compatibilidad con los navegadores más usados.
Finalmente, desde el punto de vista del ámbito Informático, mejoró respecto a disponibilidad de soporte para la nueva versión Moodle 2.7, actualización permanente de acuerdo a los estándares TIC y mayor seguridad en toda la plataforma.

DECIMA Y ULTIMA PAUTA GENERAL: comprender y difundir la enorme importancia de la actualización del Moodle para optimizar el uso del Entorno Virtual, especialmente desde el punto de vista metodológico.

Las **principales novedades para los docentes** se centran en las posibilidades de:

- Seleccionar plantilla personalizada: diversos tipos de plantillas para el curso de acuerdo a los requerimientos pedagógicos (ej. Inicial, Básica e Intermedia).
- Mejorar la gestión de los archivos, por ejemplo, subir los archivos arrastrándolos (drag and drop), y también, si el docente necesita un archivo que se encuentra en otra aula que es de su propiedad, poder accederlo directamente sin necesidad de volver a cargarlo.
- Facilitar la subida de tareas en grupo, de manera de calificar a un alumno y que la calificación automáticamente se replique a los otros integrantes, así como también da la posibilidad de limitarla para que una vez que 1 alumno suba el trabajo los demás integrantes no puedan subir otros trabajos.
- Contar con un nuevo sistema de matriculación de usuarios; bloqueo por cupo de alumnos, implementación de rango de fechas para la matriculación de alumnos, matriculación directa en grupos de trabajo que permiten una mayor y más rápida coordinación de los mismos.
- Integrar (o vincular) con repositorios externos (google drive, dropbox, etc)

5 Conclusiones y Recomendaciones

La actualización de la plataforma fue un proceso laborioso (y puede considerarse inevitable) pero con resultados altamente satisfactorios. Moodle, como software libre, a través de los años y de sucesivas versiones enriquecidas por la programación de miles de informáticos en forma global, ha llegado a igualar y, en un gran número de funcionalidades, superar las prestaciones de las más afamadas plataformas comerciales.

Con nuestro aporte en el presentar este trabajo, pretendemos orientar a los responsables de plataformas virtuales Moodle que necesiten pasar por los procesos de actualización. Entre las recomendaciones generales más importantes podemos mencionar, básicamente:

- No dejar pasar tanto tiempo entre versión y versión para realizar procesos de actualización.
- Realizar pruebas en un ámbito de laboratorio y otra prueba en servidor remoto sobre una copia de respaldo del sitio en funcionamiento (no visible para el resto de los usuarios para evitar confusiones y preferentemente en modo mantenimiento).
- Realizar la completa y correcta documentación de cada paso requerido en el proceso de actualización, a medida que se hacen las pruebas, y para que sea utilizado luego en el proceso final.

Es importante remarcar nuevamente que, cualquier proceso de actualización de la plataforma de software que soporta un Entorno, se enmarca no sólo en la necesidad de acompañar las innovaciones tecnológicas, que se suceden a través del tiempo, sino y especialmente, en las posibilidades metodológicas que esta actualización ofrece en relación a optimizar los procesos de aprendizaje.

Referencias

1. "Diferencias entre CMS, LMS, LCMS y EVA (e-learning)", http://e-ducacion.info/e-learning/diferencias-entre-cms-lms-lcms-y-eva-elearning/
2. http://www.educdoscero.com/2009/01/moodle-la-plataforma-lms-libre-ms-usada.html
3. Ros, Iker. "Moodle, la plataforma para la enseñanza y organización escolar", *Ikastorratza, e-Revista de didáctica, 2*, 3-12. (2008) Recuperado de: http://www.ehu.es/ikastorratza/ 2_alea/moodle.pdf.
4. Torres Auad, L.; García, R. - "Hacia una deconstrucción del paradigma educativo tradicional: la modalidad virtual en la FACET", revista CET nro. 33, junio 2011. ISSN 1668-8910. (2011)
5. Guzmán, M.F.; Torres Auad, L.F.; Auvieux, N. - "Implementación de la modalidad virtual en la FACET", Jornadas 2014 en Ingeniería Eléctrica, Electrónica y Computación", ISBN 978-987-33-7036-6. (2014)
6. https://moodle.net/sites
7. https://moodle.net/stats/
8. https://moodle.org/plugins/view/format_onetopic
9. https://docs.moodle.org/26/en/Google_OAuth_2.0_setup

Get Closer to Activate It! An Educational Tool for People with Multiple Disabilities

Filip Bojanovic[1] , Cristina Manresa-Yee[1], Maria José Abásolo[2,3], Martin Larrea[4] and Cecilia Sanz[2]

[1] University of Balearic Islands, Palma, Spain
[2] III LIDI, Facultad de Informática, Universidad Nacional de La Plata, La Plata, Argentina
[3] Comisión de Investigaciones Cientificas de la Pcia. de Bs. As, La Plata, Argentina
[4] Universidad Nacional del Sur, Bahía Blanca, Argentina
filipetus@yahoo.com, cristina.manresa@uib.es, mjabasolo@lidi.info.unlp.edu.ar, mll@cs.uns.edu.ar, csanz@lidi.info.unlp.edu.ar

Abstract. This paper presents an interactive system which offers multimodal feedback to users with severe to profound disabilities to improve their relationship with the environment allowing them to control it. The system is based on the information provided by a distance sensor connected to an Arduino board. The distance information is based on the position of the user's hand, and it is then passed to six different action/reaction applications which offer motivation and engage users to train intentional motor movements of their upper body limbs and to improve their relationship with the environment.

Keywords: human-computer interaction; multimodal system; severe and profound impairments; motivational feedback, Arduino

1 Introduction

People with multiple disabilities have an intellectual disability and additionally, they may have other impairments such as visual, hearing or movement impairments. Further, they can present difficulties in understanding, communication or behaviour. Consequently, they have limited functionality and slow development processes and they need permanent supports in all areas of development [1].

People in this group may often present limited interactions with the surrounding world due to their physical and cognitive skills. Technology can play an important role for them both to increase the chances of relationship with the surrounding environment and to help them develop their independence.

In this work, we present an educational support, an interactive system addressed to users with these conditions that allows them to acquire a level of autonomy in interacting with the immediate environment by offering them the control over the environmental stimulation. We are working closely with a center for users with cerebral palsy, where there are people with this profile attending. In this case, therapists wanted to:

- Increase the intentional movements of their upper body limbs.

- Reduce their isolation.
- Control the interaction with the surrounding world.
- Achieve their active participation in the task.
- Suppress the self-stimulation by offering them external stimulation.

In previous works, in order to approach these aims, we designed and evaluated a motion-based interface that used computer vision techniques to detect the body movements. The user was captured by a webcam and the system tracked a colored band on the wrist of the user, which in turn triggered meaningful outcomes from the system (music, images, videos and so on) [2]. Then, we included a vibrotactile interface attach to the user's arm to encourage its motion [3, 4]. In these works, the controlled movement was lifting up the arm, but another interesting movement to work is the extension of the arm to get objects on a table or from a person. One solution would be to place a webcam perpendicular to the extension of the arm or to try and detect the different size of hand when getting closer or farther to a webcam facing the user [5]. The use of a sensor such as Microsoft Kinect in this specific context is not effective, because many users use wheelchairs or prams, and the computation of the depth of the arm is not effective as the background is too close [2].

There are other existing works which offer users control of the environmental stimulation by means of diverse devices such as microswitches [6], tangible user interfaces (TUI) [7] or gesture based systems [8].

In this work, we present a system that is comprised by a distance sensor and a set of action/reaction applications which trigger motivational audiovisual effects to engage the users and achieve their active participation in the proposed activities. The interactive system is addressed to users with physical and cognitive limitations and it aims at allowing users to acquire a level of autonomy in controlling the environment stimulation.

The paper is structured as follows: Section 2 describes the system, both hardware and software. The evaluation of the system is presented in Section 3. Finally, in the last section, we summarize the findings and discuss their implications.

2 The system

The interactive system is based on an Arduino board and an ultrasonic distance sensor that enable us to control the distance of the user's hand in order to trigger different feedback from the multimodal applications depending on the distance.

2.1 The hardware prototype

For this project in particular, we used the Arduino MEGA board [9] and an Ultrasonic Ranging Module HC-SR04 [10]. The HC-SR04 provides a 2cm to 400 cm non-contact measurement function and the ranging accuracy can reach to 3mm (See Fig. 1). This module includes ultrasonic transmitters, a receiver and a control circuit. The wires connected to the sensor are a 5 Volt power supply, 0 Volt ground, a trigger

pulse input and an echo pulse output. The ultrasonic sensor uses a sonar to measure the distance to an object. This sensor has two parts: echo and trig. One emits ultrasonic waves, while the other receives them. Then, the distance is calculated by timing how long it takes for an ultrasonic wave sent out by the emitter to bounce to the object and come back to the receiver.

Initially, to adapt it to our target group who does not count with a wide range of physical mobility, we decided to limit the movement range from 0 to 50 cm.

Fig. 1. Ultrasonic Ranging Module HC-SR04 [10]

2.2 The multimodal applications

The interactive system comprises the following six applications that trigger visual and audio effects:

- Volume control
- Growing rays
- Moving balls (horizontally)
- Moving squares (vertically)
- Changing colors
- Zoom in-out image

These applications try to engage the user in the use of the system to work educational aims. The applications are selected and executed from a common home window.

The multimodal applications were developed using an open source programming language and integrated development environment, called Processing [11] which counts with a convenient library to handle easily the communication with the Arduino board. Further, we used Minim as an audio library [12], which is a Java audio library designed to be used with Processing.

The applications have to include motivational material for the users because it is an important factor to engage users to use the system and to help them to concentrate on the task [13, 14]. Therefore, applications have to be configurable regarding the audio or images to play or show. Further, the applications contain dynamic and colorful shapes, to get the user's attention and focus.

Volume control

The Volume Control application allows for users to increase or decrease the volume of an audio by moving closer or further the hand towards the distance sensor. This

Volume Control application is addressed to users with vision impairments (as the visual content is not relevant), but also users who have hearing impairments can benefit of this application as this exercise can help them to practice and develop their listening.

The therapist can choose the file that will be used in the application to adapt the audio to the preferences of the user. The file can be in mp3, wav, aiff, au or snd, as these are very popular formats. In this case, the visual shown on the screen is not important, but we can set an image, so the background does not stay blank.

By moving the hand towards the ultrasonic sensor, the volume will change from low to high, as we reward more positively the effort of extending the arm which requires a greater effort from the user. If the arm is further than 50 cm, the volume will remain at the lowest volume level.

Moving Balls (horizontal)

The purpose of this application is for children to play with speed. There are five balls that are starting from different points on the screen and they just move horizontally increasing or decreasing their speed.

Users should notice that the colorful balls are going faster or slower from the left side to the right side of the screen as they move closer or further to the distance sensor.

Growing Rays

This application presents colorful and dynamic visuals on the screen depending on the distance of the user. The growing rays' application draws colorful multicolor lines in a form of a circle, with shorter or longer length depending on the distance of the user's hand: when the user is closer to the sensor, the application draws shorter lines than when the hand is moving away from it (See Fig. 2).

When the circle is completed, the screen refreshes and the previous set of lines is removed, so we can have new circle of lines without overlays.

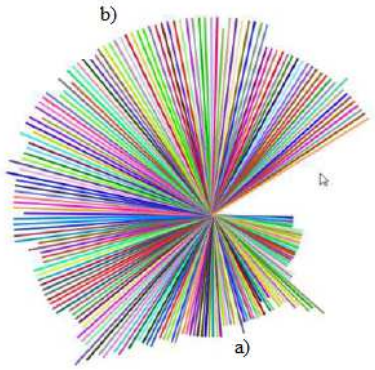


Fig. 2. Growing Rays. a) the hand is closer to the sensor so rays are shorter. b) the hand is further form the sensor, so rays are longer

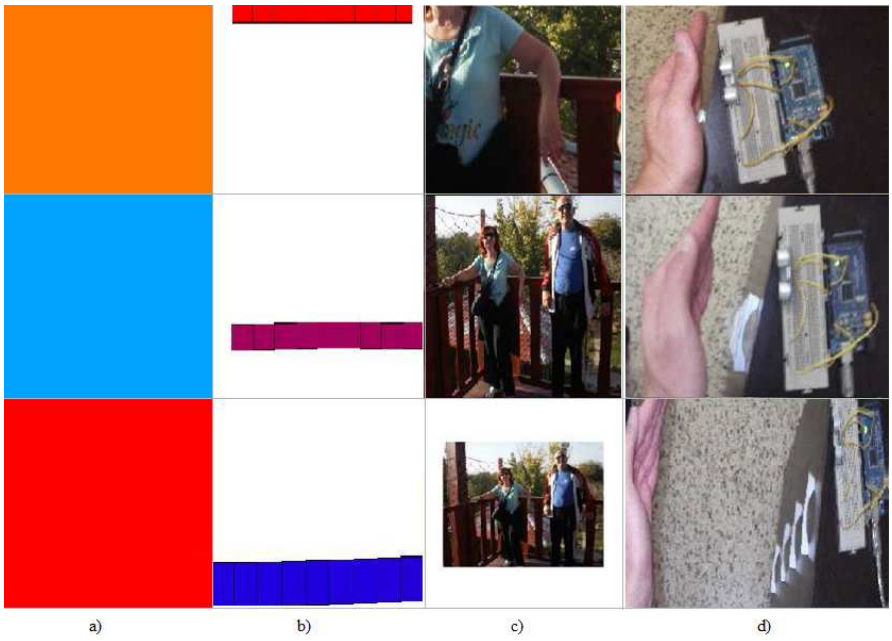


Fig. 3. a) Changing colors. b) Moving squares c) Zoom in/out image d) the hand's distance regarding the sensor.

Changing Colors

When children recognize colors they are noticing, matching, sorting, and labeling the characteristics of objects in the world around them. Recognizing colors is a skill that is often difficult for children because it is necessary to know the color words, as well as to identify the abstract characteristic of color [15].

The application with changing colors can help children to recognize different colors. By moving closer to and further away the distance sensor, different colors go appearing (see Fig. 3a and 3d). The therapist can play with the user making them stop in a certain color or just let them enjoy the different visuals.

Moving Squares (vertical)

The application is similar to the Moving Balls application, but the movement is done in the vertical axis. Ten squares start in the upper zone of the screen and while the user moves his or her hand away from the sensor, the squares go growing and moving down changing their color (see Fig. 3b and 3d). The therapist should help the user notice the colors and the shape as well as the position of the squares.

Zoom in/out image

In the application with Zoom in/out Image, the accent is put on the movement of the image that is shown on the screen. When the user is near the sensor the image is

zoomed in and as he or she is moving away, the image goes zooming out (see Fig. 3c and 3d).

The use of images as motivational material can make users more willing to actively participate in the task, therefore, the image can be configurable. Images should show people related to the user or his or her preferences: family, friends, class mates, animals or local environment.

3 Evaluation

We wanted to test the robustness and usability of the system before working with the end-users, as a non-usable system could repel users from future uses. Usability is defined as "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use" [16].

Five users (three females, two male) without disabilities were selected whose ages ranged from 24 to 25 and they were informed about the purpose of the study. Sessions were video-recorded for their posterior analysis.

The procedure of the test was first to introduce all exercises, then execute one by one each application and carry out simple tasks, finally, users assessed the functionality of the application from 0 to 5. Tasks are very simple, but they represent the kind of tasks therapists would do later with the end-users. Probably, some of the users will not be able to communicate orally, but they may have signals or sounds for Yes and No, so, the therapist should guide the session asking simple questions in a way users can understand them.

Table 1. Evaluation tasks

Heading level	Example
Volume control	T1. Describe what is happening with the audio when moving back and forth the hand to see how easy is to perceive the volume's difference T2. Find the distance where the volume is not changing
Moving balls	T3. Describe what is happening on the screen with the balls T4. Observe which ball has the shortest and which one has the longest moving path on the screen
Growing rays	T5. Describe how the lines are drawing on the screen by moving from and to the sensor
Changing colours	T6. Describe five different colors that are seen on the screen T7. Describe which color is the last one and stop in the correct position
Moving squares	T8. Wait at one position and get the squares stopping or moving
Zoom in/out image	T9. Describe what is happening on the screen with the image T10. Stop in one position where the image is not moving

Regarding the effectiveness of the applications, except for few errors in T8, all tasks were done successfully. Errors can be due to the sensor's sensibility. The arms movement should be smooth to get better results. As these users may have abrupt movements, bigger regions should be assigned for a change in the feedback.

When asking the users about the functionality of the activities and taking into account the users' target group, they all highlighted the importance of these tasks, by answering with a 4 or 5 in the questionnaire.

4 Discussion and Conclusions

In this work we have describe an interactive system that serves as an educational tool for users with severe impairments. The system is comprised by a distance sensor and a set of action/reaction applications which trigger motivational audiovisual effects to engage the users and achieve his or her active participation in the activities. These applications are dynamic and mix color, motion, and different shapes. Further, the system offers control to the user in an implicit manner. The user is not aware of the presence of a computer and they are interacting with the system in a natural way by means of the body motion.

We have presented a set of tasks as an example of what can the therapist do, but depending on the user, the therapist can give other instructions (e.g. in the Changing color application, the user can be instructed to stop in a certain color).

We remark considerations taken into account in the design of the applications:

- The positive feedback has to be very motivational and meaningful for the users.
- The user has to be aware of the feedback returned by the system.
- The system has to be robust and respond promptly to the user's actions.
- The system has to be configurable.

Future work will include usability tests with end-users to validate the system. Further, other activities can be implemented to enrich the set of applications. It would be also interesting to see in future experiments, when the user is competent with the system, how he or she works alone with the system without the help of a therapist.

Acknowledgments

This work is supported partially by the project TIN12-35427 granted by the Spanish MINECO, FEDER funding and FRIVIG A1/037910/11 granted by MAEC-AECID.

References

1. Soro-Camats, E., Rosell, C., Basil, C.: El alumnado con pluridiscapacidad: características, evaluación y necesidades educativas. In: Soro-Camats, E., Basil, C., and Rosell, C. (eds.) Pluridiscapacidad y contextos de intervención. pp. 5–32. Universitat de Barcelona (2012).
2. Manresa Yee, C., Muntaner, J.J., Arellano, D.: A Motion-based Interface to Control Environmental Stimulation for Children with Severe to Profound Disabilities. In: CHI '13 Extended Abstracts on Human Factors in Computing Systems. pp. 7–12. ACM, New York, NY, USA (2013).

3. Manresa-Yee, C., Morrison, A., Larsen, J. V, Varona, J.: A Vibrotactile Interface to Motivate Movement for Children with Severe to Profound Disabilities. In: Proceedings of the XV International Conference on Human Computer Interaction. pp. 10:1–10:4. ACM, New York, NY, USA (2014).
4. Manresa-Yee, C., Morrison, A., Muntaner, J.J.: First Insights with a Vibrotactile Interface for Children with Multiple Disabilities. In: Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems. pp. 905–910. ACM, New York, NY, USA (2015).
5. Manresa-Yee, C., Varona, J., Mas, R., Perales, F.J.: Hand Tracking and Gesture Recognition for Human-Computer Interaction . Electron. Lett. Comput. Vis. Image Anal. 5, 96–104 (2005).
6. Lancioni, G.E., Singh, N.N., O'Reilly, M.F., Oliva, D.: Using a hand-tap response with a vibration microswitch with students with multiple disabilities. Behav. Cogn. Psychother. 30, 237–241 (2002).
7. Lancioni, G.E., O'Reilly, M.F., Singh, N.N., Sigafoos, J., Oliva, D., Severini, L.: Enabling two persons with multiple disabilities to access environmental stimuli and ask for social contact through microswitches and a VOCA. Res. Dev. Disabil. 29, 21–28 (2008).
8. Mauri, C., Solanas, A., Granollers, T.: Nonformal Interactive Therapeutic Multisensory Environment for People With Cerebral Palsy. Int. J. Hum. Comput. Interact. 28, 202–2012 (2012).
9. Arduino, http://www.arduino.cc/.
10. Micropic: Ultrasonic Ranging Module HC-SR04, http://www.micropik.com/PDF/HCSR04.pdf.
11. Processing, https://processing.org/.
12. Minim, https://github.com/ddf/Minim.
13. Manresa-Yee, C., Ponsa, P., Salinas, I., Perales, F.J., Negre, F., Varona, J.: Observing the use of an input device for rehabilitation purposes. Behav. Inf. Technol. . 33, 271–282 (2014).
14. Hattie J., Timperley, H.: The power of feedback. Rev. Educ. Res. 77, 81–112 (2007).
15. O'Hanlon, C.G., Roberson, D.: Learning in context: linguistic and attentional constraints on children's color term learning. J Exp Child Psychol. 94, 275–300 (2006).
16. ISO 9241-210:2010: Ergonomics of human-system interaction - Part 210: human-centred design for interactive systems. Int. Organ. Stand. (2010).

Integración de Indicadores Sociométricos de Dinámicas Grupales en Ambientes Virtuales de Trabajo Colaborativo Orientados a Intervenciones Didácticas

Norberto Charczuk, Darío Rodríguez, Ramón García-Martinez

Laboratorio de Investigación y Desarrollo en Espacios Virtuales de Trabajo
Grupo de Investigación en Sistemas de Información
Departamento de Desarrollo Productivo y Tecnológico. Universidad Nacional de Lanús
http://sistemas.unla.edu.ar/sistemas/gisi/LIDEVT.htm
norcknet@gmail.com, darodriguez@unla.edu.ar, rgm1960@yahoo.com

Resumen. En el marco de las tecnologías aplicadas al ámbito educativo y en el caso de los EVTC (entornos virtuales de trabajos colaborativos) donde la presencialidad es reemplazada por la participación de los integrantes en la realización de una tarea. La sociometría (SNA – Social Network Analysis) es un instrumento de aproximación para la identificación de los distintos roles de los participantes, este trabajo propone una integración de dos métodos previamente analizados y evaluados con el fin de una correcta intervención didáctica sobre los participantes de los EVTC por el tutor/docente en el logro de los objetivos planteados.

Palabras claves. Tecnologías aplicadas – ámbito educativo - entornos virtuales de trabajos colaborativos – sociometría - SNA (Social Network Analysis) – integración – métodos – intervención didáctica – logro de objetivos

1. Introducción

Los nuevos entornos de enseñanaza-aprendizaje presentan nuevos desafíos a los docentes, las nuevas tecnologías implementadas conjuntamente con modelos pedagógicos no tradicionales requieren otro grado de observación. Dentro de un entorno virtual de trabajo colaborativo (EVTC) en donde cada participante se encuentra frente a distintas modalidades de trabajo (sincrónicas/asincrónicas), cada una de ellas mediadas por tecnologías, el presente trabajo es una propuesta de un indicador que resulta de la interacción de dos factores que hacen a la interpretación de resultados obtenidos mediante distintas técnicas sociométricas, con el fin de detectar los roles dentro de un grupo en un EVTC (entorno virtual de trabajo colaborativo) con el fin de una correcta intervención didáctica por parte del docente/tutor. Dentro de este contexto, se presenta el marco teórico con detalle de observación de grupos, características de los distintos roles y adaptación de técnicas sociométricas como herramienta de observación de las interacciones (sección 2); se delimita el problema (sección 3); se propone una solución basada en la integración de las métricas del campo sociométrico a dinámicas sociales en grupos virtuales y un procedimiento de cálculo de los indicadores definidos en la adaptación (sección 4); se ilustra la aplicación de las definiciones y técnicas propuestas en una prueba de concepto

(sección 5); y se formulan conclusiones preliminares sobre esta línea de trabajo (sección 6).

2. Marco Teórico

En esta sección se presenta el marco teórico del trabajo que considera la integración de dos propuestas de métricas dentro de los EVTC, la forma de interactuar de los participantes de grupos de trabajo (sección 2.1), el concepto y características de rol en grupos de trabajo (sección 2.2) y la sociometría como herramienta de observación de roles (sección 2.3).

2.1. La Interacción de los Participantes

En los EVTC, al estar mediados por los distintos artefactos tecnológicos, cobra mayor presencia el entorno. La inteligencia del sistema postula un nuevo principio de conocimiento que no es el holismo. Éste solo es posible a condición de concebir el sistema no solo cómo un término general, sino como un término genérico o generador, es decir un nuevo paradigna donde cobra relevancia el conjunto de relaciones [Morin, 1977]. Con el enfoque de sistema, tratamos con un concepto de tres caras, el sistema, las interacciones y la organización.
En este trabajo profundizamos el análisis de las interacciones. Las mismas se han medido bajo dos intrumentos previamente estudiados, el tiempo de respuesta que tiene cada participante y la cantidad de mensajes enviados/recibidos en un lapso de tiempo (densidad de comunicación) dentro de un grupo. La forma de detectar la interacción de los participantes de todo grupo en EVTC genera varias posturas, la comunicación asincrónica es todo un desafío "…las respuestas no son tan simples. En tanto que la tecnología ofrece interacción professor-estudiante y estudiante-estudiante, no es tan personalizada o casual como puede suceder en un salon normal. Así que si son críticos altos niveles de interactividad o trabajo en equipos, confiar solamente en una clase "webizada" puede no brindarles todo lo que están buscando. Además, el número de estudiantes que pueden participar en una clase en línea y aún tener algún nivel de actividad es a menudo menor que en la situación cara-a-cara.."[Rosenberg,2001].
Dentro de los elementos que involucran la interacción, no solamente a los EVTC, que son los que atañe a nuestro trabajo, se encuentran los orientados en la educación a distancia, en donde "…el desafío permanente de la educación a distancia implica recorder el sentido politico con el que nació la oferta, reconocer si los soportes tecnológicos que se están utilizando son los más adecuados para el desarrollo de los contenidos, identificando la respuesta de la enseñanza y la concepción del aprendizaje que subyace y analizar de que manera se consideran los desafíos de la "distancia" entre los alumnus y los docentes, y de los alumnos entre sí…"[Litwin, 2000]

2.2. Concepto y Características de Rol en Grupos de Trabajo

Dentro del grupo, cada individuo ocupa una posición (status). A cada posición le corresponde una función, es decir, el cumplimiento de una conducta (rol) determinada por normas. [Pasel & Asborno, 1993]. Dentro de los elementos que hacen a la tarea en cualquier ámbito, pero específicamente dentro del análisis a los que competen en los

trabajos colaborativos mediados por la tecnología de la información, se debe tener presente el rol que presenta cada integrante en cada uno de ellos "al status alumnos corresponde una serie de roles. Los objetivos se lograrán mejor si los roles que es necesario cumplir están bien definidos; cada miembro debe comprender su tarea, aceptarla y percibir la relación entre la función que él cumple y la que realizan sus compañeros. [Cols, 1977]. Los miembros de un grupo se comunican con regularidad, se toman en cuenta mutuamente y tienen conciencia, en distinto grado, de poseer características comunes que establecen su diferencia con los otros. Algunos grupos tienen un alto grado de cohesión, mientras que en otros el sentimiento de pertenencia es más débil [Cohen y Bailey, 1997; Etzkowitz, 2003].
En los EVTC es de alto impacto el rol de cada participante, donde si el grupo de pertenencia es el resultado de un conjunto de factores, por ejemplo, si los mismos se conocen de materias o actividades dentro del ámbito académico, o en su defecto, nunca han compartido alguna actividad previa. Estas situaciones potencian el rol de cada uno de los participantes.

2.3. La Sociometría como Herramienta de Observación de Roles

Una correcta observación de la dinámica grupal suele ser un elemento insustituible [Meddley y Mitzel, 1963; Rosenshine, 1973; 1986] pero dado que en el trabajo colaborativo, mediado por la tecnología, el elemento observable se ha de ponderar y registrar por otros indicadores que han de representar los elementos que hacen al entorno virtual. La sociometría es la técnica que investiga el índice de centralismo, canales de comunicación y vivencias dentro de un grupo humano. Fue creada por el psiquiatra Jacob Levy Moreno (psiquiatra americano nacido en Bucarest, Rumania, 1889-1974). Entre las técnicas desarrolladas en esta disciplina se encuentran los test sociométricos y de percepción sociométrica, cuyo objetivo es el estudio sistemático de las propiedades psicológicas de la población, poniendo en acción una técnica experimental fundada sobre métodos cuantitativos. Contemporáneos a Levy Moreno, se encuentran dos sociólogos Emile Durkheim y Gabriel Tarde, en donde la noción de representación colectiva es fundamental en Durkheim, es decir una sociología típicamente molar, una sociología de gran conjunto. Pero los grandes conjuntos no nacen ya hechos, es decir que debajo de los grandes conjuntos hay corrientes de imitación (Tarde) y luego de invención. No se trata de una sociología de la representación, se trata de hacer una sociología de lo que hay debajo de la representación [Deleuze 1986]. Debajo de la representación se encuentran los roles de los participantes, de ahí la influencia y el poder que puede llegar a ejercer una persona sobre el grupo. El poder no es una propiedad, en un grupo social donde los roles no han sido asignados, el poder se ejerce, cómo una estrategia. De ahí la necesidad de contar con métricas para una adecuada intervención didáctica sobre los participantes y evitar los roles de marginación o rechazo por parte de los participantes ante la concreción de una actividad propuesta.

3. Delimitación del Problema

Las técnicas sociométricas se utilizan para descubrir cuáles son los actores de la dinámica de un grupo de trabajo, lideres, marginados, rechazados, normales, entre otros. Es un problema abierto poder determinar este tipo de roles en grupos que

interactúan mediados por tecnología informática, sean grupos áulicos o de trabajo mediados por un entorno virtual. Ante éstos desafíos nos encontramos con dos elementos vinculantes, el determinado por la "visibilidad" de cada integrante. Dicho concepto permite dos miradas, la física , el observer a los participantes por un lado y la otra que es product de las interacciones de los participantes.
En los entornos virtuales el vínculo socializador se da a través de las interacciones [Rodríguez y García-Martinez, 2014]. Luego, las relaciones todo-partes y partes-todo quedan determinadas por el flujo de las interacciones, esto es consistente con el hecho que los sistemas no se constituyen de "partes" o de "constituyentes", sino de acciones entre unidades complejas, constituidas a su vez por interacciones [Morin, 1980]. Con base en técnicas sociométricas como el test de Bastín [1966] y Leibovich 1980], en [Charczuk et al., 2013,2014] se ha planteado el interés de desarrollar e integrar técnicas sociométricas para grupos de desarrollo de proyectos mediado por EVTC.

4. Solución Propuesta

La integración de los indicadores de las interacciones de los participantes se encuentra enmarcada por dos propuestas de adaptación de técnicas sociométricas, que se unifican para generar un nuevo indicador que permita mayor certeza en la toma de decision al abordar una intervención didáctica.
En esta sección se presentan definiciones a utilizar para la reformulación de técnicas de diagnóstico sociométrico de dinámicas grupales para usar en ambientes de trabajo colaborativo (sección 4.1) y se presenta el proceso de cálculo de los indicadores de diagnóstico/pertenencia en la dinámica grupal (sección 4.2).

4.1. Definiciones

En esta sección se presentan definiciones de conceptos sociométricos, redefinidas en trabajos previos [Charczuk at al., 2013; 2014], para permitir su aplicación en espacios virtuales de trabajo colaborativo.

4.1.1. Densidad de comunicación (DC)
Es la cantidad de mensajes enviados por unidad de tiempo que le envía el sujeto Si al sujeto Sj

4.1.2. Unidad de Interacción
Es el tiempo de respuesta de los mensajes que le envía el sujeto Si al sujeto Sj.

4.1.2.1. Matriz de Interacciones (MA)
Asignando un valor de ponderación al tiempo t (que se muestra en la Tabla I), se puede construir la matriz MA de interacciones del grupo de trabajo, siendo MA(i,j) la ponderación de tiempos entre mensajes enviados / contestados por el sujeto Si al sujeto Sj.

4.1.2.2. Matriz de Interacciones (MB)
Se puede construir la matriz MB de interacciones del grupo de trabajo, siendo MB(i,j) la cantidad de mensajes enviados / contestados por el sujeto Si al sujeto Sj.

4.1.3. Matriz de Densidad de Comunicaciones Sujeto-Grupo (MDCSG)
Se construye con base en la Matriz MB y contiene en la posición MDCSG(i,j) la densidad que tiene el sujeto Si respecto de cada miembro del grupo.

4.1.4. Matriz Integración (MI)
Se construye con los valores obtenidos de multiplicar el elemento de la matriz MA(i,j) por el elemento de la matriz MDCSG(i,j)

4.1.5. Índice de autocolaboración (IACi)
Es el valor representativo del grado de colaboración que tiene el sujeto Si respecto de sí con relación al grupo. Este indice se calcula mediante la expresión (Donde: x representa el valor de ponderación, n representa la frecuencia relativa del valor x, N representa la frecuencia absoluta):

$$IAC = \frac{\sum_{i=1}^{n} xi.ni}{N}$$

4.1.6. Índice de heterocolaboración IHCi)
Es un valor representativo del grado de colaboración que tiene el grupo respecto al sujeto Si. Este índice se calcula mediante la expresión (x, n, N; iden IACi):

$$IHC - \frac{\sum_{i=1}^{n} xi.ni}{N}$$

4.1.7. Índice de realidad-colaborativa (IRCi)
Es un valor que representa la diferencia entre los índices anteriores y permite determinar el grado de colaboración sujeto-grupo/grupo-sujeto. Este índice se calcula mediante la expresión: IRCi=10.(IACi – IHCi). Dicho valor tiene el coeficiente 10 que multiplica la diferencia entre índices, debido a que la diferencia entre valores puede resultar un valor que por redondeo no resulte lo suficientemente significativo. Se propone que este índice se interprete según la siguiente regla de decisión:

IRCi > 0 El sujeto Si se sobre-estima en referencia a lo estimado por el grupo en el proceso colaborativo.
IRCi = 0 La valoración del sujeto Si coincide con la que tiene el grupo de él.
IRCi < 0 El sujeto Si se considera por debajo en referencia a lo estimado por el grupo respecto de la colaboración.

4.1.8. Índice de realidad absoluto (IRCA)
Es el valor absoluto del índice de realidad-colaborativa de un sujeto y la de un grupo se obtiene por acumulación del IRCA de los sujetos. El valor medio del índice de realidad absoluto (MIRCA), indica la valoración global del grupo, si el mismo es cercano a cero, es un grupo muy maduro en lo referente a su grado colaborativo, cuanto más alejado del valor nulo indica lo contrario. Se determina por la siguiente expresión (donde L representa la cantidad de sujetos del grupo):

$$V_{MIRCA} = \frac{\sum_{i=1}^{n} IRCAi}{L}$$

4.1.9. Escalón de segmentación (ES)
Es el valor obtenido de la división de la resta del mayor valor de Densidad de Comunicación menos el menor por 5 (genera 5 quintiles)

4.1.10 . Grado de Colaboración Sujeto-Grupo (GCSG)
El GCSG del sujeto Si permiten observar mediante el agrupamiento de las cantidades de interacciones entre el sujeto Si y el resto de los miembros del grupo cual es la conducta de Si en interacción con el grupo. Se denotara con GCSG(Si,P) la Cantidad

de Ponderaciones con valor P que tiene el sujeto Si. La valoración de GCSG del sujeto Si estará dada por el vector:

(aceptadori, normali, rechazadori)

Donde:

Aceptador(i) = GCSG(Si,5) + GCSG(Si,4)
normali = GCSG(Si,3) + GCSG(Si,2)
rechazadori = GCSG(Si,1) + GCSG(Si,0)

Las reglas de decisión que se proponen en este trabajo son:

Si es aceptador del grupo si
aceptadori >= normali + rechazadori
Si es normal del grupo si
normali >= aceptadori + rechazadori
Si es rechazador del grupo si
rechazadori >= normali + aceptadori

En caso que la ponderación se encuentre con el mismo valor en dos de los tres componentes (Aceptador, Normal, Rechazador), serán compartidos los atributos de la ponderación.

4.1.11. Grado de Colaboración Grupo-Sujeto (GCGS)

El GCGS del Grupo respecto al sujeto Si permite observar mediante el agrupamiento de las cantidades de interacciones entre el resto de los miembros del grupo y de Si cual es la conducta del grupo en interacción con el sujeto Si. Se denotara con GCGS (Si,Q) la Cantidad de Ponderaciones con valor Q que tiene el grupo respecto del sujeto Si. La valoración de GCGS del sujeto Si estará dada por el vector:

(importantei, normali, marginadoi)

Donde:

importantei = GCGS (Si,5) + GCGS (Si,4)
normali = GCGS (Si,3) + GCGS (Si,2)
marginado i = GCGS (Si,1) + GCGS (Si,0)

La reglas de decisión que se proponen en este trabajo son:

Si es importante del grupo si
importantei >= normali + marginado i
Si es normal del grupo si
normali >= importantei + marginado i
Si es marginado del grupo si
marginado i >= normali + importantei

En caso que la ponderación se encuentre con el mismo valor en dos de los tres componentes (Importante, Normal, Marginado), serán compartidos los atributos de la ponderación. El rol “importante” puede corresponder a un líder potencial.

4.1.12 Canales de comunicación

Que se denotaran con la letra K, son las distintas vías por las cuales todos los sujetos del grupo se pueden comunicar. Siendo que la cantidad máxima de posibles canales de comunicación se encuentra determina por la siguiente expresión (donde L representa la cantidad de sujetos del grupo):

$$K = \frac{L}{2} * (L - 1)$$

4.1.13 Porcentaje de carencia de comunicación

La carencia de comunicación está determinada por los valores totales de Ceros (no existe comunicación según el criterio de ponderación de la Matriz MRDCI) divido la cantidad posible de comunicaciones en la Matriz de Interacciones.

$$\%\ Carencia\ de\ Comunicación = \frac{Cantidad\ de\ Ceros}{(L-1)*L} * 100$$

4.2. Procedimiento Propuesto

Paso 1:	Relevamiento de los valores de tiempo de respuesta y mensajes enviados/recibidos en un lapso de tiempo.
Paso 2:	Construcción de las Matrices de Interacciones . MA(i,j), MB(i,j)
Paso 3:	Construcción de las Matrices de Densidad de Comunicaciones Sujeto-Grupo (MDCSG) y la Matriz Integracion.
Paso 5:	Calculo del IACn+1, j y del IHCi,n+1
Paso 6:	Calculo del Índice de realidad colaborativa (IRCi) para el sujeto Si.
Paso 7:	Interpretación del Índice de realidad colaborativa (IRCi) resultante para cada sujeto Si.
Paso 8:	Cálculo del escalón de segmentación (ES).
Paso 9:	Asignación de los valores comprendidos entre los correspondientes rangos de los escalones de segmentación.
Paso 10:	Orlado de la Matriz MI de Interacciones Sujeto-Grupo (SG) con la fila Grado de Colaboración Sujeto-Grupo (GCSG)
Paso 11:	Interpretación de la Matriz MI de Interacciones Sujeto-Grupo (SG)
Paso 12:	Orlado de la Matriz MI de Interacciones Grupo-Sujeto (GS) con la columna Grado de Colaboración Grupo-Sujeto (GCGS).
Paso 13:	Cálculo de: canales de comunicación y porcentaje de carencia de comunicación.
Paso 14:	Cálculo de la desviación del valor de MI(i,j) respecto del valor medio del indice de integración.
Paso 15:	Graficación de la desviación del parámetro de integración y el valor medio, respecto de los enlaces de comunicación.

5. Caso de Estudio

En esta sección se presenta el caso de estudio (sección 5.1), la aplicación del procedimiento propuesto (sección 5.2), y la interpretación de los resultados obtenidos (sección 5.3).

5.1. Presentación el Caso de Estudio

Las métricas propuestas fueron evaluadas usando los correos electrónicos de los estudiantes de un curso que trabajaron durante un período determinado en la resolución de un problema de un asignatura de grado con intercomunicación mediada a través de un espacio virtual de trabajo colaborativo. El curso está integrado por 4 alumnos que el docente dividió aleatoriamente del curso.

5.2. Aplicación del Procedimiento Propuesto

Paso 1-2: Se completa la matriz MA de Interacciones, en este caso se han analizado 4 sujetos con referencia al tiempo de respuesta de los participantes de cada interacción sus respectivos mensajes enviados/contestados. Ponderados según la Tabla I.

Tabla I. Ponderación de tiempos entre mensajes

Caracterización de los tiempos entre mensajes enviados/contestados	Ponderación
Difícilmente responde al primer mail, excepto que le envié otro	**0**
Demora más de 48 hs en responder el mail.	**1**
La respuesta se encuentra entre las 24 y 48 hs de enviado el mail.	**2**
Responde dentro de las 24 hs de enviado el mail	**3**
Responde dentro de las 24 hs de enviado el mail y con posibles soluciones/sugerencias para resolver la inquietud	**4**
Además de responder el mail antes de las 24 hs con una solución/sugerencia posible, intenta otra vía de comunicación	**5**

La segunda propuesta de adaptación de técnicas sociométricas se encuentra basada en la cantidad de mensajes enviados/recibidos dentro de un rango de tiempo; permitiendo obtener un factor que hemos denominado densidad de comunicación.

enviados/contestados por el sujeto Si al sujeto Sj.

	S1	S2	S3	S4
S1	0	4	3	3
S2	4	0	4	4
S3	4	4	0	2
S4	1	1	1	0

Matriz MA correspondiente a los valores de la tabla I

	S1	S2	S3	S4
S1		41	35	33
S2	40		47	44
S3	44	43		21
S4	12	15	18	

Matriz MB. Cantidad de mensajes enviados/recibidos en unintervalo de tiempo.

Intervalo de la muestra: 35 días
Paso 3-4

	S1	S2	S3	S4
S1		1,171	1	0,943
S2	1,143		1,343	1,257
S3	1,257	1,229		0,6
S4	0,343	0,429	0,514	

Matriz MDCSG

	S1	S2	S3	S4
S1		4,69	3,00	2,83
S2	4,57		5,37	5,03
S3	5,03	4,91		1,20
S4	0,34	0,43	0,51	

Matriz Integración

Paso 5-6-7:

	S1	S2	S3	S4	IAC
S1		4,686	3,000	2,829	3,505
S2	4,571		5,371	5,029	4,990
S3	5,029	4,914		1,200	3,714
S4	0,343	0,429	0,514		0,429
IHC	3,314	3,343	2,962	3,019	

	IAC	IHC	IRC
S1	3,505	3,314	1,9048
S2	4,990	3,343	16,476
S3	3,714	2,962	7,5238
S4	0,429	3,019	25,905
		MIRCA	**12,952**

Paso 7-8-9 :

Escalón Seg
5,371
4,366
3,360
2,354
1,349
0,343

	A	B	C	D	E	0
5,371						
4,366						
3,360						
2,354						
1,349						
0,343						

Paso 10-11-12:

RANGOS

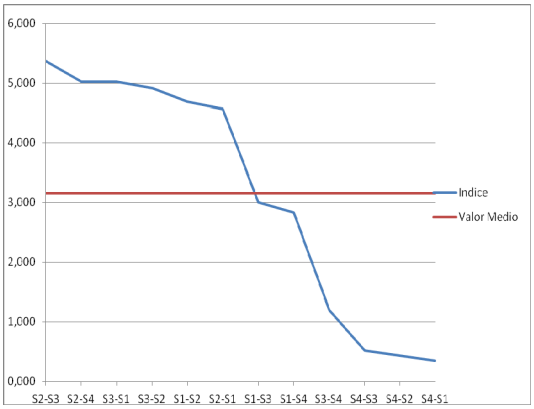
		S1	S2	S3	S4	A	B	C	D	E	0	Acep	Nor	Rec
	S1		4,686	3,000	2,829	1		2					*	
	S2	4,571		5,371	5,029	3						*		
	S3	5,029	4,914		1,200	2				1		*		
	S4	0,343	0,429	0,514						3				*
RANGOS	A	2	2	1	1									
	B													
	C			1	1									
	D													
	E	1	1	1	1									
	0													
	I	*	*			Importante								
	N			*	*	Normal								
	M					Marginado								

Aceptador
Normal
Rechazador

Paso 13: Canales de Comunicación: 6 Porcentaje de Carencia de Comunicación:0

Paso 14-15:

Sujetos	Indice	Valor Medio
S2-S3	5,371	3,160
S2-S4	5,029	3,160
S3-S1	5,029	3,160
S3-S2	4,914	3,160
S1-S2	4,686	3,160
S2-S1	4,571	3,160
S1-S3	3,000	3,160
S1-S4	2,829	3,160
S3-S4	1,200	3,160
S4-S3	0,514	3,160
S4-S2	0,429	3,160
S4-S1	0,343	3,160



6. Conclusiones

De los resultados prelimares obtenidos en el EVTC de experimentación, se observan elementos vinculantes con la educación a distancia, como así mismo aplicables a

redes sociales, en donde los participantes cuentan con comunicaciones sincrónicas y asincrónicas. Los valores que conforman cada par (ejemplo: S2-S3) nos indica el grado de vinculación entre los participantes, cuanto menor es el valor, menor intercambio de mensajes y permite detectar la intervención del docente/tutor.
A futuro, se prevé: [a] hacer una validación sistemática de los indicadores en una muestra amplia de grupos de trabajo colaborativo en espacios virtuales; y [b] explorar la adaptación de otros indicadores sociométricos; y [c] proponer un indicador de nivel de aporte cualitativo de los mensajes.
El componente humano que interviene en la interacción mediada por EVTC propio de la dinámica grupal, genera efectos en los aprendizajes individuales y grupales, planteando la necesidad de disponer de instrumentos que asistan el análisis diagnostico para la necesaria intervención del docente en los procesos pedagógicos en espacios virtuales.

7. Referencias

Bastín, G. (1966). *Los test sociométricos*. Kapeluz.

Charczuk, N., Rodríguez, D., García-Martínez, R. 2013. *Propuesta de Técnicas de Diagnóstico Sociométrico de Dinámicas Grupales para Utilizar en Ambientes de Trabajo Colaborativo*. Proceedings VIII TEYET. ISBN 978-987-1676-04-0. UNSE

Charczuk, N., Rodríguez, D., García-Martínez, R. 2014. *Métricas de Participación en Ambientes de Trabajo Colaborativo Orientadas a la Generación de Intervenciones Didácticas*. CACIC. ISBN 978-987-3806-05-6. Universidad Nacional de La Matanza.

Cirigliano, G., & Villaverde, A. 1966. *Dinámica de Grupos y Educación*. HVMANITAS.

Cohen, S., Bailey, D. 1997. *What Makes Team Work: Group Effectiveness Research from the Shop Floor to the Executive Suite*. Journal of Management, 23(3): 239-290.

Cols, S. A. (1977). *Conducción del aprendizaje*. Marymar.

De Ketele, J.-M., & Roegiers, X. 1995. *Metodología para la Recogida de Información*. Editorial La Muralla.

Deleuze, Gilles.2014. *El Poder, curso sobre Foucault II*. Editorial Cactus. Bs.As.

Etzkowitz, H. 2003. *Research Groups as 'Quasi-firms': The Invention of the Entrepreneurial University*. Research Policy, 32: 109–121.

Leibovich, É. 1980. *Medida de la relación social en el aula*. CINAE.

Li Charlene & Bernoff Josh 2008. *Groundswell: Winning in a world transformed by social technologies*. Harvard Business Press

Litwin Edith(compiladora) 2000. *La educación a distancia*. Colección Agenda educativa.

Morin, E. (1980). *Ciencia con Conciencia*. Anthopos.

Pasel, S., & Asborno, S. 1993. *Aula-Taller*. Aique Grupo Editor S.A.

Perez-Alvares, S. 1991. *El Diagnóstico de la Situación Educativa*. Bs As: Braga 3° Edición.

Pérez-Alvarez, S. 1991. L*as investigaciones exploratorias y descriptivas en las Ciencias de la Educación*. Braga.

Querrien, Anne 1979. *Trabajos elementales sobre la escuela primaria*. La piqueta. Madrid

Rivas, Pablo Ramirez (compilador) 2011. *Fraternidad y Conflicto:enfoques,debates y perspectivas.* Editorial:Ciudad Nueva.Buenos Aires

Rodriguez, D., Ramon Garcia-Martinez, R. 2014. *A Proposal of Interaction Modelling Formalisms in Virtual Collaborative Work Spaces*. LNSE, 2(1): 76-80. ISSN-2301-3559.

Rodríguez, D; Baldizzoni, E; Martins, S; Charczuk, N; García-Martínez, R. *Líneas de I+D+I del Laboratorio de Investigación y Desarrollo en Espacios Virtuales de Trabajo (LIDEVT GISI UNLa).*Proceedings XVI Workshop de Investigadores en Ciencias de la Computación. Pág. 912-915. 2014.WICC 2014, 7 y 8 de Mayo, Ushuaia, Tierra del Fuego,UNTDF

Rosenberg, Marc(2001). *E-Learning: Strategies for Delivering Knowledge in the Digital Age.* McGraw Hill Companies,USA.

Williams,Marcia & Paprock Kenneth & Covington Barbara. (1999). *Distance Learning The Essential Guide*. Sage Publications.

Propuesta de Virtualización Móvil en la Facultad de Ingeniería de la UNLPam

Filippi, José Luis[1], Ballesteros Carlos[1], Bertone Rodolfo[1,2]

[1] Facultad de Ingeniería, Universidad Nacional de la Pampa,
Calle 9 y 110, (6360) General Pico, La Pampa. Argentina
{filippij}@ing.unlpam.edu.ar
{charlyballes}@gmail.com
[2]{pbertone}@ada.info.unlp.edu.ar

Abstract. Una de las características principales de la sociedad actual está determinada por el uso de los dispositivos móviles de última generación.
Con una disponibilidad tecnológica omnipresente en las reparticiones públicas en general, es posible brindar información a los actores que las conforman, en nuestro caso particular hacemos referencia a la Facultad de Ingeniería de la UNLPam.
En éste artículo se propone una suite de aplicaciones móviles que combina el uso de dos sistemas. El primero ligado al sitio de almacenamiento para alojar la información que se va a difundir entre los diferentes actores. El segundo por una suite de aplicaciones móviles, que permite acceder a la información correspondiente, con las actividades que se desarrollan en la institución. Esta suite está constituida por aplicaciones móviles que se implementan en la Facultad de Ingeniería de la UNLPam por el GIAU. [1]

Keywords: Dispositivos móviles, Aprendizaje móvil, Formación a Distancia.

1 Introducción

La aparición de los dispositivos móviles de última generación posibilita nuevas formas de acceder a la información que las diferentes instituciones públicas y/o privadas generan y distribuyen utilizando el potencial de la Internet.

Las nuevas herramientas tecnológicas disponibles en la web viabilizan que la gestión de la información se efectúe con inmediatez haciendo posible a los destinatarios un acceso desde cualquier lugar y en cualquier momento.

Sin embargo las instituciones en general y en nuestro caso particular la Facultad de Ingeniería de la UNLPam, recibe y genera gran cantidad de información, la que se distribuye con inmediatez, utilizando los medios tradicionales de difusión.

Sin duda el proceso de digitalización de la información crece a gran velocidad, al mismo tiempo se observa la presencia de un importante número de dispositivos móviles (celulares y tabletas digitales) entre sus actores (docentes, estudiantes, no docentes, directivos, entre otros). Información que se logra a partir del método de observación directa, que detecta un importante número de estudiantes, docentes, y

personal no docente con dispositivos móviles de última generación con acceso permanente a redes wifi.

A partir de una realidad mediada por contenidos digitales multimedia y dispositivos tecnológicos de última generación, queda en evidencia la importancia de virtualizar la Facultad de Ingeniería con información resultante del procesamiento permanente de digitalización.

En este contexto se presenta el siguiente desafío, pensar aplicaciones móviles como entornos inspiradores, que más allá de su arquitectura tecnológica, permitan construir sobre ellos verdaderos espacios de sentido, con finalidades específicas claras; que además de ser arquitecturas de participación, estimulen a ir más allá de sus paredes virtuales, y permitan dar un valor agregado a la actividad cotidiana. [2]

2 Metodología

Una problemática que se presenta en los centros universitarios en general, está relacionada con el procesamiento, almacenamiento y difusión de la información resultante del proceso de digitalización.

En nuestro caso particular es observable la ausencia de un sistema de virtualización que concentre el material de forma organizada y permita el acceso a través de dispositivos móviles.

Remzi Ates Gürsimsek profesor de la universidad de Stanford e investigador en diseño colaborativo y la innovación en los mundos virtuales, sostiene que los dispositivos móviles y las aplicaciones específicas de cada lugar (incluyendo aplicaciones de medios sociales) están en camino de ser el paradigma dominante. [3]

Considerando esta premisa proponemos un sistema de virtualización acorde a las tecnologías móviles presentes en nuestra Facultad.

Surgen en primer instancia diferentes preguntas, donde el análisis de las posibles respuestas nos llevan a centrarnos en una visión institucional, a través del consejo superior y/o consejo directivo con el apoyo técnico del personal de sistemas.

A modo de ejemplo citamos dos preguntas que sirven de disparador a otras, y que generan una base para un análisis pormenorizado.

- ¿Dónde almacenar la información que genera la institución de forma cotidiana? Y las respuestas posibles: Servidor de terceros o propios.
- ¿La información que se digitaliza es de carácter público, privado o mixto?
- ¿Qué tipos de dispositivos tecnológicos se manipulan para acceder a la información?

Si bien el objetivo de este trabajo no consiste en el análisis de las respuestas puntuales, es importante tener en cuenta que la necesidad de implementar dos sistemas que trabajen en forma vinculada, a fin de cubrir todas las alternativas posibles a partir de esas respuestas.

El primer sistema está ligado al sitio de almacenamiento que se requiere para alojar la información que se espera difundir entre los diferentes actores que constituyen la institución. Para ello se propone como requisito poseer un servicio de datos en la web que cumpla con varias condiciones:

a) Poseer una interface accesible que permita a los editores efectuar las publicaciones sin dificultad.
b) Que no demande gasto monetario en alojar la información en los servidores.
c) Sin límite en cuanto a la cantidad de información que se puede procesar.
d) Accesible desde cualquier lugar y en cualquier momento.

El segundo sistema está conformado por una suite de aplicaciones móviles que permite a la Facultad de Ingeniería de la UNLPam, mantener una red de información permanente con sus estudiantes, docentes y no docentes. Este sistema es el particularmente analizado en este trabajo.

3 Propuesta de una suite de aplicaciones móviles en la Facultad de Ingeniería de la UNLPam

Comprender el verdadero potencial de la tecnología móvil y aprender a evolucionar con él, es el desafío para la sociedad en general y en nuestro caso particular la comunidad educativa universitaria.

Diferentes autores referencian a la aparición de nuevos escenarios culturales que se caracterizan por la posibilidad de convergencia de medios y de la participación productiva de los actores, lo cual nos presenta el desafío de pensar como docentes, nuevas prácticas educativas altamente inmersivas y ubicuas. [4]

Se trata de generar situaciones que hagan participes a los estudiantes de la complejidad a abordar, pudiendo comprender y producir nuevas narrativas, Maggio acentúa la enseñanza enriquecida con escenarios de alta disposición tecnológica. [5]

El trabajo que se propone describe los diferentes componentes que conforman una aplicación móvil que gestiona la información que se procesa en la Facultad de Ingeniería de la UNLPam, y que permite a la institución crear una comunicación dinámica con los diferentes actores que la conforman.

Los dispositivos móviles ofrecen a la comunidad educativa un gran número de posibilidades, entre las que se encuentran:

- Ampliar la oferta educativa a través de entornos virtuales móviles de enseñanza/aprendizaje.
- Automatizar las actividades académicas, mejorando la calidad del servicio prestado.
- Incrementar el trabajo colaborativo entre docentes, alumnos, directivos y administrativos.
- Agilizar la comunicación.

A partir de evidenciar un número importante de dispositivos móviles con conectividad wifi permanente y la capacidad digital de sus usuarios para utilizar las aplicaciones sin dificultad, surge la suite de aplicaciones móviles que se detalla a continuación. La figura 1 presenta una posible solución que se encuentra en su fase de análisis y conformará en un futuro la plataforma móvil a implementar.

Fig.1

La suite de aplicaciones móviles está conformada por:

- Contactos

La opción contactos, presenta un listado con los datos de los todos los actores que conforman la comunidad educativa (estudiantes, directivos, docentes y no docentes). Permite efectuar una llamada telefónica a cualquier sujeto dentro de la Facultad de Ingeniería o enviarle un correo electrónico a su casilla.

- Deporte

Es conocido por todos los beneficios que produce la actividad física en la salud de las personas. En éste espacio encontraras información relacionada con las actividades deportivas que se realizan en la Facultad de Ingeniería como en otras facultades que pertenecen a la UNLPam. Entre las actividades más comunes encontramos convocatorias a competencias, horario y lugar de encuentros deportivos, resultados finales, fotos y videos de los principales momentos.

- Comedor

El comedor universitario es un servicio que presta la UNLPam a todo el personal universitario, siendo sus principales beneficiarios los estudiantes. Esta aplicación permite acceder a las novedades que se publican en éste sector de la Universidad. Se encuentra información relacionada con la venta de tickets, menú semanal, horario del comedor, becas en tickets, entre otras posibilidades.

- Agenda

La aplicación agenda lista los diferentes eventos que se efectúan en nuestra facultad tal como reuniones científicas, congresos, presentación de trabajos final de carrera, conferencias, actos académicos, entre otros. La consulta se genera seleccionando la fecha desde el calendario en curso.

- Cursos

Permite el acceso a la plataforma de M-learning, una metodología de enseñanza y aprendizaje que se efectúa a través de múltiples contextos, de la interacción social y de contenidos, mediados con el uso de dispositivos electrónicos (teléfonos de última generación, tabletas digitales). [7]

- Ubicación

La cantidad de inscriptos a nuestra facultad anualmente está conformada por un alto porcentaje de estudiantes que pertenecen a otras localidades. La aplicación Ubicación muestra un mapa con las referencias a las principales delegaciones y establecimientos de nuestra ciudad. El estudiante que desconoce nuestra localidad podrá encontrar la ubicación de la Facultad de Ingeniería, Veterinaria, Ciencias Humanas, Comedor Universitario, Rectorado, Biblioteca, correspondiente a la UNLPam, y otras instituciones como Municipalidad, Cooperativa Eléctrica, Hospital, Clubs, entre otras.

- Videos

Esta aplicación muestra una lista de los eventos que se registran en formato de video y que son publicados para ser accedidos por toda la comunidad educativa. Se encuentran publicaciones sobre Jornadas Tecnológicas, Promoción de Carreras, Video Institucional, entre otros.

- Imágenes

Esta aplicación muestra un listado de imágenes que se publican sobre los eventos en que participa la Facultad de Ingeniería a través de sus actores, estudiantes, docentes, directivos y no docentes. De igual forma se encuentran imágenes de las áreas que conforman el establecimiento, comedor, residencia estudiantil, laboratorios, aulas, sala de videoconferencia.

- Noticias

La Facultad de Ingeniería posee el área de difusión que se encarga de editar y transmitir las noticias relacionadas con los sucesos más destacados. Esta aplicación permite recorrer las novedades por fecha de publicación. Al mismo tiempo funciona como repositorio digital al permitir acceder a cualquier noticia publicada, sin importar su antigüedad.

- Biblioteca

La aplicación biblioteca permite efectuar consultas sobre la disponibilidad bibliográfica, fechas de vencimiento, reservas, entre otras opciones. Al mismo tiempo permite acceder a documentos digitales de diferente índole.

- Laboratorios

La aplicación laboratorios permite a los estudiantes conocer el horario que cada asignatura utiliza para efectuar sus prácticas educativas. Al mismo tiempo se encuentra información digitalizada sobre las normas de seguridad e higiene que se deben cumplir para hacer uso de los mismos en forma segura y responsable.

- Gestión de aula

Es una aplicación que permite el acceso al sistema de gestión de aula, muestra la distribución de los horarios en que se desarrollan las actividades educativas y el aula que corresponde a cada una.

- Ómnibus

La aplicación ómnibus muestra el recorrido que las diferentes líneas de colectivos realizan por la ciudad, destacándose las principales paradas de interés general.

- Barra de estado

La aplicación cuenta con una barra de estado donde se definen iconos de acceso rápido a la ubicación de la facultad de Ingeniería, permitiendo al estudiante que ingresa por primera vez a nuestra ciudad llegar a destino sin extraviarse. Presenta además acceso directo al sitio web oficial donde encuentra información complementaria a la aquí expuesta. Otros accesos directos como email y número de teléfono permiten una comunicación rápida con la institución.

La suite de aplicaciones móviles que se describe no es una plataforma cerrada, por el contrario, su diseño modular permite añadir nuevas aplicaciones a través de periódicas actualizaciones. Fig.2

Fig.2

3 Conclusiones

Actualmente hay más de seis mil millones de suscripciones de telefonía móvil en todo el mundo, y por cada persona que accede a Internet desde un ordenador dos lo hacen desde un dispositivo móvil. Dada la ubicuidad y la funcionalidad de las tecnologías móviles en rápida expansión, la UNESCO está entusiasmada con su potencial para mejorar y facilitar el aprendizaje, sobre todo en comunidades donde las oportunidades educativas son escasas. [6]

La presencia de los dispositivos tecnológicos se caracterizan por su reducido tamaño, bajo costo, y rápida conexión con redes wifi, hacen posible pensar que el proceso educativo mediado por ésta tecnología es posible en y desde cualquier parte del mundo. La ubicuidad deja de ser una característica propia de la tecnología y pasa también a transformar el proceso de enseñanza y aprendizaje. [7]

El aprendizaje móvil (mobile learning) nos permite llevar de un lado a otro un curso completo y nos invita a compartir en forma casi instantánea todo lo que sucede

alrededor con nuestras redes. Se derriban los muros de las aulas y se expande la acción hacia nuevos horizontes. [7]

El trabajo propuesto está concebido para ser una ayuda al proceso educativo formal presencial existente actualmente en nuestra facultad. A sabiendas que el desafío implica un esfuerzo importante y que el éxito no está garantizado, es imprescindible considerar aspectos que pueden influir positiva o negativamente en todo el proceso:

- Necesidad de innovar a través de diferentes propuestas de formación mediada por dispositivos tecnológicos de última generación.
- Aceptación de nuevos desafíos de los actores involucrados.
- Disposición de la arquitectura tecnológica adecuada.
- Fluidez en la comunicación durante todo el proceso.

El objetivo que se pretende alcanzar al incorporar la suite de aplicaciones móviles en la Facultad de Ingeniería de la UNLPam, tiene que ver con hacer de la institución un ambiente educativo ágil, que cumpla con las características básicas del aprendizaje móvil: [8]

- Ubicuo: posibilidad de acceso desde cualquier lugar y en cualquier momento.
- Flexible: se adapte a las necesidades de cada uno.
- Portable: su tamaño permite la movilidad con el usuario.
- Inmediato: posibilidad de acceder a la información en cualquier momento.
- Motivante: su uso potencia la motivación en el usuario.
- Accesible: de bajo costo.
- Activo: potencia en los estudiantes un papel más activo.
- Conectividad a internet: permite el acceso a la información que se almacena en internet.
- Acceso a variadas aplicaciones: permite el acceso a varias aplicaciones.
- Sensores multifunción: los dispositivos móviles de última generación disponen de sensores tipo acelerómetros, GPS, cámaras, etc., que enriquecen el proceso formativo.
- Personales: son propios de cada usuario, existe una relación personal hacia el mismo.
- Pantalla táctil: permite otra serie de utilidades que enriquecen la navegación por las aplicaciones.

En la medida que se logre fusionar los aspectos presentados en forma objetiva, se podrá alcanzar un estado de superación permanente, a través de incorporar las TICs en las instituciones educativas en el nivel superior.

Más allá de los números artículos que destacan la valiosa posibilidad que ofrece la virtualización en general en las entidades educativas, y que la Facultad de Ingeniería cuenta con la tecnología necesaria, debemos resaltar la importancia de nuestra propuesta, que permitirá el acceso a la inmensa cantidad de información institucional que se encuentra en formato digital.

Se concluye que la suite que proponemos va a permitir ahorrar espacio de almacenamiento, disminuir el costo que implica el acopio en papel, además de un acceso rápido y eficiente a través de la búsqueda digital, dejando abierta nuevas posibilidades de comunicación impensadas e imprevisibles.

4 Trabajo Futuro

A partir de una realidad social con dispositivos tecnológicos omnipresentes, se podría decir que es inevitable que los centros de formación en sus diferentes niveles formulen propuestas educativas que incorporen las nuevas tecnologías de uso corriente. En nuestro caso particular proponemos una suite de aplicaciones orientadas a dispositivos móviles de última generación.

La suite que se propone en éste artículo va a permitir que los diferentes actores que conforman la comunidad educativa accedan a información relevante del acontecer universitario, en sus diferentes formatos multimedia.

Howard Rheingold revela que se impone una nueva forma de organización social que señala como multitudes inteligentes, una forma de organización social que nace y se estructura a través de las nuevas tecnologías de la información y las comunicaciones. [9]

Es una aplicación móvil que va a exhibir a la Facultad de Ingeniería de la UNLPam, a través de dispositivos móviles de última generación. [10]

El trabajo desarrollado en la primer fase comprende el análisis de diferentes aplicación móviles educativas de nivel terciario y universitario. La fase culmina con el presente artículo, en el cual se expone la propuesta a desarrollar en la Facultad de Ingeniería de la UNLPam, por el grupo de investigación y desarrollo GIAU. [1]

El trabajo siguiente comprende el análisis de cada módulo que se pretende desarrollar, proyectando los recursos necesarios, tareas, tiempo y personal especializado en programación de dispositivos móviles.

El desafío más allá del trabajo que se describe, radica en lograr en las diferentes aplicaciones, interfaces que se adapten a los distintos dispositivos móviles (teléfonos de última generación, tabletas digitales), que mantengan la funcionalidad, y que permitan al usuario un manejo intuitivo donde rescate las experiencias de acuerdo al uso que hace de las mismas.

Referencias

1. GIAU. Grupo de Investigación de Ambientes Ubicuos. http://giau.ing.unlpam.edu.ar/
2. Barberá, E. Badía, A. Educar con aulas virtuales. Cap 3. Del aula virtual al aula presencial. Machado Libros. Edición Boardilla del Monte. Madrid. 2004
3. Remzi Ates Gürsimsek. 2011. Future of virtuality in-the-making: Reflections from Stanford University. http://blogs.ufm.dk/indlaeg/silicon-valley/future-of-virtuality-in-the-making/
4. Nicholas Burbules. WEBINAR 2012. http://www.webinar.org.ar/conferencias/entrevista-nicholas-burbules
5. Maggio, M. Enriquecer la enseñanza. Buenos Aires: Paidós. 2012.
6. UNESCO Mobile Learning Publications. ICT Educations. http://www.unesco.org/new/en/unesco/themes/icts/m4ed/mobile%E2%80%90learning%E2%80%90resources/unescomobilelearningseries/

7. Nicholas Burbules. Educ.ar 2011. http://www.educ.ar/sitios/educar/recursos/ver?id=116538
8. Guía para la implantación del MOBILE LEARNING. 2013. http://serviciosgate.upm.es/docs/asesoramiento/guia_implementacion_movil.pdf
9. Howard Rheingold, Smart Mobs: La Revolución Social Siguiente. GEDISA (2009).
10. Bucky, J, 'Definition of Mobile Device', about.com, viewed December 2013 http://operationstech.about.com/od/glossary/g/Definition-Of-Mobile-Device.htm

Analizando el uso de (Dyn)Alloy como Herramienta Educativa

César Cornejo[1], Mariano Politano[1], Fernando Raverta[1], Sonia Permigiani[1], Pablo Ponzio[1,2], Germán Regis[1], and Nazareno Aguirre[1,2]

[1] Dpto. de Computación, FCEFQyN, Universidad Nac. de Río Cuarto, Argentina
[2] Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina

Resumen Los significativos avances en técnicas automáticas de análisis, como model checking, constraint solving y computación evolutiva, adquieren constantemente mayor relevancia en actividades complejas de construcción de software, y son exitosamente utilizadas para generar tests automáticamente, refinar requisitos de software, verificar diseños, y descubrir errores de programas. Sin embargo, el uso de tales técnicas para asistir en actividades de enseñanza-aprendizaje es muy escaso. En este trabajo, analizamos el uso del lenguaje formal Alloy y su extensión DynAlloy, que incorpora acciones y programas, como herramienta educativa, para la asistencia y el soporte de tareas de elaboración y comprensión de especificaciones, y otras actividades en las cuales las especificaciones, entendidas como descripciones declarativas de software, son centrales. Mostraremos cómo el análisis automático subyacente a estos lenguajes puede emplearse efectivamente para dar soporte en la depuración de especificaciones, permite introducir naturalmente conceptos que suelen escapar a cursos introductorios, como el no determinismo, y puede facilitar el uso de abstracciones de datos adecuadas en cursos introductorios.

1. Introducción

En las últimas décadas, los enormes avances en técnicas automáticas de análisis han comenzado a dar lugar a novedosas aplicaciones en diversas áreas de la Ingeniería de Software. En efecto, algunas tecnologías de análisis automático como el *model checking* [5], constraint solving (SAT, SMT) [4] y la computación evolutiva [10], se aplican con éxito a tareas como la operacionalización de requisitos, la verificación de diseños de software, la generación automática de tests, el descubrimiento de defectos en software, entre otras. Sin embargo, la adopción de estas poderosas tecnologías como herramientas de apoyo a actividades educativas en Ciencias de la Computación es escasa. A pesar del limitado uso de estas técnicas para tareas vinculadas a la educación en Informática, es significativo el potencial que las mismas tienen, en particular como formas de automatizar, o semi automatizar, actividades de soporte a los estudiantes. Un ejemplo singular es el PEX [13], una herramienta de generación automática de tests basada en ejecución simbólica dinámica. PEX ha sido utilizada con éxito para realizar debugging automático y así guiar estudiantes en la construcción de programas,

mediante la contrastación de las soluciones de los estudiantes con implementaciones correctas (secretas) para detectar discrepancias de comportamiento y proveer las mismas como feedback a los usuarios [14]. Otro ejemplo similar, basado en tests como feedback para estudiantes en la resolución de ejercicios, lo constituye la herramienta `Codeboard`[1], que permite cargar ejercicios de programación (en varios lenguajes), proveer tests ocultos para los mismos, y ejecutar los tests sobre las resoluciones de los estudiantes, para proveer feedback previo a la entrega de los trabajos.

Mientras PEX y CodeBoard se concentran en proveer feedback automáticamente, pero vinculado a la construcción de programas, en este artículo analizamos un problema igualmente relevante, el de proveer feedback automático vinculado a la construcción de *especificaciones*. Es ampliamente aceptado en el ámbito de la Ingeniería de Software y las Ciencias de la Computación que una correcta, precisa y completa comprensión del problema a resolver, y una adecuada captura del mismo, son fundamentales para el desarrollo efectivo y eficiente de software. Más aún, en contextos de programación *in the small*, suele contarse con lenguajes formales para capturar "el problema", idealmente previo al desarrollo del software en sí mismo. De hecho, no son pocas las asignaturas de programación vistas en carreras de Computación del país y del exterior que utilizan tales lenguajes como parte de la instrucción fundamental de la disciplina. Sin embargo, la asistencia y soporte a los estudiantes en relación a la ayuda en construcción de especificaciones es casi enteramente manual, dependiendo de contar, en persona, con un instructor o asistente que funcione como "oráculo", es decir, que evalúe las soluciones de estudiantes y guíe a los mismos en la resolución de problemas con sus especificaciones. Esto deriva en frecuentes problemas relacionados a las dificultades tanto a la hora de escribir especificaciones como de comprenderlas, observados por docentes y estudiantes. Las técnicas avanzadas de constraint solving, y en particular las basadas en satisfactibilidad booleana (SAT solving) y satisfactibilidad módulo teorías (SMT solving) tienen el potencial de contribuir a paliar este problema, como ilustraremos y discutiremos en este artículo. Más precisamente, nos concentraremos en el uso del lenguaje relacional Alloy, y su extensión DynAlloy, que incorpora acciones y programas, y mostraremos cómo estos lenguajes y las herramientas de análisis (basadas en SAT solving) asociadas a los mismos pueden ser usadas para asistir en tareas como las descriptas previamente.

2. DynAlloy y su Herramienta de Análisis

En esta sección describimos Alloy y DynAlloy, los dos lenguajes que analizamos y proponemos para el análisis de especificaciones como parte de actividades educativas. Alloy [7] es un lenguaje de especificación, de los denominados *basados en modelos*. La sintaxis de Alloy es simple e intuitiva para los desarrolladores acostumbrados a lenguajes orientados a objetos, con una semántica relacional

[1] `http://codeboard.io`

simple. Sus modelos se componen de dominios de datos, propiedades y operaciones entre estos dominios. Una característica que destaca a Alloy sobre otros lenguajes formales de especificación como Z [9] y B [1] radica en su principal objetivo centrado en brindar análisis automático. Alloy Analyzer es la herramienta soporte del lenguaje que permite simular modelos y buscar contraejemplos de propiedades, en modelos acotados. El análisis de modelos Alloy se basa en la reducción de los mismos a fórmulas proposicionales cuyas satisfactibilidad se verifica mediante la utilización de SAT-solvers [6, 12]. DynAlloy [8] extiende Alloy con estado, acciones que permiten alterar el mismo, programas compuestos de acciones, y aserciones de corrección parcial que acompañan tales programas. Si uno especifica dominios de datos en Alloy, y programas sobre los mismos usando DynAlloy, es posible simular la ejecución de tales programas, y buscar contraejemplos en modelos acotados, para aserciones de corrección parcial (pre y post condiciones, etc.).

Los dominios de datos en (DynAlloy) se construyen a través de *signaturas* (el único tipo predefinido es `Int`). Suponiendo que queremos modelar secuencias de elementos es cierto tipo, podemos definir el dominio de los elementos (sobre el cual definiremos secuencias) utilizando una signatura simple, de la siguiente manera:

```
sig Elem { }
```

Las signaturas pueden también tener estructura, definida en términos de *campos*. Por ejemplo, podemos modelar secuencias de elementos de tipo `Elem` usando la siguiente signatura:

```
sig Seq {
   idx: set Int,
   length: Int,
   at: idx -> one Elem }
```

Una secuencia tiene un conjunto de índices, una longitud, y una función que para cada índice, indica cuál es el elemento almacenado en el mismo. Nótese cómo podemos definir campos de tipos relacionales (es una de las potencias del lenguaje). Ahora, si solicitamos a la herramienta que nos brinde un ejemplo de una secuencia de caracteres, seguramente nos desilusionaremos un poco, porque aún no hemos capturado en el modelo algunos elementos importantes sobre las secuencias. Debemos imponer, por ejemplo, que el tamaño de una secuencia es siempre mayor o igual a 0, el conjunto de indices es el rango de 0 a la longitud de la secuencia menos 1, etc. Estas restricciones son especificadas en Alloy mediante *hechos*, que restringen modelos con fórmulas que se imponen como válidas, como axiomas. Por ejemplo, tenemos para nuestro modelo:

```
fact SpecSeq { (all s: Seq | s.length >= 0 ) &&
   (all s: Seq, i: Int | i >= 0 && i < s.length => some s.at[i]) && ... }
```

Además de hechos, Alloy soporta *predicados* y *aserciones*. Ambas modelan propiedades; los predicados son fórmulas que capturan modelos con características particulares. Por ejemplo, podemos caracterizar *palíndromos* con un predicado como el siguiente:

```
pred esPalindromo(s: Seq) {
   all i: Int | i>=0 and i<s.length => s.at[i] = s.at[s.length - (i+1)] }
```

Las aserciones capturan propiedades a verificar, es decir fórmulas que uno espera que sean verdaderas en todos los modelos que satisfagan los hechos. Por ejemplo, la siguiente aserción:

```
assert idxDefinido {
   all s: Seq, i: Int | i>=0 and i<s.length => some s.at[i] }
```

Esta aserción indica que en toda secuencia, si un índice es menor que la longitud de la secuencia y mayor o igual a cero, entonces la secuencia está "definida" en ese índice. La diferencia principal entre predicados y aserciones está en análisis. Mientras Alloy Analyzer buscará *instancias* de predicados, la herramienta buscará *contraejemplos* de aserciones.

El modelo anterior es *estático*, en el sentido que no captura estado, ni permite cambiar estados. Si bien estos elementos se pueden codificar en Alloy (como predicados que vinculan estados previos y posteriores a la ejecución de operaciones, y modelos ad hoc de trazas de ejecución), DynAlloy provee una forma más conveniente, eficiente y clara de hacerlo, a través de *acciones* y *programas* [8]. La siguiente acción, por ejemplo, agrega un elemento al final de una secuencia:

```
act append[s: Seq, e: Elem] {
   pre { }
   post { s'.length = s.length+1 and s'.at[s'.length-1] = e and
           all i: Int | i>=0 && i<s'.length-1 => s'.at[i] = s.at[i] } }
```

Las acciones atómicas se *definen* por su pre y postcondición (es decir, su definición es un hecho). En cambio las pre y postcondiciones para programas son *aserciones*, es decir se verificarán (en escenarios acotados). DynAlloy provee tres operadores que permiten componer acciones atómicas: composición secuencial (;), elección no determinista (+) e iteración (*). Además de los operadores de composición, Dynalloy ofrece la posibilidad de incorporar *tests* (asunciones) en la especificación de los programas y azúcar sintáctico que convierten el proceso de especificación en una tarea de programación en un lenguaje imperativo. La verificación de estos programas se realiza mediante la definición de pre y postcondiciones de los mismos. A modo de ejemplo de programa más complejo anotado por su especificación, consideremos la búsqueda secuencial:

```
assertCorrectness busquedaSecuencialCorrecta {
   pre { s.length >= 0 }
   program busquedaSecuencial[s: Seq, elem: Elem, out: Boolean] {
      var i: Int;
      out := False;
      i := 0;
      while (i<s.length and not out) {
         if (s.at[i]=elem) { out := True; }
         i := i+1; } }
   post { out <=> (some x: Int | x>=0 and x<s.length and s.at[x]=elem) } }
```

DynAlloy Analyzer permite realizar diversos análisis sobre especificaciones (Dyn)Alloy. Además de buscar instancias de predicados y contraejemplos de

aserciones, permite simular ejecuciones de programas DynAlloy, y verificar aserciones de corrección parcial. La verificación se realiza, al igual que en el caso de Alloy, mediante una reducción a SAT solving, y para escenarios acotados (es decir que, para todo escenario acotado que satisfaga la precondición, al ejecutar el programa deriva necesariamente en un estado que satisface la postcondición). El comando de análisis recibe una especificación de las cotas. Por ejemplo, el comando:

```
check busquedaSecuencialCorrecta for 1 Seq, 5 Elem, 5 lurs
```

verificará que efectivamente el programa cumple con su especificación en todas las ejecuciones de hasta 5 iteraciones (5 lurs), partiendo de estados compuestos por 1 secuencia (1 Seq) y hasta 5 elementos (5 Elem). Esta comprobación podría evidenciar problemas en la especificación y/o el programa. Suponiendo que esta comprobación se satisface, no constituye una *prueba*, pues sólo garantiza la corrección del programa en un número de escenarios acotados. La Figura 1 muestra una captura de pantalla de DynAlloy Analyzer, con el programa correspondiente a la búsqueda secuencial. Los contraejemplos se pueden visualizar de diferentes maneras, como ilustraremos más adelante.

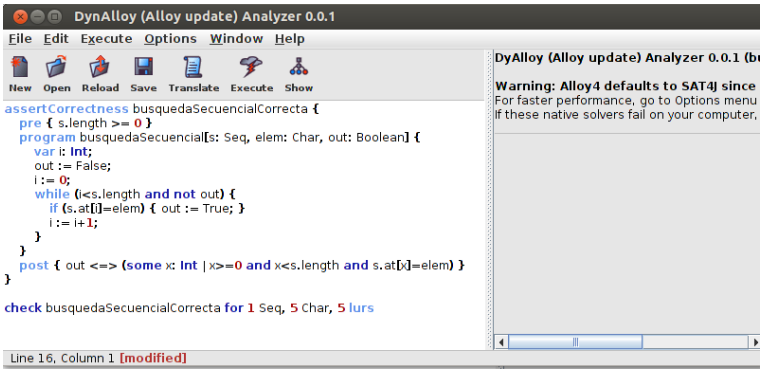


Figura 1. DynAlloy Analyzer: Interfaz gráfica con un ejemplo simple

3. Análisis del uso Educativo de (Dyn)Alloy

En esta sección discutiremos una variedad de tareas en las cuales herramientas como las asociadas a Alloy y DynAlloy pueden contribuir en el desarrollo de especificaciones.

3.1. Asistencia en la Elaboración de Especificaciones

Especificar formalmente lo que se intenta resolver mediante un programa es una tarea compleja. Demanda no sólo comprender el lenguaje de especificación y la semántica de sus construcciones, sino el uso correcto de sus construcciones, y la captura correcta mediante las mismas del problema en cuestión. Es frecuente encontrarse con dificultades vinculadas a la construcción de especificaciones inconsistentes (a veces sutilmente), casos no considerados, y demás problemas. En

estas circunstancias, y por tratarse las especificaciones de objetos formales no ejecutables, los estudiantes tienen pocas herramientas para depurar sus especificaciones más que ponerlas a consideración de docentes. Constraint solving puede usarse para ayudar en estas circunstancias, como herramienta de comprobación automática.

Consideremos el siguiente ejemplo. Supongamos que se desea especificar una operación que, dada una secuencia s, determina si la misma es o no un *palíndromo*. Esta operación puede ser especificada como mostramos en la sección anterior. Este ejemplo, de hecho, fue tomado de un examen. Un error común cometido con frecuencia en la resolución de este ejercicio en el mencionado examen se dio en el manejo de índices. Más precisamente, muchos estudiantes escribieron la siguiente especificación:

```
pred esPalindromoEst(s: Seq) {
   all i: Int | i>=0 and i<s.length => s.at[i] = s.at[s.length - i] }
```

Suponiendo que los docentes proveen la especificación correcta dada precedentemente, Alloy podría utilizarse como asistente automático, mediante el chequeo de la siguiente aserción:

```
assert esPalindromoEstEsCorrecto {
    all s: Seq | esPalindromo[s] iff esPalindromoEst[s] }
```

Alloy Analyzer producirá en este caso contraejemplos, como el que se muestra en la Fig. 2, que el estudiante puede aprovechar para depurar su especificación. En el mismo se muestra que la especificación no es equivalente para la secuencia de un elemento. Efectivamente, la especificación provista por el estudiante intenta evaluar entre la posición 0 y la posición 1 de la secuencia, con esta última inexistente. Con este razonamiento el estudiante podría corregir su especificación.

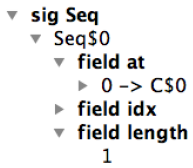


Figura 2. Contra-ejemplo generado por Alloy Analyzer

3.2. Asistencia en la Derivación/Verificación de Programas

Verificar que un programa satisface su especificación es indecidible, y por lo tanto, escapa a lo que puede resolverse automáticamente. Sin embargo, podemos automatizar parte de las tareas en este proceso, y como mínimo brindar ayuda a los estudiantes mediante herramientas, en pasos de estos procesos que corresponden a los denominados "eureka" (i.e., que requieren de la intervención por parte del usuario, para la provisión de definiciones adecuadas). Un ejemplo de esto es la verificación de programas iterativos, que demanda la provisión de *invariantes* adecuados. Consideremos, por ejemplo, el siguiente caso. Supongamos que contamos con el programa anotado para la búsqueda secuencial, provisto en la sección anterior. La verificación del mismo requiere la provisión de un *invariante* adecuado para el ciclo. Podemos sin embargo realizar algunas tareas automáticamente

usando DynAlloy Analyzer, incluso previo a la construcción del invariante. La primera es comprobar, para ejecuciones acotadas en su número de iteraciones y para conjuntos de estados acotados, si efectivamente el programa cumple con su especificación. Por ejemplo, el comando:

```
check busquedaSecuencialCorrecta for 1 Seq, 5 Elem, 5 lurs
```

verifica, como describimos anteriormente, que el programa satisfaga su especificación en los escenarios y ejecuciones acotados según se indica en el mismo. Sin embargo, esto no constituye una prueba, dado que podrían existir ejecuciones más extensas (con más iteraciones) o sobre estados más complejos que violen la propiedad. Una verificación completa del mismo demanda una prueba, que requiere un invariante para el ciclo del programa. Un posible invariante, que debe ser provisto por el usuario, es el siguiente:

```
inv: out <=> (some x: Int | x>=0 and x<i and s.at[x]=elem)
```

Ahora que tenemos una propuesta de invariante, podemos realizar nuevos análisis. Una primera comprobación consiste en chequear que, para un número acotado de escenarios, esta fórmula se comporta como un invariante, es decir, es válida antes de comenzar a iterar, y se preserva luego de cada iteración. Esto se consigue en DynAlloy simplemente agregando:

```
assert inv;
```

como sentencia inmediatamente previa al ciclo, y como última sentencia del cuerpo del mismo. Nuevamente, los contraejemplos que pudieran surgir de esta etapa ayudarán al estudiante a depurar y mejorar su invariante. Finalmente, si esta nueva etapa es superada, debemos comprobar que el invariante es *inductivo*. Si bien esta tarea se realiza siguiendo reglas de inferencia y el cálculo de *weakest precondition*, DynAlloy Analyzer puede emplearse para encontrar contraejemplos automáticamente, y depurar el invariante previo a encarar la prueba. Esto se consigue verificando las siguientes aserciones de corrección parcial:

```
assertCorrectness proofOblig1 {
   pre { s.length >= 0 }
   program init[s: Seq, elem: Elem, out: Boolean] {
      var i: Int;
      out := False;
      i := 0; }
   post { inv }
}
assertCorrectness proofOblig2 {
   pre { inv and i<s.length and not out }
   program loop[s: Seq, elem: Elem, out: Boolean, i: Int] {
      if (s.at[i]=elem) { out := True; }
      i := i+1; }
   post { inv }
}
assert proofOblig3 {
   all s: Seq, elem: Elem, out: Boolean, i: Int |
   inv and not (i<s.length and not out) =>
   (out <=> (some x: Int | x>=0 and x<i and s.at[x]=elem)) }
```

Para nuestro ejemplo, `proofOblig1` se cumple, mientras que `proofOblig2` y `proofOblig3` no se cumplen; el invariante no es inductivo, dado que debe fortalecerse con el rango de la variable `i`.

3.3. No Determinismo y Abstracciones de Datos

En asignaturas sobre metodologías de la programación (programación estructurada, verificación de programas *in the small*, etc.), suele emplearse pseudocódigo de algún tipo, para facilitar la comunicación de diseños algorítmicos, y no estar atado a un lenguaje de programación particular. Se usa desde pseudocódigo informal, hasta lenguajes del estilo del lenguaje de comandos con guardas (GCL) de Dijkstra. Si bien estos lenguajes ofrecen facilidades como no determinismo (en IF o DO, en el caso de GCL, por ejemplo) o tipos de datos más abstractos que los disponibles en lenguajes de programación, el no contar con herramientas de software que soporten los lenguajes limitan su utilización. De hecho, si miramos bibliografía que utilice lenguajes como GCL (e.g., [2,3]), rara vez se utiliza no determinismo. En cuanto al uso de abstracciones de datos, es común observar un uso intensivo de cadenas de caracteres y de tipos numéricos en cursos introductorios. La utilización de lenguajes y herramientas de análisis como DynAlloy pueden ayudar a resolver parcialmente este problema. Consideremos el caso de no determinismo. Para esto pensemos en el acertijo conocido como el *Problema de las Jarras de Agua*: dadas dos jarras de agua de 3 y 4 litros respectivamente, sin marcas de nivel de contenido, podemos conseguir tener exactamente 2 litros en una de ellas si las acciones que podemos hacer son pasar el contenido de una a otra, vaciar o llenar cualquiera de las jarras? En DynAlloy podemos capturar este problema naturalmente con elección no determinista, e incluso ilustrar la definición ad hoc de tipos de datos de manera sumamente conveniente, como mostramos a continuación:

```
sig Estado {
   jarraA: Int,
   jarraB: Int }

fact{all e: Estado | gte[e.jarraA,0] and gte[e.jarraB,0]
   and lte[e.jarraA,4] and lte[e.jarraB,3] }

action vaciarA[e:Estado]{
   pre{gt[e.jarraA,0]}
   post{eq[e'.jarraA,0] and eq[e.jarraB,e'.jarraB]} }
action llenarA[e:Estado]{
   pre{lt[e.jarraA,4]}
   post{eq[e'.jarraA,4] and eq[e.jarraB,e'.jarraB]} }
...
pred jarrasVacias[e: Estado] {
   eq[e.jarraA, 0] and eq[e.jarraB, 0] }
pred final[e: Estado] {
   eq[e.jarraA, 2] or eq[e.jarraB, 2] }

program acertijoJarras[e:Estado]{
   [jarrasVacias[e]]?;
```

```
   repeat{
      vaciarA[e] + vaciarB[e] + llenarA[e] +
      llenarB[e] + transferirDeA[e] + transferirDeB[e] } ;
   [final[e]]?
}
run acertijoJarras for 7 but 5 int lurs exactly
```

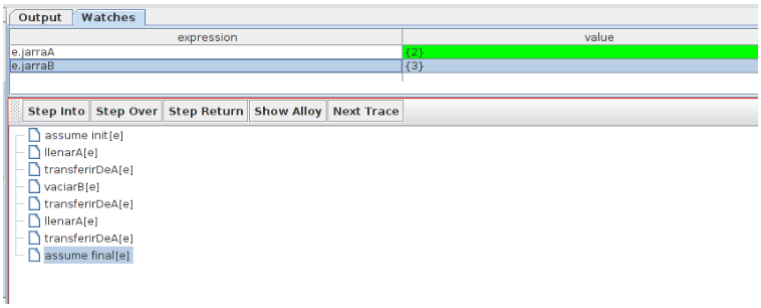


Figura 3. Traza de Ejecucion del problema de las Jarras de Agua

La Figura 3 ilustra la herramienta en ejecución, mostrando una solución al acertijo.

4. Discusión, Conclusiones y Trabajos Futuros

La Ingeniería de Software Educacional [14] propone el uso de tecnología de análisis con fines educativos. Las ventajas de este uso son variadas, especialmente en momentos en los cuales se empieza a observar un uso regional más intensivo de plataformas de educación a distancia. En este tipo de contextos, contar con herramientas de *tutoring* y asistencia automática a los estudiantes (que podría derivar incluso en corrección automática de ejercicios o similares) son sumamente útiles. Mientras existen casos exitosos de tales tecnologías, como PEX [14] y CodeBoard, los mismos se concentran en código. En este trabajo, nos centramos en tareas vinculadas a la construcción de *especificaciones*, centrales en el desarrollo de software, y mostramos cómo el constraint solving, específicamente SAT solving, puede ayudar en tareas como la asistencia a la construcción de especificaciones, y la evaluación preliminar de invariantes. Más aún, lenguajes como Alloy y DynAlloy, estudiados en este artículo, brindan flexibilidad en abstracciones de datos (conjuntos, relaciones built-in) y construcciones de programación (elección no determinista, iteración acotada), que pueden usarse para incorporar, por ejemplo, resolución de acertijos como parte de la enseñanza de la algoítmica, con soporte computacional (el uso de acertijos en la enseñanza de la programación es propuesto en varios trabajos, notablemente [11]). Sin embargo, nuestra propuesta no viene sin limitaciones. Incorporar lenguajes como Alloy y DynAlloy tienen sus dificultades, incluyendo el uso de un lenguaje adicional en el proceso de enseñanza aprendizaje. En nuestra opinión, su correcta adopción demanda la

construcción de interfaces que permitan a los estudiantes incorporar la tecnología sin necesidad de aprender nuevos lenguajes y abstracciones; la discrepancia entre las construcciones de (Dyn)Alloy y las construcciones de metodologías de construcción de programas como las utilizadas en [2, 3] (estas últimas centradas en el uso de expresiones cuantificadas con diferentes tipos de "cuantificación", numéricas por ejemplo) hacen al desarrollo de interfaces adecuadas una tarea a la vez indispensable y compleja.

Este artículo presenta un análisis de cuánto tienen para aportar herramientas de análisis basado en constraint solving como Alloy y DynAlloy. En cuanto a trabajo futuro, planeamos realizar estudios de campo que nos ayuden a construir interfaces adecuadas para la tecnología estudiada, incorporar las herramientas a asignaturas de enseñanza de la programación y evaluar el impacto de las mismas en los procesos de enseñanza-aprendizaje.

Referencias

1. J. Abrial, *The B-Book: Assigning Programs to Meanings*, Cambridge University Press, 2005.
2. R. Backhouse, *Program construction: calculating implementations from specifications*, Wiley, 2003.
3. D. Barsotti, J. O. Blanco, S. Smith, *Clculo de Programas*, Universidad Nacional de Crdoba, 2008.
4. A. Biere, M. Heule, H. van Maaren y T. Walsh, *Handbook of Satisfiability*: Volumen 185, Frontiers in Artificial Intelligence and Applications. IOS Press, Amsterdam, Holanda. 2009.
5. E. Clarke, O. Grumberg y D. Peled, *Model Checking*, MIT Press, 2000.
6. N. Eén, N. Sörensson, *An Extensible SAT-solver*, en Proc. de 6th International Conference SAT 2003, LNCS 2919, Springer, 2004.
7. D. Jackson, *Software Abstractions: Logic, Language, and Analysis*, The MIT Press, 2006.
8. M. Frias, J.P. Galeotti, C. López Pombo y N. Aguirre, *DynAlloy: upgrading alloy with actions*, en Proc. de International Conference on Software Engineering ICSE 2005, St. Louis, USA. ACM Press, 2005.
9. J. Jacky, *The Way of Z: Practical Programming with Formal Methods*, Cambridge University Press, 1997.
10. J. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, 1992.
11. A. Levitin, M. Papalaskari, *Using puzzles in teaching algorithms*, en Proc. 33rd Technical Symposium on Computer Science Education SIGCSE 2002, USA, 2002
12. M. W. Moskewicz, C. F. Madigan, Y. Zhao, L. Zhang, S. Malik, *Chaff: Engineering an Efficient SAT Solver*, en Proc. de 38th Design Automation Conference DAC 2001, ACM, 2001.
13. N. Tillmann y J. de Halleux, *PEX: White Box Test Generation for .NET*, en Proc. de 2nd. International Conference on Tests and Proofs TAP 2008, LNCS 4966, Springer, 2008.
14. T. Xie, N. Tillmann y J. de Halleux, *Educational software engineering: where software engineering, education, and gaming meet*, en Proc. de 3rd International Workshop on Games and Software Engineering: Engineering Computer Games to Enable Positive, Progressive Change, IEEE Press, 2013.

Recomendaciones Personalizadas para Aplicaciones de Soporte al Aprendizaje Ubicuo

Margarita M. Álvarez, Silvina I. Únzaga, Elena B. Durán

Instituto de Investigación en Informática y Sistemas de Información (IIISI)
Facultad de Ciencias Exactas y Tecnologías (FCEyT)
Universidad Nacional de Santiago del Estero (UNSE)
Avenida Belgrano (S) 1912, Santiago del Estero, 4200, Argentina
{alvarez; sunzaga; eduran}@unse.edu.ar

Resumen. El aprendizaje ubicuo es un nuevo modelo educativo, caracterizado por la movilidad de los estudiantes, quienes disponen de dispositivos para acceder a los recursos de aprendizaje a través de redes inalámbricas. Además, los procesos educativos están relacionados con la situación que involucra al estudiante, posibilitando el acceso a los recursos educativos adaptados de acuerdo al perfil y al contexto de aprendizaje. Para ofrecer servicios personalizados a los estudiantes en contextos ubicuos hemos desarrollado una Arquitectura dirigida por Modelos Ontológicos. En este artículo presentamos una estrategia, a implementar en el marco de la mencionada arquitectura, para personalizar recomendaciones destinadas a los estudiantes. Se presenta, además, una aplicación de esta estrategia a un caso de estudio. Se concluye que tanto el marco aportado por la Arquitectura, como la estrategia propuesta, son adecuados para guiar la personalización de recomendaciones cuando se desarrollan aplicaciones de apoyo al aprendizaje ubicuo.

Palabras Clave: Aprendizaje ubicuo, Personalización, Recomendaciones, Arquitectura dirigida por modelos, Ontologías.

1 Introducción

En los últimos años, diferentes dispositivos móviles, la tecnología de radio transmisión y la tecnología de sensores se están desarrollando rápidamente y utilizando en forma cada vez más profusa. Soportados en esta tecnología, dispositivos móviles tales como asistentes personales digitales (PDA-Personal Digital Assistants), teléfonos inteligentes, terminales de Internet y etiquetas inteligentes están dando forma a un nuevo ambiente denominado "ambiente ubicuo", donde los usuarios pueden acceder fácilmente a redes de banda ancha y a diferentes servicios [13].

Este gran desarrollo de la tecnología en comunicaciones y la emergencia de nuevos paradigmas en la Web, con sus consiguientes aplicaciones en el campo de la educación, permiten poner a disposición de los estudiantes una gran diversidad de recursos educativos, crear nuevos y variados entornos de formación, personalizar el aprendizaje y posibilitar la realización de un conjunto de actividades formativas desde cualquier lugar y desde cualquier dispositivo. Los avances descriptos, han dado lugar

al surgimiento del aprendizaje ubicuo (u-learning), que es un nuevo paradigma educativo que tiene lugar en un entorno de computación ubicua, que permite el aprendizaje del contenido correcto, en el lugar más apropiado, en el momento indicado y de la manera correcta.

Los ambientes de aprendizaje ubicuo superan las limitaciones de una clase o ambiente de aprendizaje tradicional, y extienden el aprendizaje haciendo realidad la idea de aprendizaje en cualquier lugar y en cualquier momento, permitiendo a las personas acceder a mejores experiencias de aprendizaje en sus ambientes de la vida diaria. El uso de dispositivos tales como teléfonos móviles y PDA generan nuevas oportunidades para los estudiantes permitiéndoles estar conectados intensamente. Por lo tanto, los contenidos educativos pueden accederse y las interacciones pueden concretarse donde los estudiantes lo necesiten, en diferentes campos de la vida diaria, sin restricción de espacio ni de tiempo [7].

En los ambientes ubicuos es fundamental proporcionar una educación personalizada. La personalización en el contexto de la Informática se refiere a la capacidad de un sistema o aplicación para adaptarse y satisfacer las necesidades de cada usuario; teniendo en cuenta, por ejemplo, su nivel de conocimientos, estilos de aprendizaje, habilidades cognitivas, ubicación actual, motivación, intereses, idioma preferido, etc.; y así lograr que los entornos de aprendizaje permitan al estudiante una experiencia más eficaz, conveniente y exitosa de aprendizaje [7]. Por otra parte, los sistemas de recomendaciones son herramientas que generan recomendaciones sobre un determinado objeto de estudio, a partir de las preferencias y opiniones dadas por los usuarios. El uso de estos sistemas se está poniendo cada vez más de moda en Internet debido a que son muy útiles para evaluar y filtrar la gran cantidad de información disponible en la Web con el objeto de asistir a los usuarios en sus procesos de búsqueda y recuperación de información [8]. No obstante, en la mayoría de los sistemas de recomendación existentes, se utiliza sólo la información referida a las preferencias del usuario para proporcionar servicios personalizados, dejando de lado la información contextual [14], que es altamente relevante en los ambientes ubicuos.

Nosotras hemos estado investigando cómo la computación ubicua y las técnicas de personalización podrían utilizarse para mejorar el aprendizaje y cómo desarrollar eficientemente entornos de u-learning. Para alcanzar este objetivo propusimos una Arquitectura dirigida por modelos ontológicos para aplicaciones de aprendizaje ubicuo [5]. Esta arquitectura se compone por seis módulos de software (*Módulo de Registración*, *Módulo de Interfaz de Usuario*; *Módulo de Adquisición de Contexto Ubicuo*; *Módulo de Análisis de Petición de Usuario*; *Módulo de Personalización*; *Módulo de Mantenimiento de los Modelos Ontológicos*) que interactúan con diferentes ontologías que representan e integran la información del estudiante, del dominio y del contexto, y en forma general se las ha denominado ONTO-AU [1].

En este artículo presentamos, en particular, una descripción de la estrategia diseñada para generar las recomendaciones personalizadas en el módulo de personalización, que forma parte de la arquitectura mencionada.

En las siguientes secciones se presenta una revisión de trabajos relacionados a la personalización de sistemas de aprendizaje ubicuo; se describe la estrategia de generación de recomendaciones personalizadas y su aplicación en un caso concreto; y, finalmente, se expresan las conclusiones y se delinean algunos trabajos futuros.

2 Trabajos Relacionados

Algunos proyectos están investigando el uso de la computación ubicua para proporcionar nuevas perspectivas de aprendizaje. En esta sección se presentan algunos antecedentes que sugieren la aplicación de técnicas de personalización en ambientes de aprendizaje ubicuo.

En [10] se describe un sistema de registro de aprendizaje ubicuo llamado SCROLL, cuyo objetivo es ayudar a los usuarios a capturar en los logs aquello que han aprendido, revisar y reflejar sus viejas bitácoras de aprendizaje, reutilizar el conocimiento cuando lo necesitan, recordar en el momento adecuado, en el lugar correcto y recomendar adecuadamente las bitácoras de aprendizaje de los demás. El sistema puede recomendar objetos de aprendizaje para un alumno determinado teniendo en cuenta tanto el contexto como sus necesidades de estudio.

Chia-Chen Chen y Tien-Chi Huang [4] proponen un sistema de aprendizaje ubicuo sensible al contexto, basado en la identificación por radiofrecuencia, red inalámbrica, dispositivo de mano embebido, y las tecnologías de bases de datos para detectar y examinar el comportamiento de aprendizaje de los estudiantes en el mundo real. Este sistema proporciona un proceso de aprendizaje personalizado que utiliza la secuencia curricular para generar un curso de aprendizaje personalizado para cada estudiante mediante la selección dinámica de materiales didácticos óptimos. El enfoque propuesto se basa en un examen formativo para recoger los conceptos de aprendizaje incorrectos de los alumnos a través de pruebas de adaptación computarizada extraídas de un banco de preguntas. Ellos emplean para construir un camino de aprendizaje casi óptimo los patrones de respuesta incorrectas del examen formativo y buscan los materiales didácticos pertinentes de la base de datos de los planes de estudio en función del grado de dificultad del currículo.

Won-Ik Park et al. [14] proponen una técnica de personalización sensible al contexto eficiente, teniendo en cuenta la situación y los gustos de los usuarios en entornos de computación ubicua. El método se basa en el empleo de una técnica híbrida de personalización que aplica personalización sensible al contexto y personalización basada en el perfil de usuario, usando ontologías, reglas y análisis multicriterio. La personalización sensible al contexto recomienda una lista de candidatos usando las preferencias definidas en la ontología y las reglas. Y luego, mediante la personalización basada en el perfil de usuario, recomienda una lista final de la lista de candidatos utilizando el análisis multicriterio.

Shu-Lin Wang y Chun-Yi Wu [12] proponen aplicar tecnología de contexto y algoritmos de recomendación para desarrollar un sistema de aprendizaje ubicuo que ayuda a que los estudiantes, en un aprendizaje a lo largo de la vida, se den cuenta de los objetivos de aprendizaje personalizados en forma consciente al contexto. Toma en cuenta las necesidades de aprendizaje de los alumnos en un contexto real, así como las diferencias en las preferencias personales. Se utilizó el filtrado colaborativo y reglas de asociación para el desarrollo del sistema. Los materiales de aprendizaje adaptativo se recomiendan para el aprendizaje a lo largo de la vida utilizando un modelo de minería de reglas de asociación con el fin de mejorar la motivación y la eficacia en el aprendizaje de los alumnos.

En Ovalle et al. [11] se propone un modelo de recomendación personalizada de recursos educativos para cursos virtuales adaptativos basados en agentes inteligentes. Las funcionalidades del prototipo son: planificación de sus cursos virtuales, evaluación en línea, búsqueda y recuperación de objetos de aprendizaje y Servicios de Awareness. El prototipo construido presenta agentes inteligentes proactivos y deliberativos que permiten la búsqueda y recomendación de información adaptada al perfil del estudiante. Se emplearon ontologías para la descripción de los conceptos necesarios para definir la estructura del perfil del usuario y las reglas para poder seleccionar de forma adaptativa los contenidos para planificar el curso virtual y la adaptación de los Objetos de Aprendizaje a las características del usuario.

3 Estrategia de generación de recomendaciones Personalizadas

A partir de considerar diferentes aproximaciones teóricas [2, 3, 6, 9] respecto a los tipos de adaptación o personalización en e-learning, hemos determinado los siguientes tipos de adaptación para el contexto de personalización de este trabajo (Fig. 1).

Figura 1: Tipos de adaptación

Adaptación Didáctica: consiste en adaptar el diseño instruccional a partir de un planeamiento riguroso en el que se establecen criterios respecto a: los *contenidos* con que se abarcarán las unidades de aprendizaje del dominio de conocimiento, el conjunto de a*ctividades* y la *ruta instruccional* a proponer al estudiante conforme a sus características y condiciones contextuales.

Adaptación de Interfaz: se fundamenta en la disposición del sistema para el estudiante con las adaptaciones de: *Presentación*, respecto a la disposición en que los materiales y actividades son provistos, con una apariencia e interacción conforme a las condiciones de cada estudiante; y de *Navegación*, que incluye la estructura de los enlaces (basado en web) o relaciones inter-objetos para el desplazamiento por el sistema.

Adaptación Colaborativa: consiste en adaptar las recomendaciones para las actividades colaborativas del estudiante, abarcando principalmente la recomendación de colaboradores acorde a las características personales y contextuales propias de los estudiantes y de sus posibles colaboradores (*pares* o *expertos*).

Por otra parte, hay varios métodos que se pueden implementar para hacer personalización en e-learning [11]. En esta investigación, para la personalización en u-learning, adoptamos una estrategia basada en un **enfoque de personalización híbrido** que combina la *personalización basada en perfiles de usuario*, que permite adaptar a las características relevantes de los estudiantes, con la *personalización sensible al contexto*, que realiza la adaptación en base a los aspectos que caracterizan una situación de aprendizaje particular, el ambiente donde se concreta y los dispositivos y medios que se usan. Se utiliza además un enfoque semántico mediante el uso de ontologías para el modelado tanto de los datos del perfil como del contexto. En ambos casos, la técnica aplicada para filtrar los aspectos relevantes a recomendar, son las reglas. En la Fig. 2 se muestra una visión global del enfoque aplicado en el Módulo de Personalización de la Arquitectura dirigida por modelos ontológicos para aplicaciones de aprendizaje ubicuo [5].

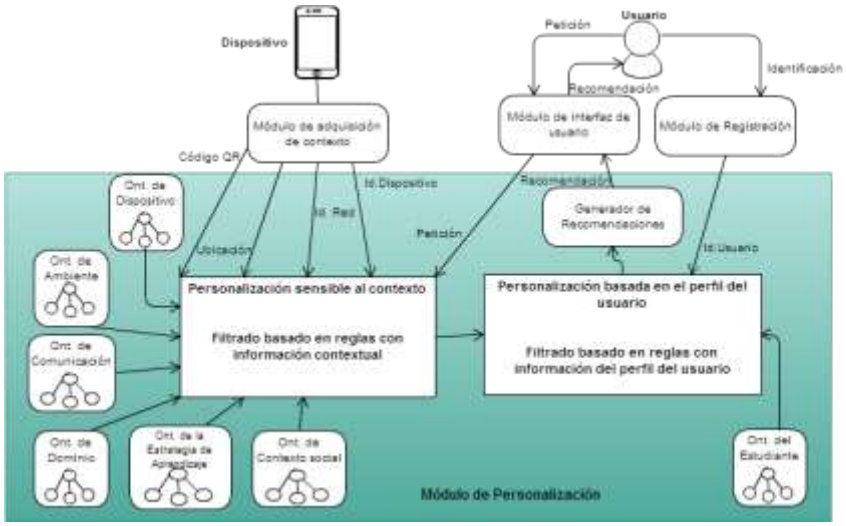


Figura 2: Visión global del enfoque aplicado en el Módulo de Personalización de la Arquitectura dirigida por modelos ontológicos para aplicaciones de aprendizaje ubicuo

4 Aplicación de la Estrategia a un caso de estudio

A continuación se muestra el uso de la estrategia descripta en la sección anterior, para diseñar recomendaciones personalizadas en una aplicación de apoyo al aprendizaje ubicuo que brinda cinco tipos de servicios. Los mismos son:

1- *Servicio 1*: Recomendar Objetos de Aprendizaje para puntos de interés seleccionados por el estudiante.
2- *Servicio 2:* Recomendar un camino de aprendizaje a partir de un objetivo de aprendizaje seleccionado por el estudiante.
3- *Servicio 3*: Recomendar puntos de interés cercanos a la ubicación del estudiante.

4- *Servicio 4*: Recomendar expertos para que asesoren al estudiante para cumplir con una tarea.
5- *Servicio 5*: Recomendar pares para que asesoren al estudiante para cumplir con una tarea.

Recomendaciones personalizadas para el servicio 1. El estudiante desea aprender acerca de un Punto de Interés (PI) determinado para lo cual sensa el código QR asociado al PI, usando su dispositivo móvil, y el sistema recomienda los Objetos de Aprendizaje (OA) apropiados, representados en la Ontología del Dominio. Para poder realizar esta recomendación se aplica personalización sensible al contexto para recomendar una lista de OA candidatos relacionados al PI y a las características técnicas del dispositivo del estudiante, representadas en la Ontología de Dispositivo. Luego, aplicando la técnica de personalización basada en el perfil del usuario, el sistema recomienda una lista final de OA candidatos de acuerdo con el estilo de aprendizaje del estudiante, representado en la Ontología del Estudiante. El siguiente ejemplo muestra las reglas que generan la lista de OA candidatos.

Ejemplo 1 :
(Defrule Servicio1)
L1 = (Select **OA** with Id_OA = Id_Punto-Interes AND Formato_OA = "video/mpeg" AND Tamaño_OA < "4200" AND Tipo_Plataforma_OA = SO AND Nombre_Platforma = "android" AND Tipo_Red_OA = "wifi" AND Tipo_Hardware_OA = "tablet 10")
L2 = (Select from L1 **OAs** with Tipo_Recurso_Educativo_OA = "simulation" OR "diagram" OR "figure" OR "graphic" OR "slide" AND Tipo_interactividad_OA = "active" AND Nivel_Interactividad_OA = "high" OR "very high" AND Dificultad_OA = "easy")
Esta regla filtra los OA que cumplan con las siguientes condiciones:

A) Estén relacionados con el PI cuyo código QR fue sensado por el dispositivo móvil del estudiante.
B) Sus características técnicas coincidan con las características técnicas del dispositivo usado por el estudiante (Tablet con pantalla de 10", Sistema Operativo ANDROID y software que puede mostrar videos con formato mpeg).
C) Sus características pedagógicas coincidan con el estilo de aprendizaje (visual-activo) y nivel de conocimiento (bajo).

Recomendaciones de un camino de aprendizaje a partir de un objetivo de aprendizaje seleccionado por el estudiante. El estudiante selecciona un objetivo de aprendizaje y el sistema lo guía basado en su ubicación actual, su perfil de usuario e historia de aprendizaje (modelados en la Ontología del estudiante) y los datos recolectados por los sensores de su dispositivo móvil.

Para realizar este servicio se usará un algoritmo heurístico para determinar un camino de aprendizaje personalizado, que consiste en determinar los PI que el estudiante visitará buscando el mejor camino de aprendizaje. El siguiente ejemplo muestra las reglas que generan el camino de aprendizaje personalizado.

Ejemplo 2
(Defrule Servicio 2)
L1=(Select Tema-EA with Id-Objetivo-EA = Objetivo-seleccionado)
Por cada tema de L1 aplicar
L2 = (Select Tema-D with Id_Tema_D = Id_Tema L1 AND Select PI with Id_PI_Tema_D = Id_PI)
L3 = (Select FROM L2 PI with Estado_PI_ES = "no realizado")
L4 = (Ordenar L3 haciendo:
{Calcular distancia de PI en relación a ubicación del alumno
Ordenar distancias de menor a mayor})

Esta regla selecciona una secuencia de temas (L1) desde la Ontología de Estrategia de Aprendizaje (EA) que corresponda con el objetivo seleccionado por el estudiante. Luego por cada tema de la secuencia recupera desde la ontología del dominio (D) los puntos de interés asociados al mismo (L2). A continuación, la regla selecciona de L2 aquellos PI que el estudiante no visitó (L3). Para cada PI calcula la distancia en relación a la posición actual del estudiante y genera una lista por orden creciente de estos PI.

Recomendaciones personalizadas de puntos de interés cercanos a la ubicación del estudiante. El sistema recomienda los PI más cercanos a la ubicación actual del estudiante considerando su historia de aprendizaje modelada en la Ontología del Estudiante. Un ejemplo de regla que genera la lista de PI candidatos se muestra a continuación.

Ejemplo 3:
(Defrule Servicio 3)
L1 = (Select Puntos_Interes with (DIFERENCE BETWEEN (Ubicacion_Estudiante, Ubicacion_Puntos_Interes) <= 10 mts.))
L2 = (Select FROM L1 PI with Estado_PI_ES = "no realizado")
Esta regla filtra los PI cuya distancia a la ubicación del estudiante es menor que 10 mts. (L1). De allí, selecciona los PI no realizados por el estudiante (L2).

Recomendaciones de expertos para que asesoren al estudiante para cumplir con una tarea. En este servicio el estudiante solicita el asesoramiento o colaboración de un experto en la tarea que debe realizar. El sistema le recomienda los expertos en el tema, que estén en línea y además, los expertos que se encuentren físicamente cerca de la ubicación actual del estudiante. Para ello se consulta a la Ontología de Contexto Social. Un ejemplo de regla que genera las listas de expertos candidatos se muestra a continuación.

Ejemplo 4:
(Defrule Servicio 4)
L1 = (Select Expertos with Tema-CS = Tema-seleccionado)
L2 = (Select FROM L1 Expertos with Estado-CS = "conectado")
L3 = (Select FROM L1 Expertos with (DIFERENCE BETWEEN ((Ubicacion_Estudiante, Ubicacion_Experto) <= 10 mts.))
Esta regla genera una primera lista con los Expertos del Contexto Social (CS) cuyo tema de experticia se corresponda con el tema en el que está trabajando el alumno. De esa lista se filtran para L2 aquellos expertos que cumplan con la condición de estar en línea. Para L3 se filtran de L1 los expertos que se encuentren a una distancia menor a 10 mts. de la ubicación actual del estudiante.

Recomendaciones de pares para que asesoren al estudiante para cumplir con una tarea. En este servicio el estudiante solicita el asesoramiento o colaboración de un compañero estudiante que ya haya realizado la tarea que él pretende completar. Para ello, a partir de la identificación del estudiante en el sistema y de la tarea que quiere realizar, aplicando una técnica de personalización basada en el perfil del usuario y la historia de aprendizaje, el sistema recomienda una lista de compañeros candidatos que estén en línea en ese momento y/o que se encuentren físicamente cerca de la ubicación actual del estudiante. Un ejemplo de regla que genera las listas de compañeros candidatos se muestra a continuación.

Ejemplo 5:
(Defrule Servicio 5)
L1 = (Select Estudiante with Id-Curso-Estudiante= Id-Curso AND Estado-Tarea-ES = "aprobado")
L2 = (Select FROM L1 Estudiante with Estado-ES = "conectado")
L3 = (Select FROM L1 Estudiante with (DIFERENCE BETWEEN ((Ubicacion_Estudiante, Ubicacion_Par) <= 10 mts.))
Esta regla genera una primera lista (L1) con los Pares que pertenecen al mismo curso del estudiante y tienen aprobada la tarea que el estudiante ha seleccionado para realizar. De L1 se filtran los pares que se encuentren conectados al sistema en ese momento para conformar L2. L3 está formada por los pares que se encuentren a una distancia menor a 10 mts. de la ubicación actual del estudiante.

En la Tabla 1 se resumen los tipos de adaptación aplicados, considerando lo definido al comienzo de esta sección, para cada uno de los servicios.

Tabla 1: Resumen de adaptaciones

SERVICIO	TIPO DE PERSONALIZACIÓN	TÉCNICA DE PERSONALIZACIÓN
1	• Adaptación didáctica de contenido • Adaptación de interfaz: Presentación	Híbrida que combina Personalización sensible al contexto y Personalización basada en el perfil del usuario
2	• Adaptación didáctica de secuencia	Algoritmo Heurístico para determinar camino de aprendizaje
3	• Adaptación didáctica de contenido	Híbrida que combina Personalización sensible al contexto y Personalización basada en el perfil del usuario
4	• Adaptación de colaboración	Híbrida que combina Personalización sensible al contexto y Personalización basada en el perfil del usuario
5	• Adaptación de colaboración	Híbrida que combina Personalización sensible al contexto y Filtrado Colaborativo

5 Conclusiones y Trabajos Futuros

En un ambiente de aprendizaje ubicuo es fundamental considerar los elementos que involucran al estudiante en el mundo real, ya que estos condicionan el contenido a

aprender, las actividades y la secuencia de aprendizaje, entre otras cosas. En consecuencia, uno de los principales problemas a considerar en el diseño de aplicaciones que den soporte a este tipo de aprendizaje es la percepción del contexto en el que se encuentra el estudiante.

En este artículo, hemos mostrado cómo la tecnología de la computación ubicua y las técnicas de personalización pueden ser útiles en estos nuevos escenarios de aprendizaje. En particular, hemos presentado una estrategia para generar recomendaciones en un sistema de soporte al aprendizaje ubicuo, implementable en el Módulo de Personalización de la Arquitectura Dirigida por Modelos Ontológicos.

Se pueden destacar las siguientes conclusiones en relación a la estrategia propuesta y su aplicación a un caso de estudio:

- La Arquitectura dirigida por Modelos Ontológicos aporta un marco adecuado para la aplicación de la estrategia propuesta.
- La estrategia para la generación de recomendaciones basada en un enfoque de personalización híbrido, resultó adecuada al ser aplicada a un caso particular, permitiendo adaptar los contenidos, el camino de aprendizaje, la interfaz y la colaboración, tanto a las características relevantes de los estudiantes como a los aspectos que caracterizan una situación de aprendizaje particular, el ambiente donde se concreta y los dispositivos y medios que se usan.
- El enfoque semántico aplicado, con el uso de ontologías para el modelado tanto de los datos del perfil como del contexto, resultó útil ya que permitió instanciar para un caso concreto los aspectos generales de cada ontología.
- La técnica aplicada para filtrar los aspectos relevantes a recomendar, basada en reglas, resultó eficiente permitiendo especificar claramente las condiciones de personalización tanto del contexto como del perfil del estudiante.

Actualmente, estamos trabajando en la implementación del caso particular para una aplicación ubicua que asista el aprendizaje en el curso de ingreso universitario, utilizando el software Protege para desarrollo de las ontologías.

Referencias

1. Alvarez, M. M; Duran, Elena B. and Unzaga, S.I. ONTO-AU: Una ontología para sistemas de apoyo al aprendizaje ubicuo. VIII Jornadas de Ciencia y Tecnología de Facultades de Ingeniería del NOA. V, 143-149. Tucumán (2012)
2. Brusilovsky P. Methods and techniques of adaptive hypermedia. User Modeling and User Adapted Interaction, 1996, v 6, n 2-3, pp 87-129
3. Brusilovsky, P. Adaptive Hypermedia, User Modeling and User-Adapted Interaction, vol. 11, pp. 87-110, 2001.
4. Chia-Chen Chen and Tien-Chi Huang. Learning in a u-Museum: Developing a context-aware ubiquitous learning environment. Computers & Education 59, 873–883 (2012)
5. Durán, E.B.; Alvarez, M. M. and Unzaga, S.I. Ontological Model-driven Architecture for Ubiquitous Learning Applications. 7th Euro American Association on Telematic and Information Systems (EATIS 2014), Valparaiso, Chile, 2 al 4 de Abril de 2014. Proceedings published by ACM Digital Library within its International Conference Proceedings Series, ISBN 978-1-4503-2435-9. Disponible en: http://dl.acm.org/

citation.cfm?id=2590776&dl=ACM&coll=DL&CFID=356022841&CFTOKEN=38496822 (2014)

6. García, V. Personalisation in Adaptive E-Learning Systems. A Service Oriented Solution Approach for Multi-Purpose User Modelling Systems, Doctorado, Computer Science, Institute for Information Systems and Computer Media (IICM). Graz University of Technology, Graz, 2007.
7. Graf Sabine and Kinshuk. Adaptivity and Personalization in Ubiquitous Learning Systems. A. Holzinger (Ed.): USAB 2008, LNCS 5298, pp. 331–338, 2008. Springer-Verlag Berlin Heidelberg (2008)
8. Herrera-Viedma E., Porcel C. and L. Hidalgo L. Sistemas de recomendaciones: herramientas para el filtrado de información en Internet [en linea]. "Hipertext.net", núm. 2, 2004. http://www.hipertext.net
9. Kobsa A., Koenemann J. and Pohl G. Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships, The Knowledge Engineering Review, vol. 16, pp. 111-155, 2001.
10. Mengmeng Li, Hiroaki Ogata, Bin Hou, Noriko Uosaki, Yoneo Yano Hwang, Tsai, and Yang. Personalization in Context-aware Ubiquitous Learning-Log System. Seventh IEEE International Conference on Wireless, Mobile and Ubiquitous Technology in Education. 978-0-7695-4662-9/12. IEEE DOI 10.1109/WMUTE.2012.14. pp 41-48 (2012)
11. Ovalle, D.; Salazar, O. & Duque, N. Modelo de Recomendación Personalizada en Cursos Virtuales basado en Computación Ubicua y Agentes Inteligentes. Información Tecnológica Vol. 25(6), 131-142 (2014)
12. Shu-Lin Wang and Chun-Yi Wu. Application of context-aware and personalized recommendation to implement an adaptive ubiquitous learning system. Expert Systems with Applications 38 10831–10838 (2011)
13. Weiser, M. The computer for the 21st century. Scientific American, 265(3), 94–104 (1991).
14. Won-Ik Park, Jong-Hyun Park, Young-Kuk Kim and Ji-Hoon Kang. An Efficient Context-Aware Personalization Technique in Ubiquitous Environments. ICUIMC '10 Proceedings of the 4th International Conference on Uniquitous Information Management and Communication. Article No. 60 ACM. New York, USA. ISBN: 978-1-60558-893-3 (2010).

Modelo Bayesiano para el Diagnóstico del Aprendizaje en Alumnos de Inteligencia Artificial

Pablo Pytel[1], Cinthia Vegega[1], Ariel Deroche[1], Mariana Acosta[1] & María Florencia Pollo-Cattaneo[1]

[1] Grupo de Estudio en Metodologías de Ingeniería en Software (GEMIS). Universidad Tecnológica Nacional. Facultad Regional Buenos Aires. Argentina.
{ ppytel , marianapaolacosta, flo.pollo }@gmail.com

Resumen. La comunidad educativa ha encontrado en las TIC una herramienta muy útil para asistir el proceso de enseñanza-aprendizaje. Entre sus principales tareas se encuentran la monitorización y el diagnóstico del nivel de conocimiento que poseen los alumnos. Sin embargo, estas tareas requieren el diseño de sistemas informáticos donde deben aplicarse principios y técnicas de la Inteligencia Artificial. En tal sentido, las Redes Bayesianas son un tipo de Sistema Inteligente que permite generar un modelo probabilístico a partir de la combinación de información histórica disponible y la experiencia de los docentes. De esta manera, es posible identificar el estilo de aprendizaje, el cual representa la forma en que los estudiantes adquieren y entienden los temas dictados. En este contexto, el presente trabajo se propone analizar los resultados de aplicar dicha tecnología en una asignatura de grado y así asistir a los docentes a comprender mejor el comportamiento de sus alumnos.

Palabras Claves: Educación y Tecnología. Formación de Ingenieros. Estilos de Aprendizaje. Redes Bayesianas.

1. Introducción

Dentro del marco del Plan Estratégico de Formación de Ingenieros (PEFI) 2012–2016 se ha identificado, como una de las acciones, la aplicación de Tecnologías de la Información y la Comunicación (TIC) para la enseñanza y el aprendizaje [1]. Esto ha motivado a que la comunidad educativa haya encontrado en las TIC una herramienta muy útil para asistir al proceso de enseñanza-aprendizaje [2]. Entre sus principales tareas se encuentra la monitorización del aprendizaje y la evaluación continua junto con un apoyo técnico y pedagógico permanente [3].

En ese sentido, poder diagnosticar el nivel de conocimiento o dominio que poseen los alumnos con respecto sobre los contenidos de la asignatura es una función fundamental para entender y modelar el comportamiento de los estudiantes [4]. Aunque cada persona tiene un estilo de aprendizaje propio, la incorporación en el proceso pedagógico de recursos y estrategias que abarquen los principales perfiles de los alumnos es una herramienta que le va a permitir al docente intervenir positivamente en su interés y motivación [5]. De esta manera, será posible involucrar a los estudiantes en el proceso pedagógico y lograr una mejora significativa en su

rendimiento académico. Sin embargo, la automatización de estas tareas requiere el diseño de sistemas informáticos realmente adaptativos capaces de organizarse de manera autónoma y rápida a la multiplicidad de variables que forman parte de un contexto formativo. Por lo tanto, no pueden ser alcanzados a través del uso de sistemas software convencionales, sino que deben aplicarse principios y técnicas de la Inteligencia Artificial [6].

En este contexto, los docentes de una asignatura de grado han decidido implementar un Modelo Bayesiano que permita diagnosticar el grado de entendimiento o conocimiento que tienen sus alumnos. Así, además de identificar las características que tienen los estudiantes, será posible implementar un mecanismo para recomendarle (al alumno) la manera en que deben organizar sus tiempos de estudio para poder regularizar la materia. Para ello, en este trabajo, primero se describe las características principales de la asignatura considerada (sección 2) y de la tecnología aplicada (sección 3). Luego, se presenta el modelo generado (sección 4) con el correspondiente análisis de sus resultados (sección 5). Finalmente, se indican las conclusiones y futuras líneas de trabajo (sección 6).

2. Descripción del Contexto

En el quinto nivel de la carrera 'Ingeniería en Sistemas de Información' de la Universidad Tecnológica Nacional se encuentra la asignatura 'Inteligencia Artificial'. Esta asignatura tiene como objetivo introducir al alumno en el estudio de la Inteligencia Artificial (IA) y su aplicación en el abordaje de situaciones que se presentan en su actividad profesional [7].

Para los docentes de la asignatura en la Facultad Regional Buenos Aires, esto implica involucrar al alumno con las diversas tecnologías que brinda la IA para la construcción de Sistemas Inteligentes [8], entre las que se destacan los Sistemas Basados en Conocimiento (SBC), los Sistemas Expertos (SE), las Redes Neuronales Artificiales (RNA) y los Algoritmos Genéticos (AG). Complementado con estas arquitecturas, dentro del temario de la asignatura, también se incluyen los aspectos ingenieriles a dicha construcción, los cuales se engloban dentro de la 'Ingeniería del Conocimiento' o INCO [9]. En tal sentido, el equipo docente hace hincapié en las técnicas asociadas a la Adquisición de Conocimientos tales como Emparrillado y Análisis de Protocolos (AP). Asimismo, se desarrollan conceptos asociados a los Métodos de Búsqueda y Lógica de Primer Orden.

Durante el cuatrimestre, todos estos temas son desarrollados en clase siguiendo una planificación definida previamente por la cátedra. Al finalizar cada clase, los alumnos reciben la lista de apuntes correspondiente al tema que deben estudiar y un conjunto de ejercicios para practicar. Para regularizar la cursada se deben aprobar dos trabajos prácticos grupales y un examen integrador:

- Por una parte, cada grupo (entre 3 y 6 integrantes) debe presentar la resolución de un AP como primer trabajo práctico; y, como segundo, un informe sobre la construcción de un Sistema Inteligente (RNA o AG) que se aplique a la solución de un problema seleccionado por los alumnos. Para mayor información sobre este trabajo práctico se recomienda consultar [10].

- Por otra parte, cada alumno debe aprobar un examen donde se evalúan todos los temas (teóricos y prácticos) desarrollados mediante 21 preguntas de opción múltiple. Mayor detalle sobre la estructura de este examen se puede encontrar en [11; 12]. En caso de no aprobar esta evaluación parcial, el alumno podrá rendir hasta dos instancias de recuperatorio con preguntas y ejercicios similares.

Uno de los problemas identificados en la asignatura es la falta de estudio de los alumnos durante el cuatrimestre. Dado que la mayoría están próximos a finalizar sus estudios, ya se desempeñan laboralmente, lo cual acota los tiempos libres debido a la extensión de la jornada laboral y los traslados hogar/facultad/lugar de trabajo. Por ese motivo, la gran mayoría tiende a estudiar sólo en las fechas próximas a los exámenes, imposibilitando el alcance de niveles de profundidad conceptual adecuados para la aprobación. Esto ha motivado que el equipo docente haya decidido desarrollar un recurso didáctico y tecnológico que aproveche los tiempos ociosos con los que cuenta todo estudiante en cualquier lugar en donde se encuentre [13]. Basado en tecnología móvil, la aplicación IApp tiene como objetivos facilitarle al alumno el proceso de repaso, autoevaluación, corrección y afianzamiento del aprendizaje, y brindarle a los docentes información sobre el estado de dicho proceso. Para lograr una exitosa inserción de la aplicación, y mejorar así el rendimiento académico del alumnado, es imprescindible identificar cuáles son las características de los estilos de aprendizaje y su relación con el contenido de la asignatura. De esta manera, se podrá contar con información útil y válida para que la cátedra pueda tomar las decisiones correspondientes.

3. Materiales y Métodos

Para llevar a cabo el estudio propuesto se utilizan las Redes Bayesianas [14]. Este tipo de Sistema Inteligente permite la implementación de un grafo acíclico dirigido, el cual combina la potencia del Teorema de Bayes con la expresividad de las redes semánticas para generar predicciones causales. Una de sus ventajas es que, además de poder ser construida en forma automática a partir de un conjunto de observaciones previas, tiene la capacidad de incorporar conocimientos especializados del dominio junto con dicha información histórica [6]. En tal sentido, se logra obtener un sistema especializado, denominado Sistema Experto, el cual es capaz de hacer inferencias semejantes a las humanas.

Aunque las Redes Bayesianas surgen del campo de la IA, gracias a su gran flexibilidad se aplican a múltiples campos y disciplinas. Dentro del ámbito educativo existen muchos trabajos tales como [6; 15-19] donde se aplica para el modelado del estilo de aprendizaje de estudiantes. A partir de los resultados cuantitativos y cualitativos generados por el modelo, es posible representar el conocimiento (o entendimiento) que los alumnos tienen sobre los contenidos de un curso o asignatura. De este modo, se puede estudiar al alumno desde el punto de vista de lo que éste ya sabe para ser contrastado con el conocimiento experto de los docentes [19]. Dado que cuando el estilo de enseñanza coincide con las características de los estudiantes se logra un mayor grado de aprendizaje [15], está información será muy útil para que docentes planifiquen e implementen futuras estrategias pedagógicas.

4. Modelo Bayesiano

Para la construcción del Modelo Bayesiano propuesto se han empleado datos históricos de exámenes parciales y trabajos prácticos, junto con un cuestionario teórico llevado a cabo dos semanas antes del examen parcial. A partir de estos datos, se han generado redes bayesianas parciales que luego fueron integradas en la red que se muestra en la Figura 1. A su vez, usando la información suministrada por los docentes, a esta red se le han agregado los nodos indicados en la Figura 2. Dichos nodos permiten predecir el grado de conocimiento o entendimiento que tiene el alumno sobre cada tema y, de esa manera, poder recomendar los tiempos de estudio correspondientes. La descripción detallada de los datos, así como del trabajo realizado para la construcción y validación del modelo, se encuentra en el reporte técnico indicado en [20].

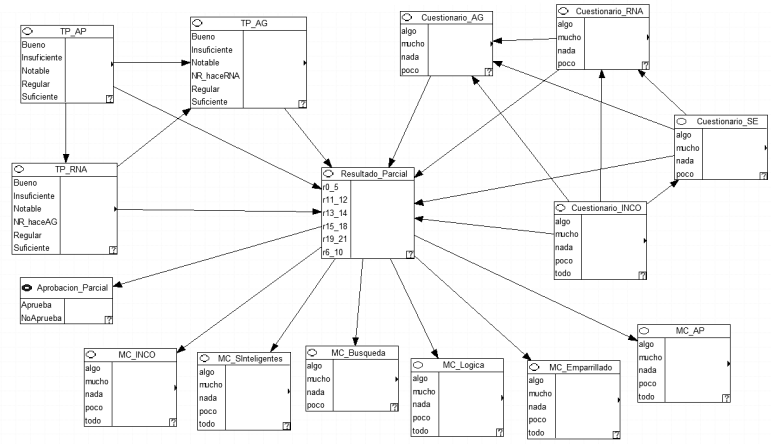


Fig. 1. Red Bayesiana integrada para predecir el desempeño del alumno en el examen parcial.

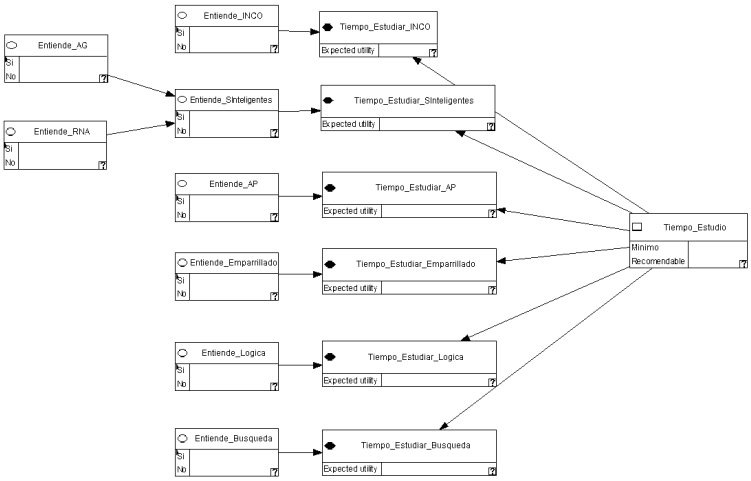


Fig. 2. Nodos utilizados predecir el grado de entendimiento y recomendar tiempo de estudio.

Como se puede observar en la Figura 1, a partir de los nodos asociados a las notas de los trabajos prácticos (nodos TP_AP, TP_RNA y TP_AG) y al desempeño en el cuestionario (Cuestionario_AG, Cuestionario_RNA, Cuestionario_SE y Cuestionario_INCO) es posible predecir las posibilidades que tiene el alumno de aprobar el examen parcial (nodos Resultado_Parcial y Aprobacion_Parcial). Estas probabilidades luego son propagadas para determinar el desempeño que tendrá el alumno en cada uno de los temas evaluados (MC_INCO, MC_SInteligentes, MC_Busqueda, MC_Logica, MC_Emparrillado y MC_AP). Luego, aplicando la experiencia y conocimientos de los docentes, se han incorporados nuevos nodos, que usando los resultados de los nodos anteriores, permiten pronosticar el grado de entendimiento que tiene el alumno sobre cada tema (nodos Entiende_INCO, Entiende_AG, Entiende_RNA, Entiende_SInteligentes, Entiende_Busqueda, Entiende_Logica, Entiende_Emparrillado y Entiende_AP que se muestran en la Figura 2). Con estos pronósticos, finalmente, es posible estimar el tiempo mínimo de estudio que debería emplear el alumno para entender los conceptos principales del tema (tiempo 'Minimo') y un tiempo adicional recomendable para entender el tema completamente (tiempo 'Recomendable'). Estos tiempos son indicados por los nodos Tiempo_Estudiar_INCO, Tiempo_Estudiar_SInteligentes, Tiempo_Estudiar_Busqueda, Tiempo_Estudiar_Logica, Tiempo_Estudiar_Emparrillado y Tiempo_Estudiar_AP, los cuales son acumulados en el totalizador Tiempo_Estudio.

5. Discusión de los Resultados del Modelo Bayesiano

Con el Modelo Bayesiano debidamente construido y validado [20], se lo emplea para diagnosticar el grado de aprendizaje de los alumnos de la asignatura, y así identificar mayor información sobre su estilo de estudio. Para ello, primero se analiza el estilo normal de aprendizaje de un alumno que cursa la asignatura (sección 5.1) y luego de los comportamientos de casos extremos que pueden ocurrir (sección 5.2).

5.1. Análisis del Estilo Normal de Aprendizaje de los Alumnos

Para analizar el comportamiento normal de un alumno que cursa la asignatura se le solicita al modelo que realice una predicción sin haberle definido previamente ninguna evidencia en particular sobre un alumno. Esto genera que se propaguen las probabilidades definidas a priori en base a la información suministrada y se obtengan los resultados que se pueden observar en las Figuras 3 y 4. Como se puede notar en la primer figura, todos los alumnos tienden a aprobar el examen parcial (61% en el valor 'aprueba' del nodo Aprobacion_Parcial), y con una buena nota ya que obtienen entre 15 y 18 respuestas correctas (con un 36% en el nodo Resultado_Parcial). Este comportamiento es conocido y se debe principalmente a los trabajos realizados anteriormente por la cátedra en [10-12] donde han logrado que la mayoría de los alumnos aprueben la evaluación parcial. Sin embargo, a este hecho, el modelo aporta mayor detalle sobre otros aspectos que se describen a continuación:

- Con respecto a los trabajos prácticos:

  Se puede observar que, en general, los grupos de alumnos presentan trabajos prácticos satisfactorios siendo muy pocos los desaprobados (valor 'insuficiente' en los nodos TP_AP, TP_AG y TP_RNA). Además, se puede notar que el de Análisis de

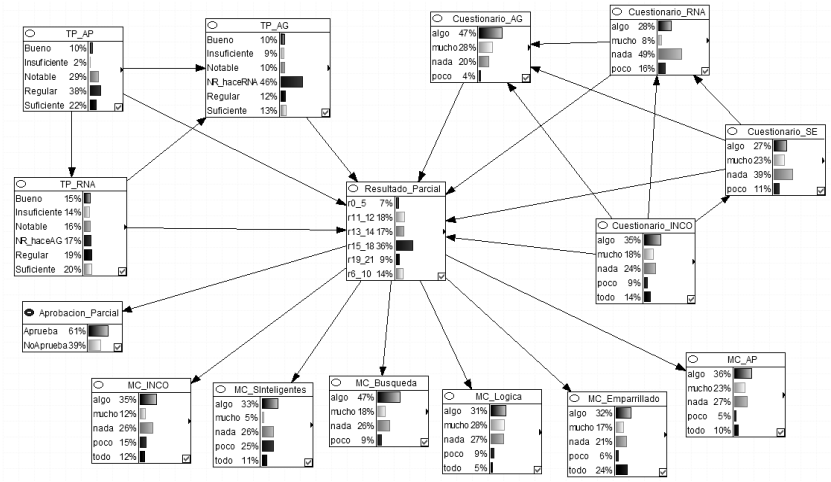


Fig. 3. Diagnóstico del comportamiento normal de un alumno que cursa la asignatura.

Protocolos tiende a recibir mejores notas (valor 'notable'). Por otra parte, se observa que la mayoría de los alumnos prefieren llevar a cabo la implementación de una Red Neuronal Artificial antes que un Algoritmo Genético. Según lo sugerido por los docentes, esto se debe a que la primera tecnología requiere menor tiempo de codificación que la segunda (sobre todo al aplicar herramientas ya disponibles).

- Con respecto al cuestionario tomado antes del examen parcial:

  Es importante destacar el bajo rendimiento de los alumnos en estas preguntas teóricas. Para los temas Redes Neuronales Artificiales y Sistemas Expertos hay una probabilidad del 49 y 39% respectivamente de no saber 'nada' del tema (esto es equivalente a haber respondido hasta una sola respuesta del tema), mientras que para Algoritmos Genéticos e Ingeniería del Conocimiento se podría decir que saben 'algo' del tema (lo que equivale a dos respuestas correctas de cada tema).

  Lo notable es que, a pesar de esta aparente falta de entendimiento, el alumno igual tiene buenas probabilidades de aprobar la evaluación parcial. Se presume que esto se debe a que, como se mencionó anteriormente en la sección 2, los alumnos comienzan a estudiar sólo cuando se aproxima la fecha del examen. Esto significa que, en menos de dos semanas, logran aprender lo mínimo necesario de los contenidos teóricos y prácticos para aprobar el parcial. Más allá que esta situación puede ser afortunada para los alumnos, no es satisfactoria desde el punto de vista pedagógico ya que los conocimientos adquiridos son muy débiles y temporales (es muy probable que en poco tiempo, olviden lo poco que aprendieron).

- Con respecto a la distribución del desempeño en los temas del parcial:

  Para la mayoría de los temas, los alumnos tienden a saber entre 'algo' y 'nada'. En otras palabras, un alumno suele contestar en promedio la mitad de las preguntas de cada tema, lo cual le es suficiente para aprobar. Esto confirmaría lo indicado en el párrafo anterior. No obstante, en los ejercicios de Lógica de Primer Orden, Emparrillado y Análisis de Protocolos puede suceder que se responda correctamente la mayoría y, en algunos casos, todas las preguntas. Dado que estos ejercicios son variados y no son sencillos (un alumno respondiendo al azar no tiene

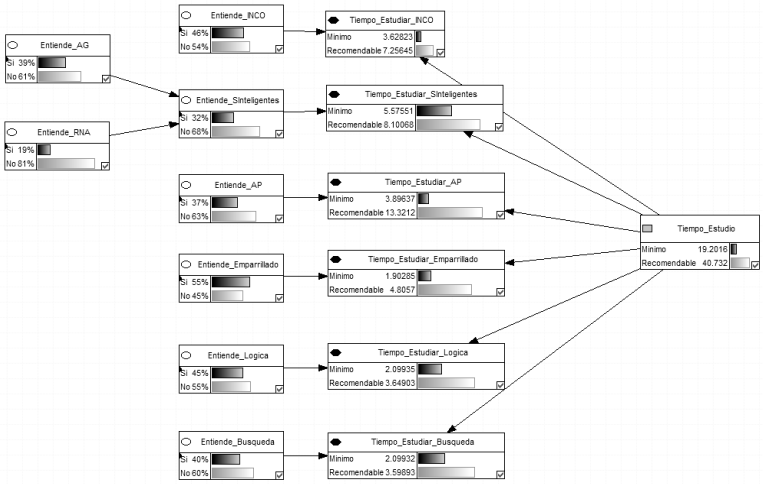


Fig. 4. Tiempos de estudio estimados según el comportamiento normal de un alumno.

oportunidades de acertar la mayoría de las respuestas, y menos todas), se concluye que, al momento de estudiar, le dedican mucho tiempo a la práctica de estos temas.

- Con respecto al grado de entendimiento de los temas:
  Si se observa la Figura 4, se detecta que el alumno promedio no logra entender satisfactoriamente ninguno de los temas dictados. En general, no se puede saber en forma concluyente si entiende, o no, el tema por ser la diferencia de probabilidades en el orden del 10%. Al contrario, no cabe duda que los alumnos tienen problemas con Sistemas Inteligentes y Métodos de Búsqueda (por ser la probabilidad de que haya entendido del 32 y 40% respectivamente).
- Con respecto a los tiempos estimados de estudio:
  Finalmente, considerando los resultados anteriores, el modelo estima que se debería emplear entre 19 y 41 horas para profundizar en el contenido de la asignatura y de esta forma reforzar el entendimiento en todos los temas. Si se considera que en un cuatrimestre hay alrededor de 12 semanas de clases antes del examen parcial, esto implica que un alumno que comienza a estudiar desde la primera clase debería estudiar entre 1½ y 3½ horas por semana, lo cual es razonable. En cambio, si empieza en las últimas dos semanas, debería emplear aproximadamente esa misma cantidad de horas *por día*, lo cual se requiere un gran esfuerzo y es poco viable.

5.2. Análisis del Estilo de Aprendizaje de Casos Extremos de Alumnos

Como última parte del análisis propuesto, se analiza el comportamiento de dos tipos extremos de alumnos. Por una parte, se analizan las predicciones para un alumno que al momento de responder el cuestionario no sabía nada de la materia (indicadas en la Figura 5); y, por otra parte, el caso del alumno que había estudiado antes del cuestionario por lo que respondió correctamente todas las preguntas (Figura 6).

En el primer caso, al indicar como evidencia que el estudiante tiene el valor 'nada' en cada uno de los cinco nodos correspondientes al cuestionario se obtienen los resultados que se muestran en la Figura 5. Como se observa, este tipo de alumno tiene

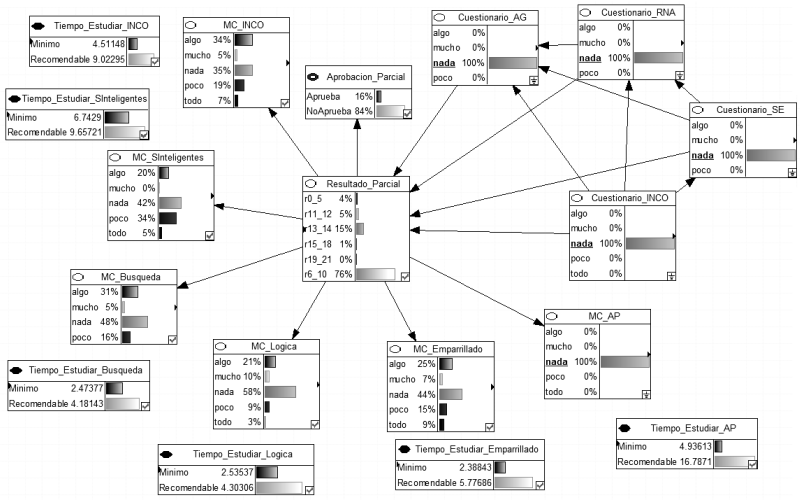


Fig. 5. Predicciones para un alumno que no sabía nada al responder el cuestionario.

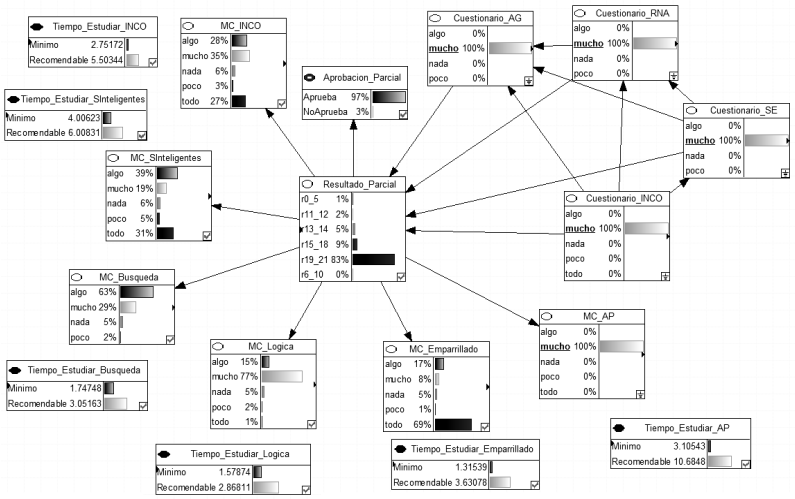


Fig. 6. Predicciones para un alumno que sabía todo al responder el cuestionario.

una probabilidad casi igual al 85% de no aprobar el examen parcial (con más del 75% de tener menos de 11 respuestas correctas). Esto le permite a los docentes confirmar que regularizar la cursada no es demasiado fácil y que sólo aprueban los alumnos que hayan adquirido algún conocimiento sobre los temas tratados. Dicha situación es muy útil e interesante porque también aplica a los estudiantes que comienzan a cursar la asignatura por primera vez. Si éstos no estudian nada durante el cuatrimestre, seguramente deberán recursarla. En cuanto al tiempo de estudio recomendado, para un alumno en esta situación se considera conveniente que invierta entre 23½ y 50 horas en el entendimiento de los temas teóricos y prácticos de la asignatura. En un

cuatrimestre normal, esto equivale a entre 2 y 4¼ horas por semana desde la primera clase hasta el día del parcial.

Los resultados correspondientes al segundo caso estudiado se muestran en la Figura 6, donde se ha asignado el valor de 'mucho' a todos los nodos del cuestionario. Se puede decir que, esta es la situación en la que el alumno ha estudiado durante todo el cuatrimestre. Este tipo de alumno tiene un 97% de posibilidades de aprobar la evaluación parcial y con grandes posibilidades de conseguir una buena nota (al tener una probabilidad superior al 80% de tener más de 18 respuestas correctas). Pero, se debe recordar que esto es un modelo probabilístico, por lo que, a pesar de tener muy buenas posibilidades, de todas formas el alumno podría desaprobar. Para evitarlo, el modelo permite identificar los temas en que este tipo de alumno puede presentar un menor rendimiento (con mayor probabilidad en valores cercanos a 'algo' que 'mucho'). Por lo tanto, sería recomendado emplear entre 8½ y 14½ horas para reforzar el estudio de la teoría de Sistemas Inteligentes e Ingeniería del Conocimiento, así como la práctica de Métodos de Búsqueda.

6. Conclusiones

En el presente trabajo se presentan los resultados de la aplicación de un Modelo Bayesiano en el ámbito de la asignatura 'Inteligencia Artificial' en la UTN-FRBA. A partir de las predicciones generadas por el modelo, se identifican y analizan tres tipos de estilos de aprendizaje.

Del análisis realizado se destaca que, aunque los alumnos suelen aprobar la evaluación parcial, en general el entendimiento de los temas no es sólido. Sin embargo, para Lógica, Emparrillado y Análisis de Protocolos se observa un mayor rendimiento. Dado que el desarrollo de estos temas prácticos se encuentra distribuido en varias clases (y en una única clase como sucede con el resto), esto parecería motivar a los alumnos comiencen antes su estudio y, por lo tanto, a lograr un mejor entendimiento. Estas conclusiones permiten a los docentes determinar cambios en la planificación de las clases y la implementación de nuevos mecanismos pedagógicos para mejorar el nivel del aprendizaje y entendimiento de los contenidos.

Por otra parte, el Modelo Bayesiano será integrado con la aplicación IApp. A partir de la información generada por esta aplicación, el modelo podrá predecir si un alumno se encuentra en condiciones de aprobar, o no, el examen; y así recomendarle cómo debe organizar sus tiempos de estudio. Asimismo, el modelo generará información pertinente a los docentes sobre la situación en que se encuentra cada alumno, la cual será retroalimentada mediante actividades dentro y fuera del aula.

Referencias

1. Secretaría de Políticas Universitarias (2012) *Plan Estratégico de Formación de Ingenieros 2012-2016*. Ministerio de Educación de la Nación Argentina. http://tinyurl.com/kz9lrpz Disponible online en Junio de 2015.
2. Tornese, E. B., Algieri, R. D., Dogliotti, C. G., Gazzotti, A., Gómez, A., Jiménez, H. N., Mazzoglio & Nabar, M. J. & Rey, L. (2010) *Enseñanza audiovisual mediante TICs en Neurociencia*. 12 Jornadas Latinoamericanas de Neurociencia Cognitiva, Buenos Aires.

3. Seoane-Pardo, A. M. & García Peñalvo, F. J. (2006) *Criterios de calidad en formación continua basada en eLearning. Una propuesta metodológica de tutoría on-line*. http://tinyurl.com/mb39dg4 Disponible online en Junio de 2015.
4. Özyurt, Ö., & Özyurt, H. (2015) *Learning style based individualized adaptive e-learning environments: Content analysis of the articles published from 2005 to 2014*. Computers in Human Behavior, 52, 349-358.
5. Díaz-Mosquera, E. (2012) *Estilos de aprendizaje*. Revista EíDOS 5, Julio-Diciembre 2012, 5-11. ISSN: 1390-499X. http://tinyurl.com/pwx7hwg Disponible online en Junio de 2015.
6. Benítez, N. G., Cabrera, Y. F. & Ramos, J. F. C. (2014) *La integración de las Tecnologías de la Información y las Comunicaciones al control de indicadores de promoción en cohortes estudiantiles de la Educación Superior Cubana*. Revista Referencia Pedagógica, 2(1), 30-57.
7. UTN-FRBA (2008) *Programa la asignatura Inteligencia Artificial (Plan 2008)*. http://tinyurl.com/p6tks2j Disponible online en Junio de 2015.
8. Russell, S. J., Norvig, P., Davis, E., Russell, S. J. & Russell, S. J. (2010) *Artificial intelligence: a modern approac*h (Vol. 2). Englewood Cliffs: Prentice hall.
9. García-Martínez, R. & Britos, P. (2004) *Ingeniería de Sistemas Expertos*. Editorial Nueva Librería. ISBN 987-1104-15-4.
10. Acosta, M., Vegega, C., Deroche, A., Pytel, P. & Pollo-Cattaneo, Ma F. (2014*) La implementación de tecnología informática en el aprendizaje de Redes Neuronales: del estudio "de manual" a la experiencia "en el aula"*. Libro de Actas de IX Congreso de Tecnología en Educación y Educación en Tecnología. Págs. 47-56. Editorial UNdeC (Chilecito, Argentina). ISBN 978-987-24611-1-9.
11. Deroche, A., Pytel, P. & Pollo-Cattaneo, F. (2013a) *Propuesta de mejora en asignatura de grado mediante Explotación de Información*. Proceedings VIII Congreso de Tecnología en Educación y Educación en Tecnología. Artículo ID 5492. ISBN 978-987-1676-04-0.
12. Deroche, A., Raus, N. A., Lujan, F. N., Vegega, C., Pytel, P. & Pollo-Cattaneo, M. F. (2013b) *Caso de Estudio de la aplicación de Procesos de Explotación de Información en Educación*. Memorias del 1er Congreso Nacional de Ingeniería Informática / Sistemas de Información (CoNaIISI 2013). ID 35-427-1-DR. ISSN: 2346-9927.
13. Deroche, A., Acosta, M., Vegega, C., Bernal Tomadoni, L., Straccia, L., Pytel, P. & Pollo-Cattaneo, M. F. (2015) *Propuesta de Desarrollo de Aplicación Móvil para la Evaluación Dinámica en Asignatura de Grado en Ingeniería en Sistemas de Información.* Proceedings XVII Workshop de Investigadores en Ciencias de la Computación (Tecnología Informática Aplicada en Educación). Artículo 7074. ISBN 978-987-633-134-0.
14. Barber, D. (2012) *Bayesian Reasoning and Machine Learning*. The MIT Press.
15. García, P., Amandi, A., Schiaffino, S., & Campo, M. (2007) *Evaluating Bayesian networks' precision for detecting students' learning styles*. Computers & Education, 49(3), 794-808.
16. Schiaffino, S., Garcia, P., & Amandi, A. (2008) *eTeacher: Providing personalized assistance to e-learning students*. Computers & Education, 51(4), 1744-1754
17. Itoh, H., Itoh, Y., Funahashi, K., Yamamoto, D., Saito, S., Matsuo, H., & Takumi, I. (2014) *Forecasting students' future academic records using Bayesian network*. In Soft Computing and Intelligent Systems (SCIS), 2014 Joint 7th International Conference on and Advanced Intelligent Systems (ISIS), 15th International Symposium on, IEEE 458-462.
18. Millán, E., Jiménez, G., Belmonte, M. V. & Pérez-de-la-Cruz, J. L. (2015) *Learning Bayesian Networks for Student Modeling*. In Artificial Intelligence in Education. Springer International Publishing, 718-721.
19. López-Faican, L. G. & Chamba-Eras, L. A. (2015) *Redes bayesianas para predecir el estilo de aprendizaje de estudiantes en entornos virtuales*. AtoZ: novas práticas em informação e conhecimento, 3(2), 107-115.
20. Pytel, P., Vegega, C., Deroche, A. & Pollo-Cattaneo, M. F. (2015) *Implementación del Modelo Bayesiano para el Diagnóstico del Aprendizaje en alumnos de Inteligencia Artificial*. Reporte Técnico GEMIS-TD-2014-01-TR-2015-01. Grupo de Estudio de Metodologías para Ingeniería en Software, UTN-FRBA. http://tinyurl.com/q2qzelm Disponible online en Julio de 2015.

Herramienta de autor para la creación de actividades educativas basadas en Interacción Tangible

Artola Verónica[1,2], Sanz Cecilia[1], Moralejo Lucrecia[1], Pesado Patricia[1,3], Baldassarri Sandra[4]

[1] Instituto de Investigación en Informática LIDI, Facultad de Informática, Universidad Nacional de La Plata
[2] Becaria doctoral del Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
[3]CIC (Comisión de Investigaciones Científicas de la Pcia. de Buenos Aires), Argentina.
[4] AffectiveLab, GIGA, Universidad de Zaragoza. España

{vartola, csanz, lmoralejo, ppesado}@lidi.info.unlp.edu.ar
sandra@unizar.es

Resumen: En este artículo se presenta el diseño de una herramienta de autor para la creación de actividades educativas basadas en interacción tangible. En primer lugar se presentan antecedentes en relación a la interacción tangible y sus posibilidades para el escenario educativo. También se revisa el concepto de herramienta de autor y se enumera una serie de características deseables para aquellas herramientas de autor orientadas al contexto educativo. Finalmente, se describe TIES (*Tangible Interaction in Educational Scenario*) que es la herramienta de autor diseñada, foco de este trabajo. Se fundamenta la necesidad de contar con ella y se presentan algunas conclusiones y líneas de trabajo futuro.

Keywords: Herramienta de Autor, Interacción Tangible, Escenario educativo.

1 Introducción

En el año 1993, Wellner, Mackay y Gold afirmaban que: "*la mayor parte de nuestro trabajo con la computadora ocurre sentados delante de, y mirando, una sola pantalla que brilla intensamente unido a una serie de botones y un ratón. Nuestras diferentes tareas se asignan a ventanas superpuestas. Así desde el aislamiento de nuestras estaciones de trabajo tratamos de interactuar con nuestro entorno, pero los dos mundos tienen poco en común*" [1].

Ishii y Ullmer [2] también sostenían que vivimos divididos entre nuestro entorno físico y el ciberespacio. Enfatizaban sobre la división existente entre el mundo de los bits y de los átomos y la necesidad actual del ser humano de querer tener esa co-presencia estando todo el tiempo conectados. A partir de estas ideas y teniendo en cuenta las ideas de Weiser [3] sobre computación ubicua, Ishii propone aprovechar los *affordance* de los objetos de nuestro entorno para alcanzar una mejor interacción entre la gente y las tecnologías de la información. En su artículo "*Tangible Bits*" describe su propuesta de cómo disminuir la brecha entre el ambiente físico y el ciberespacio [2]. Así surge la noción de interacción tangible (IT), en la que a través de la manipulación física de las representaciones tangibles (objetos del entorno), las representaciones digitales se alteran. Las formas físicas, entonces, pueden servir al mismo tiempo como representación y control de sus contrapartes digitales. Hoy en día se habla de interfaces tangibles sustentadas en el entorno físico, con el empleo de objetos, superficies y espacios que encarnan información digital [4].

Existe un amplio abanico de aplicaciones basadas en interacción tangible para el escenario educativo y para el entretenimiento. Sin embargo, cada una de estas aplicaciones se diseña ad-hoc para cada necesidad específica. Han empezado a aparecer algunos trabajos que presentan *toolkits* para ayudar al desarrollador a abstraerse de las capas de más bajo nivel involucradas en las aplicaciones de IT y concentrarse en aquellas de más alto nivel [5][27][28]. Esto es importante porque las aplicaciones basadas en IT involucran la detección de objetos (ya sea a través de marcadores, a través del procesamiento de la imagen del objeto, o de RFDI – *Radio Frecuency Identification*), y en muchos casos, requieren de librerías para la captura e identificación de la imágenes de interés en el entorno físico en el que se trabaja. Al mismo tiempo, estas librerías deben "dialogar" con las librerías o módulos propios de la aplicación que se está desarrollando. Para el caso del diseño de actividades educativas basadas en interacción tangible, se ha visto la necesidad de los docentes de trabajar y participar en la creación de éstas [6]. Resulta impensable que los docentes se deban enfrentar al desarrollo de estas actividades educativas, sin contar con herramientas de autor para abordar la tarea. En este

trabajo se presentan los primeros pasos en el diseño de una herramienta de autor basada en plantillas, que permite al docente configurar sus propias actividades basadas en interacción tangible. De esta manera, se otorga una capa de abstracción que vuelve transparente para los docentes los aspectos de más bajo nivel involucrados en la IT. Se considera que la herramienta de autor es un aporte para el crecimiento de las posibilidades de IT en el escenario educativo.

El artículo se organiza en las siguientes secciones: en la sección 2, se hace una revisión de antecedentes que vinculan IT con el escenario educativo, también se describen algunos trabajos que proponen *toolkits* para el desarrollo de aplicaciones de IT; en la sección 3 se presenta el concepto de herramienta de autor y se describen los aspectos deseables para aquellas herramientas de autor orientadas al escenario educativo; en la sección 4, se presenta el aporte de este trabajo que es TIES una herramienta de autor orientada a la construcción de actividades educativas basadas en IT; en la sección 5 se describen aspectos técnicos en relación al desarrollo de TIES; en la sección 6, se explican algunas consideraciones que han aportado a la creación de la herramienta de autor; finalmente, en la sección 7 se delinean las primeras conclusiones en relación a TIES y las líneas de trabajo futuro.

2 Antecedentes

Los sistemas basados en interacción tangible han despertado la atención de los investigadores de diferentes disciplinas tales como la psicología cognitiva, la educación, y la tecnología educativa. Existe una creencia generalizada en relación a la manipulación de los objetos físicos y su vinculación con el aprendizaje y el desarrollo de las personas [7]. Por ejemplo, Montessori [8] creía que a través del juego con objetos físicos se permitía a los niños involucrase en una actividad con un propósito específico, considerando a estos objetos físicos como herramientas para su desarrollo. Los bloques de Dienes [9] constituyen otro ejemplo, en donde se presenta la manipulación de objetos físicos para la enseñanza de conceptos abstractos, tales como variable y valor. O´Malley y Fraser [7] presentan una interesante revisión de las posibilidades de la interacción tangible para el escenario educativo, y exponen un detallado análisis de las vinculaciones entre las representaciones físicas y las representaciones virtuales. En sus conclusiones explican que la importancia de las actividades físicas y multisensoriales, están siendo incrementalmente reconocidas, especialmente en los primeros años de desarrollo de las personas. Sin embargo, observan también que estas actividades se están realizando sin la utilización de las Tecnologías de la Información y la Comunicación, y que la utilización de actividades que integren la interacción con objetos físicos y digitales sería deseable.

Marshall et al. [10] afirman que el enfoque tangible se está volviendo una alternativa para las tradicionales interfaces gráficas de usuario. Estos autores pueden ver dos tipos de aprendizaje reflexivo derivado del trabajo con interfaces tangibles. Primero, el alumno puede aprender a usar las manipulaciones tangibles para alguna tarea. En este caso el aprendizaje es altamente situado, es decir propone cómo usar una determinada herramienta en un contexto particular. Segundo, el sistema con IT puede encapsular algún modelo teórico del mundo. En esta situación, el alumno puede explorar el modelo a través de la interacción con las manipulaciones tangibles. Así, según la visión de Marshall et al. [10], hay dos tipos de actividad que se pueden dar trabajando con manipulaciones tangibles: 1) una actividad expresiva donde lo tangible encarna de alguna manera el comportamiento del alumno (ya sea física o digitalmente) y 2) una actividad exploratoria donde lo tangible encarna un modelo que se le da al alumno para explorar y analizar, y que ha sido planificado por algún diseñador. El alumno explora ese modelo y lo contrapone con su modelo mental.

Por otra parte, el uso de superficies activas, tales como las *tabletops,* refuerzan algunas posibilidades para la integración de actividades basadas en interacción tangible en el escenario educativo [11] [12] [13]. En particular, los trabajos de [14] [15] ponen de manifiesto las posibilidades de las aplicaciones tangibles basadas en *tabletops* para el desarrollo de experiencias educativas colaborativas cara a cara.

El desarrollo de aplicaciones basadas en interacción tangible ya forma parte, desde hace años, del quehacer de los investigadores y docentes, y se ha ido incrementando en la actualidad. Existen algunas orientadas a acompañar el proceso de "*storytelling*" [4], otras orientadas a la enseñanza de programación [16] [17] [18], orientadas a las planificación y logística [19], a la enseñanza de temas vinculados con el arte [20], al escenario de educación especial y la comunicación [21], por señalar algunas.

Sin embargo, en todos los trabajos mencionados previamente, la creación de las aplicaciones no es trivial y se suele realizar ad-hoc. Se observa por tanto la necesidad de contar con herramientas que faciliten y

posibiliten que los docentes se involucren con el diseño de actividades basadas en IT, que puedan explicitar sus consignas, el *feedback*, las imágenes, y la secuencia entre actividades, por ejemplo.

Se han revisado algunos trabajos que proponen agregar capas con la intención de tener diferentes niveles de abstracción en el desarrollo de una aplicación tangible, transparentado así las cuestiones de implementación de más bajo nivel, que están relacionadas, por ejemplo, con la detección de los objetos físicos utilizados en el entorno real para interactuar con el entorno virtual [22]. Tal es el caso de *ToyVision* en el que se plantea una arquitectura en capas y se orienta a facilitar el desarrollo de juegos de tablero. *ToyVision* ofrece posibilidades para modelar el tipo de objeto físico con el que se trabajará en la aplicación y la forma en que éste será reconocido [5]. A partir de estos estudios y de la realización de experiencias de actividades de IT para el escenario educativo [6], se considera de importancia la implementación de herramientas de autor que se enfoquen en la creación de actividades educativas basadas en IT. Este trabajo focaliza en esta línea.

En la siguiente sección se describe el concepto de herramienta de autor, para luego presentar TIES una herramienta de autor especialmente diseñada para la creación de actividades educativas basadas en IT.

3 Herramientas de autor para el escenario educativo. Aspectos deseables

Las herramientas de autor son programas que permiten al usuario crear sus propios objetos multimediales sin necesidad de recurrir a un especialista informático. Estas herramientas fueron desarrolladas con la idea de que todas las personas contaran con la facilidad expresiva de plasmar sus ideas en formato digital. Por ello, habitualmente se trata de software con interfaces amigables, que cuentan con gran cantidad de ayudas y proponen el uso de plantillas predeterminadas para facilitar el manejo, por parte de aquellos que no saben programar. El trabajo con las herramientas de autor orientadas al escenario educativo puede enriquecer las propuestas de enseñanza y también ampliar las posibilidades de expresión y comunicación de los alumnos [23].

Para Gómez Villa et al. [24], "*las herramientas de autor son aquellas que permiten, mediante un proceso más o menos complejo de compilado, la generación de un programa que funciona independientemente del software que lo generó*", y en un sentido más específico, se puede incluir dentro de este grupo a "todas aquellas herramientas que permiten generar actividades, materiales y recursos en formato multimedia".

En el trabajo "Análisis comparativo de herramientas de autor para la creación de actividades de Realidad Aumentada" [25], se describen las herramientas de autor orientadas al ámbito educativo y se tienen en cuenta una serie de características comunes, que se vinculan con aspectos pedagógicos, tecnológicos y de diseño visual, tales como:

- Construir diversos tipos de actividades didácticas a partir de plantillas, específicamente diseñadas, con el fin de atender a diferentes objetivos didácticos.
- Incluir la posibilidad de incorporar enunciados y objetivos para las actividades a diseñar.
- Configurar una retroalimentación al alumno que puede ser textual, sonora y/o visual. Esto permitirá al alumno conocer más acerca de su desempeño en una cierta actividad y aprender a partir de su resolución.
- Armar secuencias de actividades de acuerdo a algún objetivo pedagógico.
- Ofrecer diferentes posibilidades de navegación entre las actividades.
- Personalizar aspectos de presentación y diseño visual de las actividades que se están creando.

En base a estas observaciones, se diseñó una herramienta de autor, llamada TIES para la creación de actividades educativas basadas en interacción tangible, que toma en consideración las características que son comunes a las herramientas de autor orientadas al ámbito educativo. Su diseño se plasmó a partir de trabajos previos de los autores en el área de IT y en el estudio de herramientas de autor para el escenario educativo.

4 TIES: *Tangible Interaction in Educational Scenario*

En este apartado se presenta el desarrollo de la primera versión de una herramienta de autor que permite la creación de actividades educativas basadas en interacción tangible. Además, esta herramienta cuenta con un componente que permite reproducir las actividades generadas por la herramienta de autor antes mencionada. Por lo tanto, se hará una distinción entre estos dos componentes que conforman TIES: *TIESViewer* y *TIESAuthor* (ver Fig. 1). *TIESViewer* es un módulo que permite al alumno/destinatario la ejecución de un

proyecto de IT configurado desde *TIESAuthor* y se ejecuta sobre una *tabletop*. En este trabajo se focaliza sobre el componente *TIESAuthor* que se describe a continuación, aunque para una mejor comprensión del lector se hará referencia a algunos aspectos de *TIESViewer*.

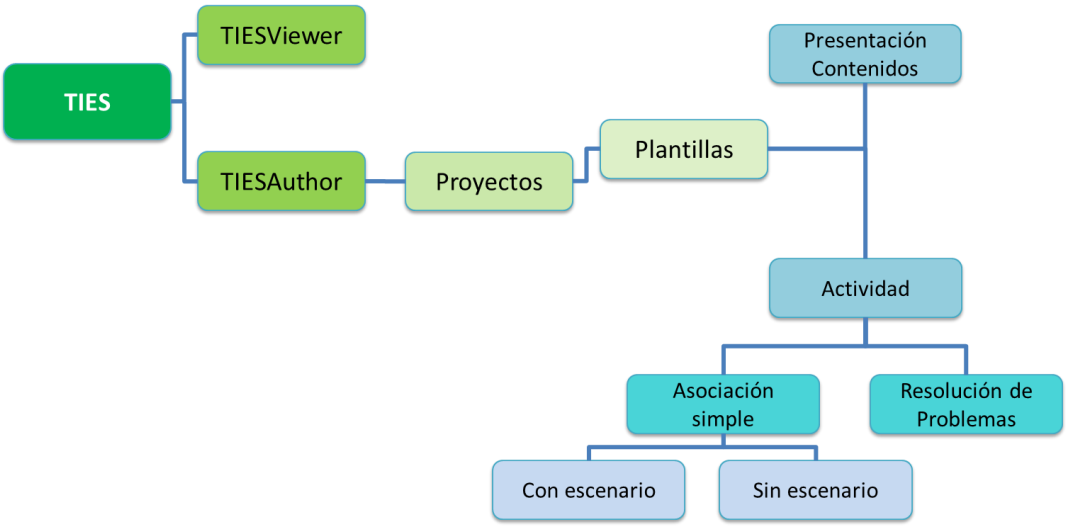


Fig. 1. Arquitectura de TIES

TIESAuthor

Este módulo es una aplicación web que permite la creación de proyectos de IT orientados al escenario educativo, para ser reproducidos luego con el visualizador *TIESViewer*. *TIESAuthor* presenta dos niveles de trabajo: el nivel de Proyecto IT que permite realizar un empaquetamiento de las actividades para conformar una secuencia con ellas y el nivel de creación de las actividades a partir de plantillas específicas (ver Fig. 1).

- **Nivel de Proyecto IT**

Las actividades creadas desde el módulo *TIESAuthor* se pueden empaquetar en un proyecto IT. Al crear un proyecto, el docente debe ingresar:

- Datos generales: nombre y temática del proyecto, nombre del autor y su dirección de correo electrónico y etiquetas para clasificarlo (éstas servirán para su búsqueda).
- Contenidos y Actividades: se deben agregar los contenidos y actividades que conforman el proyecto.
- Tipo de proyecto: se configura si se tratará de un proyecto para que los destinatarios compitan en las actividades, colaboren o trabajen de forma individual. También es posible elegir un proyecto mixto en este sentido. Esta elección limitará el tipo de actividades que podrán ser incluidas y la forma en que se visualizarán sobre la *tabletop*.
- Secuencia de Actividades: se puede definir el orden en que se presentará cada una de las actividades, lo que determina la secuencia de navegación para el destinatario, y es lo que *TIESViewer* utilizará para mostrar en un determinado orden las actividades.
- *Feedback* o retroalimentación de cierre del proyecto: se puede elegir entre un resumen de los puntajes obtenidos en cada una de las actividades y/o un *feedback* cualitativo que el docente haya establecido. Así cuando un alumno finalice la realización de todas las actividades con *TIESViewer*, se mostrará el *feedback* y el docente (si lo configuró de esta manera) puede recibir por correo electrónico dicho resumen.

En *TIESAuthor* se pueden distinguir dos tipos de contenidos, que se corresponden con diferentes tipos de plantillas dentro de la herramienta: la de presentación de información y las de actividades.

La plantilla de presentación de información posibilita incluir una introducción o cierre de cada actividad, detallar consignas, etc. Se pueden incluir videos, imágenes, sonidos y textos. Para la presentación de información se debe cargar el título y, luego, con un editor de texto enriquecido se completa la información deseada. Además, se puede incorporar un archivo de audio para que se ejecute mientras se realiza la lectura del texto con imágenes. En esta plantilla el docente tiene que configurar o bien el tiempo de duración de esta presentación, antes de avanzar al siguiente componte del paquete, o bien agregar un componente de acción para que el destinatario sea quien seleccione cuando avanzar.

- **Nivel de Actividades**

Las actividades se construyen a partir de plantillas previamente diseñadas. De momento se han creado dos tipos de plantillas: de resolución de problemas y de asociación simple (con o sin escenario). Cada plantilla de actividad permite presentar su objetivo y consigna y, en el caso de las actividades de asociación, además, se puede configurar un *feedback* para el alumno. El *feedback* se categoriza en "Error", "Acierto" o "Cierre". Es decir, el docente puede especificar una retroalimentación para cada uno de estas tres categorías.

Actividades de resolución de problemas

La plantilla de resolución de problemas, está disponible solo para proyectos colaborativos. Esta permite a los docentes generar actividades en las que los alumnos experimenten una dinámica colaborativa para resolver un problema.

Este tipo de actividad, tiene como objetivo que un grupo de alumnos arriben a la solución de un problema donde se plantean una serie de interrogantes. Los interrogantes deben poder ser respondidos a partir de la selección de un determinado objeto tangible entre los disponibles para la actividad.

Inicialmente, se ofrece información a cada alumno a modo de pista que se da a partir de la interacción con los objetos tangibles y la *tabletop*. Luego, se transita por una etapa de debate y análisis conjunto de la información recibida. La actividad termina cuando los participantes llegan a una conclusión del problema. El grupo gana si su conclusión concuerda con la real, y se le da una calificación de acuerdo a su desempeño. Si la conclusión no es la correcta, puede volver a intentarlo o ver la solución.

La actividad de resolución de problemas se encuentra dividida en etapas, cada una con un propósito y una duración diferentes:

- **Primera etapa:** se reparten un conjunto de objetos tangibles a cada participante. Estos, a través del objeto, pueden acceder a información única en relación al problema a resolver, que se llaman pistas acerca del problema. Por ejemplo, se apoya un objeto sobre la *tabletop* y se muestra información sobre el problema a resolver. De esta forma, el conocimiento de cada participante, y por lo tanto, su aporte en el trabajo grupal, es clave para el arribo a la solución del problema (ver Fig. 2).
- **Segunda etapa:** los jugadores se reúnen alrededor de la *tabletop* para compartir las pistas que cada uno recibió y analizó, y discuten la solución del problema planteado (etapa colaborativa propiamente dicha). Durante esta etapa el grupo puede solicitar información adicional a través de pistas extra (objetos tangibles adicionales) o consultando posibles vinculaciones entre las pistas ya entregadas (vinculan apoyando sobre la *tabletop* diferentes objetos) (ver Fig. 3).
- **Tercera etapa:** el grupo presenta su conclusión respondiendo a las preguntas planteadas en el problema. Se presenta la calificación obtenida y los participantes reflexionan sobre su desempeño y el resultado final de la actividad. El desempeño se vincula con el tiempo consumido en cada etapa y la cantidad de información adicional solicitada.

Para crear este tipo de actividad el docente configura los siguientes aspectos:

1. **Configuraciones generales:**
   a. Duración de las etapas: por cada etapa configura el tiempo de duración.
   b. Cantidad máxima de información a recordar por cada participante: se posibilita seleccionar frases acerca de la información que le brinda un determinado objeto al ser apoyado sobre la *tabletop*.
   c. Cantidad máxima de pistas extras a pedir: cuántos objetos extras puede solicitar el grupo, para que al interactuar sobre la *tabletop,* se obtenga información adicional acerca del problema.
   d. Cantidad máxima de relaciones a realizar: cuántas veces se puede vincular, apoyando sobre la *tabletop,* dos objetos para tener información sobre posibles relaciones entre ellos.

e. Decisión sobre permitir o no ver solución del caso, y de ser permitido, debe definir la imagen con la resolución del caso.
f. Decisión sobre permitir o no volver a intentar solucionar los interrogantes.

2. **Presentación del caso:** se puede ingresar un video con la consigna del problema a resolver.
3. **Cantidad de participantes:** se configura la cantidad de participantes. Por cada participante se debe establecer los objetos de interacción asignados. Además, por cada objeto se configura la información que aparecerá al apoyarlo sobre la *tabletop*: título, texto, imagen (opcional), y frases posibles de ser recordadas: se ingresan aquellas frases que podrán ser recordadas (seleccionadas) por un participante (ver Fig. 2).
4. **Pistas extras:** por cada pista extra, representada a través de un objeto tangible, se establece la información (texto y/o imagen) que aparecerá al colocarlo sobre la *tabletop* (ver Fig. 3).
5. **Relaciones entre objetos**. Por cada relación se configura (ver Fig. 3):
   a. Identificación de los dos objetos de la relación: se debe elegir dos objetos de los cargados previamente.
   b. Información (texto y/o imagen) a mostrar al apoyarse en la *tabletop* los objetos a relacionar.
6. **Interrogantes a resolver**. Por cada interrogante se detalla (ver Fig. 4):
   a. Texto del interrogante: cuál es la pregunta a resolver.
   b. Objeto que responde al interrogante: el docente debe elegir cuál de los objetos entre los cargados, responde al interrogante planteado.

De esta manera, el docente configura en TIES la actividad de resolución de problemas, considerando las consignas, *feedback*, y posibilidades de cada etapa.

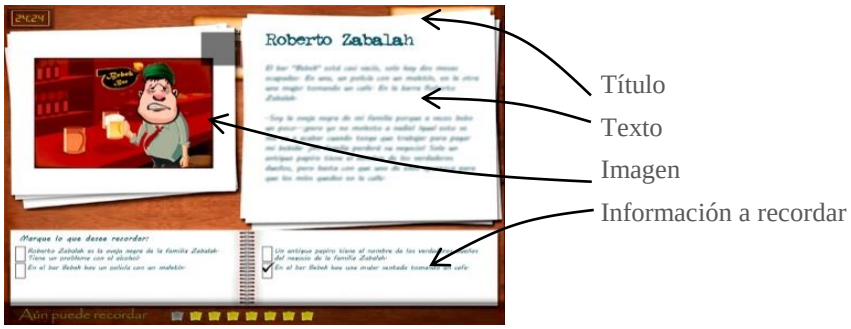


Fig. 2. Visualización de un objeto de interacción en *TIESViewer*

Fig. 3. Visualización en *TIESViewer* de información adicional. La imagen de la izquierda corresponde a la interacción con una pista extra. La imagen de la derecha corresponde a la interacción con dos objetos en busca de relaciones entre los mismos. En la parte superior de cada una, los recuadros grises indican la posición donde se encuentran apoyados los objetos. En ambas se visualiza la imagen y el texto, previamente configurados por el docente en *TIESAuthor*.

Fig. 4. Visualización de los interrogantes en *TIESViewer*

Actividades de asociación simple

Las actividades de asociación simple, pueden incluirse en proyectos colaborativos o individuales, y permiten a los alumnos asociar objetos tangibles a representaciones gráficas o imágenes en el plano.

Se diferencian dos tipos de asociaciones:

- **Simple sin escenario**

Aparece el gráfico o imagen a asociar sobre la superficie de la mesa, sobre un fondo de color, sin mostrar la relación de esa imagen con un escenario o contexto específico (ver Fig. 5).
Para crear una nueva actividad de asociación simple, el docente debe establecer:

- Nombre de la actividad.
- Color de fondo.
- Consigna
- Tareas. Las tareas corresponden a las asociaciones a realizar por los participantes. Por cada tarea el docente puede determinar:
  a. Una imagen.
  b. Tamaño y posición
  c. Nombre: El nombre será utilizado para relacionar el objeto tangible con el gráfico
  d. *Feedback* de tarea. Puede elegir sonido y/o imagen, en caso de respuesta correcta, y otro diferente para el caso de error o una imagen para cuando neutral en el caso que no conteste.

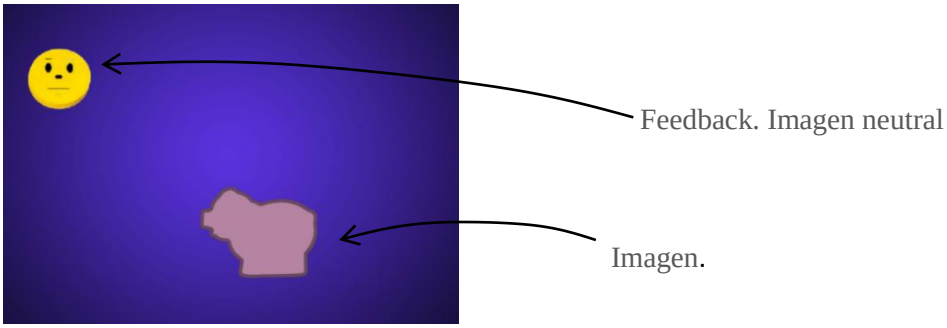


Fig. 5. Visualización de una actividad de asociación simple en TIESViewer

- **Simple con escenario**

La particularidad de este tipo de actividades es que las asociaciones de objetos tangibles a representaciones 2D están relacionadas con un escenario. Para este tipo de actividad el docente debe realizar la configuración como en el caso de la asociación simple sin escenario, pero además debe elegir una imagen de fondo que represente un escenario que tenga sentido para las tareas de asociación que se plantean. En la Figura 6, se

visualiza un ejemplo de aplicación de este tipo de actividad (ejecutada con *TIESViewer*) donde el alumno debe colocar en el carrito de compras, correspondiente a un escenario de un supermercado, los objetos tangibles que se asocian a las imágenes de productos que aparecen sobre la *tabletop*. En la Figura 7, la persona debe poner el objeto tomate real sobre la imagen del carrito. Es necesario que estos objetos posean marcadores que permitan a la aplicación identificarlos. *TIESAuthor* brinda la posibilidad de imprimir los marcadores que identifican a cada objeto involucrado en la actividad.

Fig. 6. Visualización de una actividad de asociación simple con escenario en *TIESViewer*.

Fig. 7. Ejemplo de aplicación de asociación simple con escenario.

5 Aspectos técnicos

TIESViewer fue desarrollado en el entorno *Adobe Flash Builder 4.5*[1] y *ActionScript 3.0*[2] *(AS3)*, con entorno de ejecución *Adobe AIR*[3]. La utilización de *Adobe AIR* surgió debido a las restricciones de seguridad de *Adobe Player* que imposibilitan el almacenamiento local de ficheros. *Adobe AIR* es un entorno de ejecución versátil que permite la creación de aplicaciones de escritorio (de propósito general), a partir de tecnologías de desarrollo de páginas web, como pueden ser *HTML*, *Ajax* o *Flash*.

TIESAuthor fue desarrollado en lenguaje *PHP*, utilizando *Laravel*, versión 4.2. El diseño de la herramienta se basó en el *Framework Bootstrap v3.3.2*.

[1] Adobe® Flash® Builder™ es un entorno de desarrollo integrado (IDE) para la construcción de aplicaciones enriquecidas de Internet (RIAs) multiplataforma. Está construido sobre Eclipse y ofrece todas las herramientas necesarias para desarrollar aplicaciones que utilicen el código abierto Flex y ActionScript 3.0.
Flash Builder se ejecuta en Microsoft Windows y Apple Mac OS X, y está disponible en varias versiones. http://www.adobe.com/

[2] ActionScript es el lenguaje de programación para los entornos de tiempo de ejecución de Adobe®Flash® Player y Adobe®AIR™. Entre otras muchas cosas, activa la interactividad y la gestión de datos en el contenido y las aplicaciones de Flash, Flex y AIR. Se ejecuta mediante la máquina virtual ActionScript, que forma parte de Flash Player y AIR. El código ActionScript suele transformarse en formato de código de bytes mediante el compilador. Entre los compiladores se incluyen el incorporado en Adobe® Flash® Builder™. El código de bytes está incorporado en los archivos SWF ejecutados por Flash Player y AIR. Obtenido de http://help.adobe.com

[3] Adobe AIR es multiplataforma y gratuito. http://www.adobe.com/

6 Algunas consideraciones

Los autores trabajan en la temática desde hace varios años, en particular en el desarrollo de diferentes aplicaciones de IT, sobre una *tabletop*, orientadas al escenario educativo. Una de estas aplicaciones es AcoTI (*Augmentative Communication through Tangible Interaction*), la cual es una herramienta educativa que se orienta a asistir al desarrollo de competencias comunicacionales de alumnos con Necesidades Complejas de Comunicación (NCC), quienes usualmente son usuarios de Comunicación Aumentativa y Alternativa (CAA). Es importante destacar que se ha llevado a cabo un trabajo de campo con alumnos de España y con docentes y profesionales de instituciones educativas de Argentina y España [6] [21]. Estos antecedentes sin duda resultaron relevantes para el desarrollo del trabajo presentado aquí. Las evaluaciones resultaron claves para considerar los siguientes aspectos en el diseño de TIES:

1. Posibilidad de configuración de las imágenes/gráficos a asociar con los objetos tangibles en una actividad.
2. Posibilidad de configurar diferentes tipos de *feedback* en las actividades (error, acierto, neutral), y que estos puedan ser visuales y/o sonoros.
3. Posibilidad para las actividades de asociación con escenario, de establecer el fondo que representa el escenario particular.
4. Formas diferentes de presentar la consigna (audio, video, imagen y/o texto).
5. Encadenamiento o secuenciación de actividades configurables
6. Realización de actividades en forma individual, colaborativa y/o en formato de juego de competencia.

Se considera que TIES ha involucrado de esta forma varias de las decisiones e ideas de los usuarios finales (docentes y alumnos) a los que está destinada la herramienta de autor, ya que estos han testeado aplicaciones que han desarrollado los autores, y han definido las necesidades de configuración y de diseño que se requerían, en cada caso. Al mismo tiempo, han confirmado la necesidad de contar con una herramienta de diseño de actividades de IT.

Sin embargo, se cree oportuno que se realicen pruebas de uso de la herramienta con estos usuarios para analizar aspectos de usabilidad, y esto será motivo de futuras publicaciones.

7 Conclusiones y Trabajos futuros

Las aplicaciones basadas en IT resultan una alternativa a las actuales interfaces gráficas de usuario. Se han estudiado diferentes autores que consideran que la interacción tangible constituye una oportunidad para el escenario educativo, en particular, para el desarrollo de actividades colaborativas cara a cara, soportadas en superficies interactivas tipo *tabletop*. También el uso de objetos físicos del entorno resultan, en determinados contextos y diseños educativos, una oportunidad para el aprendizaje.

Las actividades educativas basadas en IT requieren de diferentes capas de diseño y desarrollo, que van desde cuestiones de alto nivel, más vinculados al contexto de uso de la aplicación (en este caso el educativo), y por tanto, involucran al diseño instruccional, a cuestiones de más bajo nivel relacionadas con la forma de llevar adelante la detección de los objetos físicos y su mapeo con los objetos virtuales.

TIES permite crear actividades educativas para todos los niveles educativos y constituye de esta forma un aporte en este escenario de creación de actividades educativas basadas en IT, para que los docentes puedan participar en su diseño e implementación. Se cree que como principal resultado, esta herramienta abrirá el camino para la mayor vinculación de la interacción tangible con el escenario educativo.

Como trabajo futuro, se espera realizar evaluaciones de usabilidad de TIES, e integrarla con *AuthorAR*, que es una herramienta de autor para la creación de actividades educativas de Realidad Aumentada [26].

8 Bibliografía

1. Wellner P., Mackay W.& Gold R. (1993). Computer augmented environments: Back to the real world. Communication of the ACM, Vol. 36, No. 7, August1993.
2. Ishii H. & Ullmer B. (1997). Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (p. 241).
3. Weiser M. (1991).The Computer for the 21st Century. Scientific American, 1991, 265 (3), pp. 94-104.
4. Mazalek A., Davenport G. & Ishii H. (2002). Tangible Viewpoints: Physical Navigation through Interactive Stories.

5. Marco J., Cerezo E. & Baldassarri S. (2012). Modelado de controles tangibles para juegos con ToyVision. Actas del XIII Congreso Internacional de Interacción Persona Ordenador (Interacción 2012), 3-5 octubre 2012, Elche, España.
6. Sanz C., Guisen M. A., De Giusti A., Baldassarri S., Marco J. & Cerezo E. (2013). Games as Educational Strategy: A Case of Tangible Interaction for Users of Alternative and Augmentative Communication. *In International Conference on Collaboration Technologies and Systems. 377-381.* San Diego, California, USA.
7. O'Malley C & Fraser D.S. (2004). Literature Review in Learning with Tangible Technologies. NESTA Futurelab.
8. Montessori M. (1917). The Advanced Montessori Method. New York: Frederick A Stokes.
9. Dienes Z.P. (1964). *Building Up Mathematics* (2nd ed). London: Hutchinson International
10. Marshall P., Price S. & Rogers Y. (2003). Conceptualising tangibles to support learning. Conference on interaction Design and Children. IDC '03
11. Dillenbourg P. & Evans M. (2011). Interactive tabletops in education. International Journal of Computer - Supported Collaborative Learning 6(4), 491- 514.
12. Evans M.A., Rick J., Horn M.S., Shen C., Mercier E., McNaughton J., Higgins S., Tissenbaum M., Lui M. & Slotta J.D. (2012). Interactive Surfaces and Spaces: A Learning Sciences Agenda. *Symposium presented at the International Conference of the Learning Sciences*.
13. Kharrufa A, Leat D. & Olivier P. (2013). Designing for Reflection: A Case Study with Digital Tabletops and Digital Mysteries. *In:* Yang, H., Wang, S, ed. *Cases on E-Learning Management: Development and Implementation*. Hershey, Pennsylvania, USA: Information Science Reference, 2013, 268-292.
14. Kharrufa, A. S. (2010). Digital tabletops and collaborative learning. PhD Dissertation, Newcastle University. Recuperado de https://theses.ncl.ac.uk.
15. Martinez-Maldonado R., Kay J., & Yacef K. (2011). Visualizations for longitudinal participation, contribution and progress of a collaborative task at the tabletop. In Proceedings of the International Conference on Computer Supported Collaborative Learning 2011 (CSCL 2011), 25-32.
16. Suzuki H. & Kato H. (1995). Interaction-level support for collaborative learning: Algoblock–an open programming language. In Proc. CSCL '95, Lawrence Erlbaum.
17. Horn M. & Jacob R.J.K. (2006). Tangible Programming in the Classroom: A Practical Approach. Extended Abstracts CHI 2006, ACM Press.
18. Wang D., Qi Y., Zhang Y. & Wang T. (2013). TanPro-kit: a tangible programming tool for children. In Proceedings of the 12th International Conference on Interaction Design and Children, 344-347. ACM.
19. Zufferey G., Jermann P., Lucchi A. & Dillenbourg, P. (2009). TinkerSheets: Using Paper Forms to Control and Visualize Tangible Simulations. In Proceedings of the Third International Conference on Tangible and Embedded Interaction (TEI'09), 377-384.
20. Ryokai K., Marti S. & Ishii H. (2004). I/O Brush: Drawing with Everyday Objects as Ink. Proceedings of CHI 2004, 24-29, 2004, ACM.
21. Sanz C., Baldassarri S., Guisen M. A., Marco J., Cerezo E. & De Giusti A. (2012). ACoTI: herramienta de interacción tangible para el desarrollo de competencias comunicacionales en usuarios de comunicación alternativa. Primeros resultados de su evaluación. *En VII Congreso de Tecnología en Educación y Educación en Tecnología*. Pergamino, Buenos Aires, Argentina.
22. Shen C., Vernier F., Forlines, C. & Ringel, M. (2004). DiamondSpin: an extensible toolkit for around-the-table interaction. In Proc. CHI '04, 167-174, 2004.
23. Camarda P., Minzi V. (2012). "Primaria Digital, Aulas digitales móviles, Manual general introductorio". Primera edición. Buenos Aires. Ministerio de Educación de la Nación, 2012. ISBN 978-950-00-0949-2.
24. Gómez Villa M., Franco Morales, A., Martínez Valenzuela, J., Pastor Marín, P., Marín Saorín, S., Reyes Camacho Marín, S. y Villalba del Baño. J. (2002). "Herramientas de Autor e integración Curricular: 'Las Aventuras de Topy', una aplicación multimedia para el desarrollo de la comunicación alternativa y aumentativa en el aula". Actas del II Congreso Nacional de Nuevas Tecnologías y Necesidades Educativas Especiales, Murcia, España.
25. Moralejo, Sanz & Pesado (2014). "Análisis comparativo de herramientas de autor para la creación de actividades de Realidad Aumentada. Estudio de sus características específicas para el escenario educativo". . (http://hdl.handle.net/10915/43605) (24-07-2015).
26. Moralejo L., Sanz C., Pesado P., Baldassarri S. (2013). "AuthorAR: Authoring Tool for Building Educational Activities based on Augmented Reality". Conferencia. CTS 2013. Mayo de 2013, San Diego (California). ISBN 978-1-4673-6403-4.
27. Greenberg S. (2002) Enhancing creativity with groupware toolkits. Groupware: Design, Implementation, and Use. Springer. 2003. 1–9.
28. Hansen T.E., Hourcade J.P., Virbel M., Patali S. and Serra T. (2009). PyMT: a post-WIMP multi-touch user interface toolkit. International Conference on Interactive Tabletops and Surfaces (ITS '09). ACM, 17–24.

Contextualizando el Conocimiento de Usuarios desde sus Comentarios en Foros

Silvana Aciar y Gabriela Aciar[1]

[1]Universidad Nacional de San Juan, Argentina.
saciar@iinfo.unsj.edu.ar - gaby_aciar@yahoo.com.ar

Abstract. Este artículo propone una solución al problema de buscar usuarios que respondan preguntas sobre un tema. La solución propuesta es un sistema recomendador que sugiere usuarios en base al conocimiento sobre el tema. Este conocimiento es obtenido desde las publicaciones realizadas en los foros. El valor del conocimiento es diferente dependiendo del contexto en el que se utilice. En este trabajo se presenta una forma de contextualizar el conocimiento de una persona para luego utilizarlo en un sistema recomendador. Un caso de estudio en el dominio de educación, demuestra que se obtienen recomendaciones más precisas si tiene en cuenta información del contexto.

Keywords: Text Mining, Sistemas recomendadores, Foros, Interacciones de usuarios.

1. Introducción

Las tecnologías de la información y comunicación (TICs) han cambiado el modo de interacción entre las personas. La penetración masiva de Internet en los hogares hace posible que los usuarios tengan acceso a la gran cantidad de información disponible en la web. Esto provoca que en muchas ocasiones se produzca la paradoja de no saber qué escoger a pesar de que a nuestro alrededor se acumula información que podría ser de nuestro interés. Esta situación también se observa en el ámbito educativo. Hoy en día, la fuente de información más consultada es Internet. Actualmente las plataformas virtuales de aprendizaje presentan la misma información a los usuarios que entran en ella, información estándar con links a otras páginas [1]. Un estudiante o docente puede estar horas mirando las diferentes alternativas y al final desistir en su búsqueda por la cantidad de tiempo que le lleva visitar cada página, o quedarse con una opción que quizás no sea la mejor para su perfil. Es por eso que el problema ya no es de no tener información, si no, qué escoger entre un número inmanejable de alternativas disponibles.

Herramientas, con posibilidades sincrónicas y asincrónicas incorporadas a la formación y la educación, propician nuevas opciones de interacción y retroalimentación, para ayudar a los pedagogos y estudiantes a adquirir/enseñar conocimiento a más sectores de la sociedad [2] [3][4].

Una forma de facilitar la interacción de docentes/estudiantes con las plataformas virtuales de aprendizajes es mediante el uso de los sistemas recomendadores [5][6][7]. El propósito de estos sistemas es el de simplificar el proceso de búsqueda de una persona solucionando el problema de sobrecarga de información[8].

Los sistemas recomendadores en e-learning brindan sugerencias a los profesores/estudiantes durante su aprendizaje para maximizar su desempeño, proveyendo la recuperación y presentación personalizada de los recursos de aprendizaje en base a las necesidades, intereses, preferencias y gustos.

Este artículo se enfoca en el problema de recomendaciones de personas. Por ejemplo un estudiante que tiene una inquietud sobre un tema en particular y necesita que otro usuario que conoce del tema lo ayude. Situación que es frecuente en un ambiente no virtual. En una plataforma virtual saber quien conoce del tema es un poco más difícil debido a que puede haber muchos estudiantes o docentes registrados y no se cuenta con el contacto presencial que favorece tener conocimiento acerca de la experiencia y conocimiento que poseen las personas. En [9] se presentó la forma de obtener ese conocimiento que poseen los usuarios de las publicaciones en foros en una plataforma virtual de aprendizaje. En base a ese conocimiento son recomendadas a otras personas para responder sus inquietudes sobre un tema particular.

El presente trabajo es una extensión de [9], se integra en las recomendaciones información del contexto del conocimiento de la persona recomendada. Información del contexto es obtenida del análisis de las publicaciones realizadas en los foros. El conocimiento es sensible al contexto, la información que posee una persona tiene valores diferentes dependiendo del contexto en la cual se vaya a utilizar.

La estructura de este artículo es la siguiente: En la Sección 2 se presenta el trabajo previo realizado para obtener el grado de conocimiento de los usuarios desde los textos escritos en un foro. El proceso realizado para la obtención del contexto desde publicaciones en los foros es descrito en la Sección 3. Sección 4 presenta el proceso de realizar las recomendaciones de usuarios integrando información del contexto. En la sección 5 se presenta un caso de estudio que muestra la efectividad de la propuesta. Finalmente las conclusiones y trabajo futuro son presentados en la Sección 6.

2. Trabajo previo

En [9] se propone el uso de sistemas de recomendación para sugerir usuarios adecuados para responder preguntas de otras personas. En la literatura hay varios estudios que recomiendan a los usuarios interactuar en una red social [5] [6] [7] muy pocos analizan la información escrita y publicada por los usuarios en las redes sociales. Parte del problema radica en la complejidad de la extracción de información de texto y convertir esa información en recomendaciones. En [9] se analiza el contenido del comentario del usuario para obtener su conocimiento sobre un tema en particular, y luego se recomienda los usuarios en base a ese valor.

La figura 1 muestra el proceso utilizado para recomendar usuarios en base al conocimiento y disponibilidad.

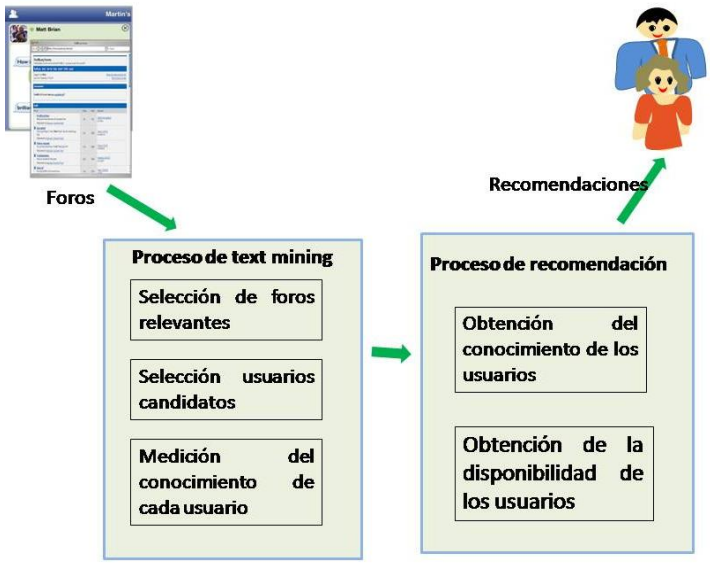


Fig. 1. Proceso para recomendar usuarios en base a las interacciones que realizan por medio de Foros

Se ha desarrollado un método de selección y recuperación de información desde los comentarios de los usuarios, métricas para obtener un valor que representa el conocimiento de los usuarios sobre un tema y se realizó la recomendación de un usuario en base a ese valor.

2.1 Selección de comentarios e identificación de los usuarios candidatos

Dada una petición de usuario *Q* donde *Q* está compuesta de palabras clave de modo que $Q = (q_1, q_2, ..., q_n)$, el primer paso es encontrar todos las publicaciones que contienen *Q*. La búsqueda de publicaciones relevante una búsqueda simple, se seleccionan todas las publicaciones que contienen las palabras de *Q* (q_i). El comentario seleccionado constituye una colección de publicaciones relevantes *Pr = {p1, p2, p3, .. pn}* donde cada publicación puede tener asociado un conjunto de comentarios y usuarios que las escribieron.

La Figura 2 muestra la estructura de los datos obtenidos a partir de las publicaciones pertinentes.

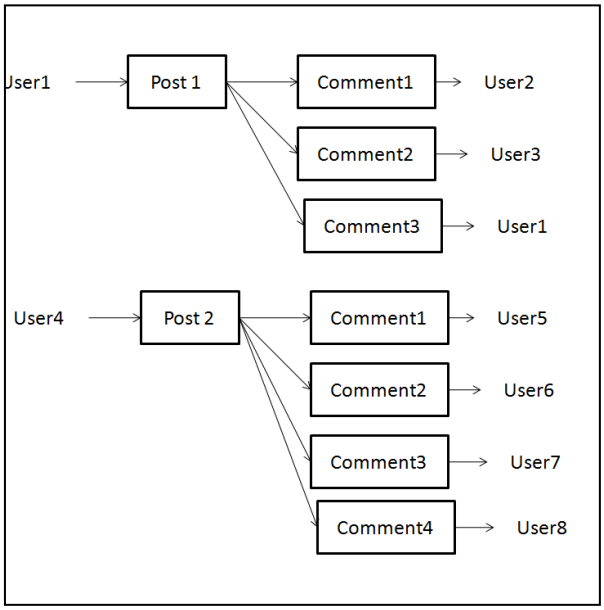


Fig. 2. Información obtenida de comentarios relevantes

Los usuarios que hicieron comentarios y los que hicieron la publicación constituyen un conjunto de usuarios candidatos *UC = {u1, u2, u3, ..., un}*. Para cada usuario candidato, se obtienen los comentarios realizados y las veces que interactuaron en la comunidad virtual.

2.2 Análisis de los comentarios del usuario

Un usuario puede responder a una pregunta *Q* si tiene conocimiento sobre *Q* y está disponible para responder preguntas. Se han definido un conjunto de métricas para obtener dicha información. La medida Información del Usuario sobre un Tema ***(UIT)*** se calcula con la información obtenida desde lo que han escrito en los comentarios, mientras que las medidas Interacción de Usuario ***(UI)*** y la Voluntad de Responder ***(WR)*** se calculan utilizando la información de las interacciones en la comunidad virtual. Más detalle del cálculo de estas medidas se encuentran en [9]

El Conocimiento del Usuario sobre el Tema ***(UKT)*** se obtiene en base a la información que tiene el usuario sobre el tema ***(UIT)*** multiplicado por el número de interacciones del usuario en la red virtual ***(UI)***. Ecuación 1 se utiliza para obtener el conocimiento de cada usuario candidato.

$$UKT(u) = UIT * IU \qquad (1)$$

El usuario más adecuado para recomendar es seleccionado en base al grado de conocimiento (***UKT)*** y la voluntad que tiene para responder una pregunta (***WR)***. ***WR*** es definida como la probabilidad que tiene un

usuario para responder una pregunta en base a su comportamiento en el pasado. Una persona que ha respondido frecuentemente preguntas de otros usuarios es más probable que esa persona responda otra pregunta respecto a otro usuario que no ha respondido preguntas en el pasado aunque se lo hayan preguntado. Más detalles del proceso obtención del conocimiento y recomendación de usuarios relevantes se encuentra en [9].

3. Obteniendo el contexto de los comentarios

El conocimiento que posee un usuario es sensible al contexto, por ejemplo: en el caso que un estudiante tenga dudas de un tema pide ayuda a un colega suyo que le pueda ayudar a resolver la duda, si la duda es respecto a un tema de ingeniería le pedirá ayuda a una persona que trabaja o se desempeña en esa área. En la solución planteada en la sección anterior no se tiene en cuenta el contexto en el cual un usuario tiene conocimiento sobre un tema .

En este trabajo el contexto se define como cualquier condición que permita identificar el área, la disciplina, las palabras claves relacionadas con el conocimiento que posee un usuario.

El contexto condicionará la aceptación o no de las recomendaciones realizadas a los usuarios. Figura 3 muestra el proceso de recomendación donde se añade información del contexto.

El contexto se modelará como un conjunto de conceptos involucrados en las publicaciones de cada usuario candidato. Estos conceptos serán identificados como aquellas palabras que fueron más mencionadas en las publicaciones. Se espera que el conjunto de palabras más mencionadas, presentadas a un usuario le permita seleccionar a la persona recomendada más adecuada para su contexto. En la Figura 4 se muestran las recomendaciones que se presentan al usuario con información del contexto en forma de palabras relacionadas.

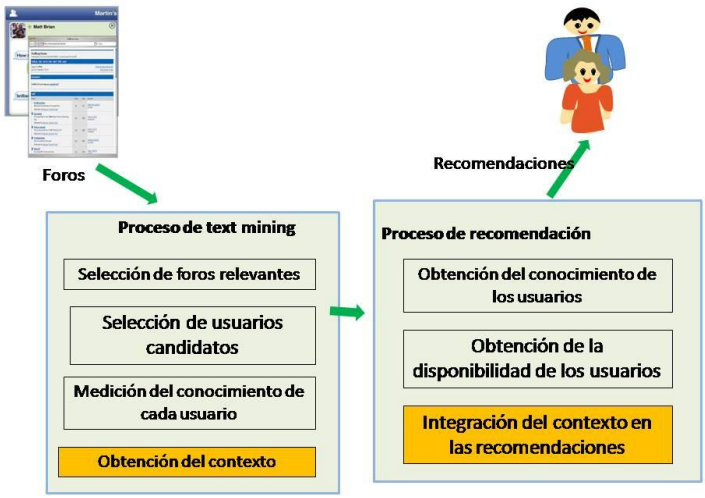


Fig. 3. Proceso para recomendar usuarios integrando información del contexto

Fig. 4. Recomendaciones resultantes con información del contexto en forma de palabras relacionadas

A continuación se definen los pasos a seguir en el proceso de obtención de información del contexto desde los comentarios escritos por los usuarios en un foro.

3.1 Preprocesamiento

Para cada usuario candidato se obtienen todas las publicaciones realizadas. Se eliminan todos aquellos términos que son irrelevantes, como lo son los artículos, las preposiciones, los adjetivos, etc. Para ello se utiliza un lista de palabras "stop words" elaborada para tal fin.

3.2 Selección de los términos relevantes

Las palabras claves que permitirán conocer el contexto en el cual un usuario tiene conocimiento sobre un tema se obtiene por medio del cálculo de las frecuencias de las palabras, las palabras más frecuentes son las que se presentarán junto con el nombre y contacto del usuario recomendado.

El contexto ***CX*** del conocimiento de un usuario ***i*** es un conjunto de duplas compuesta por las palabras más frecuentes en los textos escritos por ***i*** $\boldsymbol{CX_i}$ = ((p1, w1), (p2,w2),…(pn, wn)). Donde w_j es el peso que tiene la palabra j en todas las publicaciones del usuario i.
W_j se obtiene por medio de la medida *tf-idf* de la Ecuación 2 . F_{jk} es la frecuencia normalizada del término *j* en los comentarios realizados por el usuario *i*, los cuales les denominamos *k*. *idfj* representa la frecuencia invertida de la palabra *j*. Estas medidas se obtienen mediante las Ecuaciones 3 y 4.

$$w_j = f_{jk} * idf_j \qquad (2)$$

$$f_{jk} = \frac{frec_{jk}}{MaxFrec_k} \qquad (3)$$

$$idf_j = \log \frac{N}{n_j} \qquad (4)$$

Donde $frec_{jk}$ es la frecuencia de la palabra j en los comentarios k. $MaxFrec_k$ representa la máxima frecuencia sobre todos las palabras de los comentarios k. N es el número de fórums que existen y n_j es la cantidad de fórums que contienen el término j.

4. Proceso de recomendación

Los usuarios recomendados son aquellos que tienen el valor de conocimiento más alto y que de otro respecto al contexto. Para saber la diferencia entre los usuarios según el contexto, se modela el contexto del usuario como un vector. El vector está formado por las 5 palabras más frecuentes. Luego se realiza una comparación entre vectores formados por cadenas. Los tres usuarios menos similares respecto al contexto son recomendados, ordenados por grado de conocimiento.

5. Caso de estudio

Un caso de estudio en el dominio de e-learning fue realizado. Se crearon diferentes foros en la plataforma Moodle. En ellos participaron 12 estudiantes y 4 profesores. Se motivó a que participaran discutiendo varios temas en las áreas de Lenguaje, Matemática, Computación y Ciencias Sociales. Se obtuvieron 256 publicaciones de los usuarios.

5.1 Experimentos

La consigna del experimento fue que durante 15 días los estudiantes utilizaran el recomendador de usuarios aplicado en la plataforma Moodle para ingresar preguntas al recomendador y el mismo le sugerirá personas a las cuales contactar.

Cada vez que el usuario recibía una recomendación, luego de contactar a la persona recomendada se le pedía que evalué la recomendación. Recibía un email con el texto "La recomendación recibida ha satisfecho sus requerimientos?". Esta información se utilizó para evaluar las recomendaciones.

Se dividió a los estudiantes en dos grupos y se realizaron dos experimentos:

Experimento 1: 6 estudiantes utilizó el recomendador solamente teniendo en cuenta el conocimiento del usuario [9].

Experimento 2: Los restantes 6 estudiantes utilizaron el recomendador que realizó las sugerencias en base al conocimiento e incorporando información contextual de la persona recomendada.

5.2 Evaluación

La evaluación se realizó teniendo en cuenta las recomendaciones exitosas sobre todas las recomendaciones realizadas. Recomendaciones exitosas es definida como: dado un requerimiento del usuario, la recomendación que satisfaga sus expectativas es considerada exitosa, para ello se utiliza la información obtenida del email enviado para evaluar las recomendaciones. Si el usuario ha respondido afirmativamente que se ha satisfecho sus expectativas, la recomendación es considerada exitosa. Esta información es utilizada en la ecuación 5 para obtener un valor de precisión.

$$\text{Pr}ecisión = \frac{NR}{N} \qquad (5)$$

Donde NR es el número de recomendaciones exitosas y N el número total de recomendaciones realizadas. Esta medida se ha obtenido por cada día que se realizó el experimento (15 días) y para ambos experimentos. Los resultados se pueden observar en la figura 5. Como se puede observar las recomendaciones realizadas agregando información del contexto tienen mayor precisión. Los usuarios se encuentran más satisfechos si el recomendador les brinda más información referente a la persona a quien contactar.

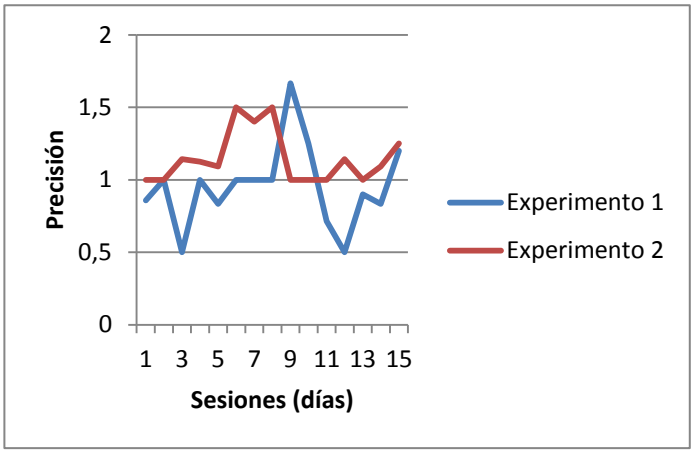


Fig. 5. Precisión de las recomendaciones obtenida en la ejecución de los dos experimentos

6. Conclusiones

En este trabajo se presentó una solución al problema de cómo obtener el contexto en el cual una persona posee conocimiento sobre un tema. El valor del conocimiento es diferente en diferentes contextos. Si la finalidad de un sistema recomendador es sugerir usuarios en base al conocimiento que tengan de un tema, es muy relevante tener en cuenta el contexto donde se generó o aplicó el conocimiento que posee la persona a recomendar. Una de las causas de la baja precisión de las recomendaciones puede ser debido a que no se tiene en cuenta el contexto en el cual se va a realizar la recomendación. La solución propuesta en este trabajo fue brindar información del contexto en las diferentes alternativas recomendadas, para ayudar al usuario a seleccionar una. El caso de estudio demuestra que al integrar dicha información se obtienen recomendaciones más precisas. La forma de obtener información del contexto, presentada en este trabajo, fue mediante el análisis de los textos escritos en las publicaciones de los usuarios en los foros.

Como trabajo futuro se prevé realizar dos mejoras: mejorar el método de obtención del contexto mediante la elaboración de mapas conceptuales con información de los foros; integrar la información contextual en el momento del cómputo de las recomendaciones y no dejar al usuario que realice la decisión final de qué alternativa recomendada seleccionar.

Referencias

1. Sekhavatian, M. Mahdavi (2011) APPLICATION OF RECOMMENDER SYSTEMS ON E-LEARNING ENVIRONMENTS, EDULEARN11 Proceedings, pp. 2679-2687.
2. Gaudioso, E.; Hernandez-del-Olmo, F.; Montero, M. "Enhancing E-Learning Through Teacher Support: Two Experiences", Education, IEEE Transactions on, On page(s): 109 - 115 Volume: 52, Issue: 1, Feb. 2009.
3. Abel, F.; Bittencourt, I.I.; Costa, E.; Henze, N.; Krause, D.; Vassileva, J. "Recommendations in Online Discussion Forums for E-Learning Systems", Learning Technologies, IEEE Transactions on, On page(s): 165 - 176 Volume: 3, Issue: 2, April-June 2010
4. Zhuhadar, L.; Romero, E.; Wyatt, R. "The Effectiveness of Personalization in Delivering E-learning Classes", Advances in Computer-Human Interactions, 2009. ACHI '09. Second International Conferences on, On page(s): 130 - 135
5. Salehi, M.; Kmalabadi, I.N. "Attribute-based recommender system for learning resource by learner preference tree", Computer and Knowledge Engineering (ICCKE), 2012 2nd International eConference on, On page(s): 133 - 138
6. Salehi, M.; Kmalabadi, I.N. "Attribute-based recommender system for learning resource by learner preference tree", Computer and Knowledge Engineering (ICCKE), 2012 2nd International eConference on, On page(s): 133 - 138
7. Khribi, M.K.; Jemni, M.; Nasraoui, O. "Automatic Recommendations for E-Learning Personalization Based on Web Usage Mining Techniques and Information Retrieval", Advanced Learning Technologies, 2008. ICALT '08. Eighth IEEE International Conference on, On page(s): 241 - 245
8. Primo, T.T.; Vicari, R.M.; Bernardi, K.S. "User profiles and Learning Objects as ontology individuals to allow reasoning and interoperability in recommender systems", Global Engineering Education Conference (EDUCON), 2012 IEEE, On page(s): 1 – 9.
9. Aciar, Silvana, and Gabriela Aciar. "Analyzing User's Comments to Peer Recommendations in Virtual Communities." Computational Intelligence in Data Mining-Volume 1. Springer India, ISBN 978-81-322-2204-0. 2015. 583-592.

Revisión de Microscopios Virtuales: en busca de características que favorezcan el trabajo colaborativo.

Lic. Martorelli Sabrina [1,4], Dr. Sergio R Martorelli [2], Dra. Cecilia Sanz [1], Lic. Patricia Pesado[1, 3]

[1]Instituto de Investigación en Informática LIDI. Facultad de Informática – UNLP,
[2]CEPAVE (CONICET-UNLP), La Plata, Buenos Aires, Argentina.
[3]Comisión de Investigación Científica de la Provincia de Buenos Aires (CIC), 526 e/10 y 11, La Plata, Buenos Aires, Argentina.
[4]Becaria de la Universidad Nacional de La Plata

smartorelli@lidi.info.unlp.edu.ar, sergio@cepave.edu.ar, csanz@lidi.info.unlp.edu.ar, ppesado@lidi.info.unlp.edu.ar

Abstract. Se presenta en este artículo un análisis de un conjunto de microscopios virtuales que se encuentran actualmente disponibles y que son utilizados en el ámbito educativo y académico. Se realiza una comparación de las diferentes características que los mismos ofrecen, con especial énfasis en aquellas características que se encuentren relacionadas con el trabajo colaborativo. Los microscopios virtuales han empezado a revolucionar la enseñanza y la investigación en disciplinas donde el uso de microscopios es esencial, la posibilidad de contar con características para la colaboración resulta un aspecto de interés y constituye un aporte en pos de acompañar procesos de trabajo y aprendizaje colaborativo. Para poder llevar adelante el análisis de los microscopios virtuales, se han definido criterios específicos los cuales han sido fundamentales también durante el proceso de recopilación bibliográfica. Este análisis constituye un aporte para los docentes e investigadores que trabajan cotidianamente con este tipo de herramientas. Se presentan los primeros resultados de este estudio, y las conclusiones a las que se arriban.

Keywords: Microscopios Virtuales, Análisis comparativo, Trabajo colaborativo

1 Introducción

En los últimos años, dentro de las disciplinas tradicionales en las que se utilizan microscopios como equipamiento elemental, se ha comenzado a trabajar cada vez más con aplicaciones llamadas microscopios virtuales (MV) que simulan, y hasta pueden complementar, el uso de microscopios convencionales. Estas aplicaciones permiten la visualización de imágenes digitales que han sido obtenidas a través de microscopios robotizados, *scanner* de preparados o cámaras digitales.

Los MV tienen varias ventajas sobre los microscopios convencionales, entre ellas: una diapositiva virtual puede ser consultada de forma remota; los especímenes no se degradan con el tiempo; y las diapositivas no se pueden romper ni perder e incluso se pueden copiar [1]. La mayoría de los autores que han escrito sobre las ventajas y desventajas de los MV, coinciden en la importancia de su utilización en la educación. Su uso permite preservar el material original, y al mismo tiempo, multiplicar los elementos disponibles en el aula para la realización de trabajos prácticos en cursos [2, 3, 4, 5, 6, 7].

Por otra parte, los ambiente colaborativos son sistemas representativos de uno de los siete paradigmas de Interacción Persona-Ordenador según la clasificación de Heim [8]. En estos ambientes se crea un espacio virtual con características específicas para que los usuarios puedan colaborar y trabajar de manera conjunta.

Por su parte, el aprendizaje colaborativo, busca aprovechar las ventajas de la interacción entre pares propiciando espacios en los cuales se dé el desarrollo de habilidades individuales y grupales. En el aprendizaje colaborativo se entrelazan elementos básicos como la interacción, la contribución individual, y las habilidades personales y de grupo. Así se orienta a compartir la autoridad, a aceptar la responsabilidad y el punto de vista del otro, y a construir consenso con los demás [9].

Resulta de interés explorar el escenario que integre las virtudes del trabajo y aprendizaje colaborativo con las ventajas aportadas por el uso de los MV.

El análisis de MV que este trabajo presenta, busca profundizar en las funcionalidades que los MV proporcionan con especial énfasis en aquellas que se relacionen con el trabajo colaborativo. Es un paso más en el camino de un proyecto que involucra además la generación de herramientas informáticas y estrategias didácticas complementarias para la enseñanza de la Patología Animal.

2 Aspectos considerados para el análisis de MV.

En este estudio se han considerado una serie de aspectos/criterios que se toman como referencia tanto para la revisión como para la posterior comparación de los MV estudiados. El primer aspecto se refiere a considerar en la búsqueda de microscopios aquellos que posean componentes que puedan facilitar el trabajo colaborativo. Se exploran entonces características y funcionalidades que puedan facilitar y promover la colaboración. El segundo aspecto que se ha tenido en cuenta y que ha servido como orientador de la búsqueda bibliográfica tiene que ver con el proveedor del MV. Existen dos grandes grupos de proveedores. Por un lado los relacionados con el ámbito educativo y académico, por lo general universidades alrededor del mundo que poseen carreras en las que se utilizan MV para sus prácticas habituales. Por otro lado, el grupo de empresas que ofrecen productos de microscopía, escáneres e cámaras fotográficas. Estos dos aspectos, junto al resto de los tenidos en cuenta se presentan en la tabla 1. Cada uno se detalla junto a una breve explicación y los valores prestablecidos como vocabulario para la caracterización, en los casos que corresponda.

Tabla 1. Aspectos considerados para el estudio de MV.

Aspecto	Descripción	Valores
Colaborativo	Aspectos de colaboración: ofrece herramientas para la comunicación, tipos de *awareness* disponibles, formas de coordinación	Tiene/ No Tiene – Se detallan descriptivamente cuáles
Proveedor	Proveedor del MV	Educativo-Académico/ privado
Tipo de Licencia	Licencia del sistema	Código Abierto / Propietaria, con o sin costo
Tipo de Aplicación	Ámbito de ejecución de la aplicación	Web / De escritorio /Móvil
Tipo de Uso	Uso libre o restringido	Libre / Restringido
Área o disciplina	A qué disciplina o área está dirigido el MV	Se describe si corresponde el área/disciplina particular
Formato de imagen	Formato de imágenes que soporta el MV para almacenar	Se detallan los tipos de formatos soportados
Tecnología/ Lenguaje	Tecnología subyacente al desarrollo	Se describe la tecnología subyacentes y los requerimientos

3 Recopilación y Análisis de Microscopios virtuales

Al momento de esta presentación se trabajó con una muestra de 40 microscopios virtuales, los cuales responden a la disponibilidad encontrada. De este total en este trabajo se detallan los 10 más relevantes, de manera tal de ejemplificar el tipo de análisis que se realiza. A continuación se presenta, en la tabla 2, los MV seleccionados.

Tabla 2. MV seleccionados para el análisis.

Descripción	Valores
Microscopio Virtual	Universidad de Loyola Chicago
Virtual Microscope 2.0	Universidad de Graz
Virtual Slidebox	Universidad de IOWA
Atlas Histológico Interactivo	Universidad de Jaen
UK Virtual Microscopes	The Open University/The Open Sience Laboratory y JISC
EPathViewer	Aperio/ Leica
iScan Image Viewer	Roche
NDP. VIEW2	Hamamatsu
Coolscope VS 'WebSlide	Bacus Laboratories
Microscopía virtual Google Earth/ Map API	Google

3.1 Microscopio Virtual de Virtual Histólogy

La Universidad Loyola de Chicago [10] ofrece en un sitio web una colección de imágenes microscópicas llamada *Virtual Histology* [11]. Dentro de esta colección las imágenes se encuentran organizadas en categorías. Para poder visualizar estas imágenes se utiliza un MV que ha sido desarrollado utilizando un software

panorámico o complemento web llamado *Zoomify* [12]. La combinación de estas aplicaciones funciona como un MV.
Zoomify permite realizar *zoom* y paneo sobre imágenes de alta calidad utilizando cualquier navegador, cualquier plataforma y cualquier dispositivo. El complemento incluye un visualizador que posee una barra de herramientas y un navegador que permiten moverse dentro de la imagen. *Zoomify* ofrece una licencia gratuita: *Zoomify HTML5 express*, y otras propietarias. La versión *Express* implementa: visualización, *zoom* y paneo, navegación utilizando mouse o teclado, modo de pantalla completa, función de ayuda y opción de almacenamiento *ZIF* de un archivo. Con esta versión no hay restricciones de la cantidad de implementaciones que se pueden realizar adquiriendo este tipo de licencia y se entrega un convertidor de imágenes, un visualizador, una guía de uso y ejemplos. La versión Pro ofrece todas características de la versión *Express* más el agregado de parámetros *HTML* y soporte *XML* que permiten incorporar las funcionalidades de *copyrighting*, marca de agua, mediciones, protectores de pantalla, rotaciones, *tours*, *slide shows*, *hotspots*, herramientas de *debug* y animaciones. Además con esta versión es posible realizar visualización de imágenes sin convertir y se puede acceder al código fuente del visualizador escrito en lenguaje *JavaScript*. Por último, la versión *Enterprise* ofrece todas las características de las otras más las funcionalidades que permiten aplicar filtros, realizar anotaciones, visualización de imágenes sincronizadas (*slidestack*) y la posibilidad de almacenar lo realizado en la imagen, por ejemplo, anotaciones. La Universidad Loyola de Chicago utiliza la versión *Express* para visualizar su colección y no presenta ninguna característica de colaboración entre las funcionalidades que ofrece.

3.2 Virtual Microscope 2.0

La Universidad Médica de Graz [13] comenzó a utilizar un MV llamado, *Virtual Microscope 1.0*, en el año 2002 dentro de un Entorno Virtual de Enseñanza y Aprendizaje propio. Luego de algunos años decidieron mejorar la implementación de esta primera versión puesto que consideran entre sus virtudes, un ahorro en el coste debido a que se logró la sustitución de cientos de microscopios físicos y un aporte de flexibilidad para sus estudiantes quienes pueden acceder desde cualquier navegador conectado a internet.
Para la nueva versión, *Virtual Microscope 2.0* [14], se ha utilizado el complemento web *Zoomify* [12]. Los creadores analizaron otros complementos, *VM of FHJoanneum* [15] y *WebMic* [16], pero finalmente optaron *Zoomify* pues cubre todos los requerimientos deseados. El MV desarrollado es una aplicación web estándar que no requiere instalación de software adicional para su normal funcionamiento. Entre las funcionalidades que provee se encuentran: mini mapa para visualización de imagen completa, y realizar *zoom* sobre la porción de la imagen que es seleccionada con un rectángulo dentro del mapa. Además es posible ajustar los diferentes marcadores disponibles (flecha, círculo, rectángulo, polígono) junto con un texto explicativo. Una novedad que este microscopio ofrece es la posibilidad de agregar un audio explicativo en lugar de texto. También dentro de las diapositivas virtuales pueden aparecer los llamados *Puntos de Interés*. Estos puntos han sido seleccionados

por docentes y son especialmente confeccionados para que puedan ser seguidos de cerca por los estudiantes.
No se ha encontrado ninguna característica que favorezca al trabajo colaborativo dentro de este MV.

3.3 The IOWA Virtual Slidebox

La Universidad de IOWA [17] ofrece un Atlas de Histología e Histopatología, llamado *The Iowa Virtual Slidebox* [18], de acceso abierto al cual se puede acceder utilizando el visualizador de imágenes *Biolucida* [19]. La colección posee más de 1000 preparados virtuales de alta resolución que pueden ser utilizados libremente.
Por su parte, *Biolucida* se presenta como una herramienta de la empresa *mbf BIOCIENCE* para la enseñanza de la Histología y Patología con diapositivas virtuales, que puede ser adquirida en tres ediciones: *Educación Médica, Investigación y Publicaciones*. *Biolucida* está compuesta por un servidor, un software del lado del servidor y un visor para el cliente o MV. Este último es una aplicación gratuita que sirve para distintos formatos de imágenes adquiridas con diversos escáneres y microscopios pero, de todas maneras, es necesario adquirir el sistema completo de *Biolucida* para cargar y organizar las diapositivas en el servidor. Entre sus funcionalidades se mencionan que posibilita el uso simultáneo de cientos de usuarios y que posee complementos para la integración con los principales entornos de enseñanza y aprendizaje disponibles. Además es compatible con imágenes generadas por otras compañías como *Zeiss*[20], *Olympus Aperio*[21] y *Hamamatsu*[22], entre otros. Los formatos de imágenes soportados son: *.jp2*, *.jpf*, *.jpx*, *.svs*, *.ndpi*, *.tif*, *.oib*. *oif*, *.tif*, y.*lsm*. Cada imagen puede ser recorrida en su totalidad y es posible realizar *zoom* a distintos aumentos. También se puede realizar capturas de las imágenes, o de porciones de las mismas y realizar ajustes de colores, brillo, contraste y gama. Existe la posibilidad de realizar anotaciones y agregar comentarios y enlaces en las mismas.
El sistema posee una sección de *bookmarks* donde se pueden encontrar las imágenes más utilizadas. Las imágenes pueden ser organizadas en grupos y es posible agregar *tags* a las mismas, lo que facilita su búsqueda y recuperación posterior.
Un aspecto interesante es que se pueden realizar análisis automatizados sobre las imágenes. El sistema se integra con otros sistemas de análisis ofrecidos por la compañía, llamados *Neurolucida* y *Stereo Investigator*, los cuales permiten rastrear neuronas o realizar estereología. El sistema ofrece versiones tanto para plataformas PC como Mac. En relación a las características de colaboración, este sistema permite compartir imágenes y resultados de análisis entre usuarios del sistema.

3.4 Atlas Histológico Interactivo

Dentro del Departamento de Biología Experimental de la Universidad de Jaén [23] han desarrollado un sitio web para los estudiantes de histología y organografía animal y vegetal el cual pretende aportar al aprendizaje del contenido práctico de estas disciplinas y servir como complemento a las observaciones realizadas en microscopios convencionales [24]. El sitio presenta un menú en donde se encuentran

las imágenes organizadas en categorías: Órganos animales, Órganos vegetales y Microscopía Virtual. Dentro de las dos primeras categorías, se encuentran las imágenes descriptas detalladamente, y sobre las cuales se puede realizar un *zoom* manual comenzando desde el aumento menor. Sobre algunas zonas de las imágenes aparecen rectángulos de colores a los cuales se les puede hacer clic para acceder a esa porción aumentada. Además para alguna de estas imágenes es posible simular la visualización en un microscopio de luz polarizada.
Resultó interesante el análisis de lo ofrecido por esta Universidad ya que incluye en estas primeras categorías un MV más "artesanal". Sin embargo en la categoría Microscopía Virtual se encuentran unas series de imágenes en las cuales se utiliza el complemento *Zoomify* en su versión *Express* para ser recorridas y constituyen otra forma de simular un microscopio.
Esta aplicación no ofrece características para el trabajo colaborativo.

3.5 UK Virtual Microscopes

El Proyecto *Virtual Microscope for Earth Sciences Project* [25] financiado por *The Open University* [26], *The Open Sience Laboratory* [27] y JISC [28], tiene como objetivo realizar un cambio en la enseñanza de las Ciencias de la Tierra, proporcionando el acceso a la colecciones de rocas que se encuentran actualmente en museos, universidades y otras instituciones de todo el mundo. Para esto se utiliza un MV con características de polarización, que permite a los usuarios examinar y explorar los minerales y las características microscópicas de rocas que pueden ayudar a desarrollar habilidades de clasificación e identificación.
El proyecto presenta diversas colecciones como la colección *UKVM* que consta de más de 100 rocas procedentes del Reino Unido e Irlanda, la colección de rocas recogidas por Charles Darwin durante su viaje en el Beagle, la colección de meteoritos que llegaron desde el espacio exterior, o la de rocas lunares recogidas por los astronautas de la *NASA*.
Al seleccionar una muestra, el microscopio se abre en una nueva ventana y se puede realizar *zoom* sobre la imagen. Cada muestra se puede ver bajo diferentes condiciones de luz polarizada (*PPL o XPL*). Un aspecto importante de la identificación de minerales es la capacidad de rotar la muestra, por lo que esta es otra de las funcionalidades ofrecidas por el microscopio.
Existe una escala milimétrica que es posible mover sobre la pantalla para realizar mediciones dentro de la imagen. Además, se pueden determinar dimensiones utilizando la herramienta de medida.
El microscopio proporciona una funcionalidad que permite compartir la imagen que se está viendo junto con las condiciones de visualización y nivel de *zoom* que han sido aplicadas. Esto se realiza mediante la copia de la URL de la imagen con ciertos parámetros de configuración que indican las condiciones aplicadas, y es la única posibilidad de colaboración que se tiene.

3.6 ePathViewer

ePathViewer [29] es una de las aplicación ofrecidas por *Aperio Digital Pathology Solutions* [30]. Fue diseñada exclusivamente para los dispositivos móviles *Apple iPad y iPhone (IOS)* y permite realizar un paneo rápido y *zoom* de diapositivas. Esta aplicación posibilita trabajar con magnificaciones entre 1x y el máximo aumento de la imagen original escaneada. Para cada diapositiva es posible ajustar metadatos, contraste, brillo y balance de color. *ePathViewer* permite realizar capturas de imágenes y enviarlas por correo electrónico. También posee una regla para realizar mediciones sobre las imágenes. La aplicación está disponible para su descarga en la tienda iTunes [31] de Apple de forma gratuita. Una funcionalidad innovadora que ofrece esta aplicación es que realizando la compra de un agregado de la aplicación (*In-Application*) es posible acceder a colecciones de diapositivas virtuales alojadas en cualquier URL, y no solo a las que se encuentran en las colecciones disponibles por defecto. Este MV no posee características de colaboración definidas.

3.7 iScanImageViewer

IScan Image Viewer [32] es un software de visualización de imágenes ofrecido por la empresa Roche [33] que permite ver preparados escaneados con *Ventana iScan Coreo Au* [34]. Este microscopio es una aplicación para *Windows* que permite abrir imágenes almacenadas localmente o en forma remota.
El microscopio cuenta la posibilidad de realizar anotaciones, medición y conteo celular. Además es posible visualizar más de una imagen simultáneamente, trabajar con magnificaciones entre 1x a 80x, rotar imágenes en 90˚ y 180˚, y ver imágenes escaneadas con volumen, en múltiples planos. Los formatos de imágenes reconocidos por el microscopio son *tif*, *jp2*, y *bif*. Este microscopio posibilita, además, la incorporación de dos complementos llamados *pathologists* e *histotechnologist*.
La única funcionalidad que podría relacionarse con la colaboración dentro de este MV es la que posibilita exportar imágenes y áreas.

3.8 NDP. VIEW

NDP. VIEW2 [35] es un software de visualización de imágenes digitales creado por la empresa *HAMAMATSU* [22] y desarrollado para preparados obtenidos con escáneres de la series *NanoZooner* [36]. Entre las operaciones disponibles se encuentran la posibilidad de realizar *zoom* y rotar imágenes en 90° o más. También es posible realizar correcciones de color tales como *gamma*, valores, brillo y contraste, y mejorar secciones con mala definición dentro de la imagen, realizando un ajuste de color. Se permite insertar flechas, dibujos, comentarios, medidas de longitud y superficie. Existe un menú de diapositivas en miniatura, y así se permite la visualización simultánea de varias imágenes y el uso de solapas para trabajar con más de una imagen a la vez. Cualquier área dentro de la imagen puede ser exportada y es éste nuevamente el único rasgo de colaboración que podría mencionarse.

3.9 Coolscope VS 'WebSlide

Coolscope VS 'WebSlide [37] es un MV ofrecido por *Bacus Laboratories* [38] y es parte de una suite llamada *COOLSCOPE VS Software Suite*, la cual permite producir, archivar y compartir las imágenes a través de intranet o Internet.
El MV permite las exploraciones de especímenes completos en 40x. Todas las herramientas que posee el microscopio y todos los ajustes que pueden realizarse sobre las imágenes, pueden ser realizadas con el ratón: apertura, ajuste de brillo, movimiento escénico, enfoque, y ampliación. Existe la posibilidad de almacenar las imágenes con las que se trabaja dentro de un servidor que es otro de los componentes de la suite (*WebSlide VS Server LE®).* Esto es ofrecido para los casos en los que se requiera visualizaciones simultáneas de los preparados virtuales.
En relación a la colaboración, y haciendo uso de otra de las herramientas de la suite, es posible compartir diapositivas virtuales a través de intranet o Internet. Se permite el control remoto de las diapositivas a través de un navegador web estándar y es posible guardar imágenes localmente.

3.10 Microscopios virtuales con Google Earth/ Map API

Una solución que ha surgido en cuanto al desarrollo de MV tiene que ver con el uso de *Google Earth API* [39] o *Google Map API* [40]. Estas APIs son orientadas a sistemas de información geográfica y trabajan en su esencia de una manera similar a la microscopía virtual realizando *zoom* y paneo de imágenes y movimientos a lo largo de archivos de gran tamaño [41]. Muchos autores coinciden que las soluciones comerciales para la microscopía virtual tienden a ser rígidas y difíciles de personalizar, probablemente para proteger los desarrollos, lo que dificulta el uso para el usuario final.
Para que una fotografía sea vista por el API de *Google Maps*, se debe segmentar en pequeñas imágenes que conforman un mosaico para cada nivel de visualización. Esta segmentación se logra utilizando softwares como *MapTiler* [42].
En este caso las funcionalidades que el MV posea serán las mismas que la API ofrezca y deberán ser adaptadas de la mejor manera posible para ser útiles para el microscopio. Entre las ventajas de trabajar con estas APIs se puede mencionar la frecuente actualización de las mismas, lo cual introduce mejoras constantemente y el uso libre para propósitos académicos. Si bien no se trata de un MV, es posible utilizar estas APIs para lograr la funcionalidad de un MV. Este es un camino a explorar.

4 Principales resultados del análisis realizado

A modo de resumen de los MV analizados se presenta la siguiente tabla comparativa.

	Colaboración	Proveedor	Tipo de Licencia	Tipo de Aplicación	Tipo de Uso	Área o Disciplina	Formato de Imágen	Tecnología / Lenguaje
3.1	No Tiene	Educativo Académico	Catalogo libre	Web	Libre	Medicina	Sin datos	Web + Zoomify
3.2	No Tiene	Educativo Académico	Sin datos	Web	Restringido a LMS	Medicina	Sin datos	Web + Zoomify
3.3	Tiene. Admite compartir imágenes con modificaciones hechas por otros usuarios	Educativo Académico	Libre el acceso al Atlas	Web	Libre	Medicina	Sin datos	Web + Biolucida
3.4	No Tiene	Educativo Académico	Sin datos	Web	Libre	Histología y Organografía	Sin datos	Web + Zoomify
3.5	Tiene. Compartir imágenes con los aspectos	Educativo Académico y Privado	Abierta	Web	Libre	Ciencias de la Tierra	Sin datos	Sin datos
3.6	No Tiene	Privado	Propietaria	Móvil	Libre	Multiples disciplinas	Sin datos	App para IOS
3.7	Tiene. Permite compartir imágenes completas y áreas de las mismas.	Privado	Propietaria	De escritorio	Restringido	Multiples disciplinas	TIF, JP2, y BIF	Sin datos
3.8	Tiene. Es Posible exportar imágenes para otros usuarios	Privado	Propietaria	Web	Restringido	Multiples disciplinas	Formato propietario	Sin datos
3.9	Tiene. Permite compartir diapositivas, y guardarlas localmente	Privado	Propietaria	Web	Restringido	Multiples disciplinas	Formato propietario	Sin datos
3.10	Depende de las posibilidades dadas usando las APIs	Privado	Código abierto	No específico	No específico	No específico	No específico	API

Uno de los objetivos principales de este análisis fue detectar características y funcionalidades que favorezcan el trabajo colaborativo. Se ha observado que de los microscopios analizados son muy pocos los que incorporan algún rasgo mínimo de colaboración. La única funcionalidad que ofrecen es la posibilidad de trabajar sobre ciertas imágenes realizando ajustes y anotaciones y luego compartir lo realizado. En este sentido son los MV virtuales pertenecientes al grupo de las empresas los que ofrecen esta característica, a la que por lo general denominan *Share* (compartir).

En cambio, para algunos de los microscopios ofrecidos por las universidades, la única funcionalidad que podría asociarse de manera indirecta con la colaboración es la posibilidad de obtener capturas de pantallas las cuales podrían ser compartidas posteriormente. De todas maneras, cualquier actividad colaborativa que pueda surgir a partir de estas capturas se resolvería fuera de la aplicación del microscopio.

Otra de las cuestiones observadas tiene que ver con las funcionales básicas ofrecidas por los MV. La mayoría de los analizados permiten recorrer una imagen, hacer *zoom*, realizar ajustes de color, brillo, contraste, y realizar mediciones. Además, y a excepción de *ePathViewer*, todos los microscopios permiten realizar anotaciones. Se destaca a *Virtual Microscope 2.0*, el cual permite además adjuntar audios a las anotaciones creadas.

Por otra parte, todas las empresas proveedoras de dispositivos y servicios relacionados con la microscopia virtual poseen algún software de visualización de imágenes, que por lo general puede ser descargado de manera gratuita, pero sólo funciona con las imágenes obtenidas de los productos que la misma empresa ofrece.

De todas maneras, muchas veces estas empresas ofrecen colecciones de imágenes de acceso gratuito que pueden ser consultadas y utilizadas haciendo uso del MV, pero si se desea trabajar con imágenes propias es necesario adquirir los servicios de almacenamiento o escaneo. Se observa además que estos MV ofrecidos por empresas son los que poseen mejores funcionalidades y actualizaciones. Esto, en contraste con los ofrecidos por universidades, de los cuales es posible encontrar solo algunas versiones, no siempre completamente actualizadas, y que por lo general, responden a necesidades específicas de la misma universidad.
Otra cuestión que se ha observado tiene que ver con que el tipo de disciplina que utiliza el microscopio puede determinar ciertas características o funcionalidades distintivas del mismo. Por ejemplo, se ha presentado el caso del *UK Virtual Microscopes*, exclusivamente usado con muestras de minerales y rocas, y por ello posee funcionalidades que permiten trabajar con luz polarizada, y rotar las muestras. Estas funcionalidades no se observan en otros microscopios utilizados por otras disciplinas como la medicina o la histología. Por su parte, *ePathViewer* es un buen ejemplo de cómo todas las aplicaciones tienden a adaptarse a los nuevos dispositivos disponibles en el mercado, en este caso móviles. Vale señalar que estas adaptaciones algunas veces pierden funcionalidades, como es el caso de realizar anotaciones, debido a las limitaciones de los dispositivos. Por ultimo *Zoomify* aparece como un ejemplo de los varios complementos web disponibles que permiten realizar *zoom* y paneo sobre imágenes de alta calidad, y que cada vez están siendo más utilizados para la construcción de MV.

5 Conclusiones y Trabajos Futuros

En este trabajo se ha presentado como aporte una revisión de software de microscopios virtuales y un análisis posterior, donde se han visto sus características y se los ha comparado. El estudio se cree de interés para el ámbito de las disciplinas que hacen uso de este tipo de instrumentos como parte de sus procesos educativos y de investigación. En particular, en el ámbito de la Parasitología, los microscopios virtuales se han vuelto una herramienta complementaria interesante para el trabajo con los alumnos, esto ha sido abordado en trabajos previos de los autores.

Uno de los criterios tomados para la búsqueda de herramientas de microscopía virtual se vinculó con las características de colaboración, sin embargo, no se han encontrado resultados interesantes en este sentido, lo que abre las puertas a comenzar a abordar la incorporación de este tipo de posibilidades para complementar las ya existentes. Esto se vuelve importante para que los alumnos puedan realizar análisis conjuntos en los MV, a partir de la identificación de elementos de interés en los preparados virtuales observados, por ejemplo. Se abre así un camino de mejoras para los actuales microscopios virtuales, con nuevas oportunidades para el escenario educativo y académico.

Referencias

1. Appleton, B., Bradley, A. P. & Wildermoth M. (2005). Towards Optimal Image Stitching for Virtual Microscopy. Proceedings: DICTA 2005
2. Treanor, D., Waterhouse, M. , Lewis F., & Quirke, P. (2007). A virtual slide library for histopathology. J Pathol. 213 (S1): S33
3. Treanor, D. (2009) Virtual slides: an introduction. Diagnostic Histopathology 15:2 99-103
4. Treanor, D. & Quirke, P. (2007). The virtual slide and conventional microscope - a direct comparison of their diagnostic efficiency. J Pathol. 213(S1):7A
5. Garcia Rojo, M., Bueno García, G., González García, J. & Carbajo Vicente, M. (2005). Preparaciones digitales en los servicios de Anatomía Patológica (I).Aspectos básicos de imagen digital. Revista Española de Patología. 38(2): 69-77
6. Conde, M. AF. (2006).Microscopia virtual: ¿Un cambio en la forma de hacer telepatología?
7. Martorelli Sabrina L, Sanz Cecilia V., Giacomantone Javier, Martorelli Sergio R. (2012). ParasitePics: un prototipo de repositorio de imágenes de Parasitología Animal
8. Heim, S. (2007). The Resonant Interface: HCI Foundations for Interaction Design.
9. Zañartu Correa, L. M. (2003). Aprendizaje colaborativo: una nueva forma de Diálogo Interpersonal y en Red. Revista Digital De Educación Y Nuevas Tecnologías
11. Virtual Histology, http://www.meddean.luc.edu/lumen/MedEd/Histo/virtualhistology.htm
12. Zoomify, http://www.zoomify.com/
14. Rehatschek, H.The introduction of a new virtual microscope into the eLearning platform o the Medical University of Graz. Med. Univ. Graz, Graz, Austria ; Hye, F. Interactive Collaborative Learning (ICL), 2011 14th International Conference on
15. FH-Joanneum, homepage. [URL (4 July 2011): http://www.fhjoanneum.
16. WebMic, http://artsandsciences.sc.edu/technology/webmic
17. Universidad de IOWA, http://www.uiowa.edu/
18. The Iowa Virtual Slidebox, http://www.path.uiowa.edu/virtualslidebox/
19. Biolucida, http://www.mbfbioscience.com/biolucida
20. Zeiss, http://www.zeiss.com/
21. Olympus Aperio www.leicabiosystems.com/pathology-imaging/aperio-digital- pathology/
22. Hamamatsu, http://www.hamamatsu.com/
23. Universidad de Jaén, http://www10.ujaen.es/
24. Atlas Histológico Interactivo, http://www.ujaen.es/investiga/atlas/
25. Virtual Microscope for Earth Sciences Project, http://www.virtualmicroscope.org/
26. The Open University , http://www.open.ac.uk/
27. The Open Sience Laboratory http://www.open.ac.uk/researchprojects/open-science/
28. JISC , https://www.jisc.ac.uk/
29. ePathViewer http://www.leicabiosystems.com
30. Aperio Digital Pathology Solutions. http://www.leicabiosystems.com
31. iTunes http://www.apple.com/itunes/
32. IScan Image Viewer, http://www.ventana.com/product/page?view=iscan
33. Roche, http://www.roche.com/index.htm
34. Ventana iScan Coreo Au, http://www.ventana.com/product/page?view=iscan
35. NDP. VIEW2, http://www.hamamatsu.com/jp/en/U12388-01.html
36. NanoZooner, http://www.hamamatsu.com/us/en/community/nanozoomer/index.html.
37. Coolscope VS 'WebSlide, http://www.nikoninstruments.com/
38. Bacus Laboratories, www.bloomberg.com/
39. Earth API, https://developers.google.com/earth/?hl=en
40. Google Map API , https://developers.google.com/maps/?hl=en
41. NYU School of Microscopio Virtual Medicina education.med.nyu.edu/virtualmicroscope/
42. MapTiler, http://www.maptiler.com/

La Facultad en 3D: un entorno común para dos propuestas educativas

Viviana Harari[1], Claudia Banchoff[1]

[1] Facultad de Informática – Universidad Nacional de La Plata (UNLP)
50 y 120. 2do. Piso. – La Plata (1900) Buenos Aires – Argentina.
{vharari, cbanchoff}@info.unlo.edu.ar

Resumen. La Facultad de Informática de la Universidad Nacional de La Plata recibe un promedio de 700 alumnos ingresantes por año, donde la gran mayoría nunca ha visitado el edificio o sus alrededores. En este contexto, se realizó un modelo 3D del edificio de la facultad de manera tal de poder mostrar las instalaciones en forma virtual. Este modelo fue utilizado en dos propuestas planteadas como trabajos de tesis de maestría que, abordando dos temáticas y estrategias distintas, permiten recorrer y conocer las instalaciones a futuros ingresantes. Este artículo describe ambas propuestas y los detalles del desarrollo del software 3D.

Palabras Claves: software libre, entornos 3D, enseñanza de programación, taller vida universitaria.

1 Introducción

La Facultad de Informática de la Universidad Nacional de La Plata es una facultad muy joven, a la cual ingresan en promedio alrededor de 700 alumnos por año. Las carreras que se dictan en la misma son muy demandadas, tanto en la región como en otras regiones del país. Muchos alumnos, principalmente los que provienen de regiones más lejanas, desconocen la zona en donde está situada y la instalación en particular.

En función de poder brindar a los alumnos que no son de la ciudad de La Plata la posibilidad de poder conocer la facultad y recorrerla como si estuvieran en el lugar, se pensó en realizar un modelo 3D representativo de la institución.

El software realizado, denominado FI3D permite básicamente, realizar un recorrido virtual por las distintas aulas, oficinas y jardines de la facultad, utilizando un avatar, pero sin ningún objetivo en particular. Algo importante de destacar es que, a través de este recorrido, también se puede acceder a una vista virtual de la terraza del edificio donde se encuentran instalados paneles solares, 72 módulos foto-voltaicos que inyectan energía a la red de la facultad y contribuyen al abastecimiento de energía del edificio. Si bien este escenario no fue tratado de manera diferente en las propuestas descriptas en este artículo, es un marco ideal para el planteo de otras nuevas iniciativas futuras.

Tomando FI3D como base, se plantearon dos propuestas que, abordando dos temáticas y estrategias diferentes, permiten recorrer y conocer las instalaciones a futuros ingresantes, con dos objetivos claros y específicos.

En la primer propuesta se plantea un "*espacio anticipatorio educativo basado en un juego de desafíos*"[1], a través del cual los alumnos ingresantes a la Facultad de Informática de la UNLP, puedan sensibilizarse con la entidad educativa antes de comenzar su vida universitaria. La propuesta abarca desde la familiarización con la infraestructura edilicia, permitiendo que el ingresante recorra e identifique las distintas dependencias de la institución, hasta las reglamentaciones administrativas relacionadas con el quehacer del alumno. Esta propuesta tiene como objetivo que a la hora de ingresar a la facultad, tanto el alumno como su entorno familiar, eviten tener sensaciones de ansiedad o angustia que muchas veces son producidas por la inseguridad de enfrentarse a lo desconocido, a lo incierto o a lo imprevisible.

La segunda propuesta se enmarca en el proyecto "*Programando con Robots y Software Libre*"[2], que desde hace más de 5 (cinco) años trabaja en la enseñanza de programación en las escuelas secundarias de la región. Muchos de los alumnos que participan en las actividades de este proyecto, luego ingresan a algunas de las carreras de la facultad, con lo cual pareció una buena estrategia presentar como escenario el mismo edificio que encontrarán al visitar las instalaciones reales.

En las siguientes secciones se describirá en detalle el software FI3D y las dos propuestas basadas en él.

2 FI3D: La Facultad en 3D

Uno de los objetivos planteados para este desarrollo fue la utilización de software libre. Se analizaron distintas herramientas y se decidió por el uso de Blender[3].

Blender es una herramienta ampliamente utilizada desde hace muchos años. Permite la creación de modelos 3D y 2D, animaciones, simulaciones, renderizado, composición y seguimiento de movimiento y creación de videojuegos. Si bien Blender es una herramienta con la cual se pueden crear aplicaciones sin escribir líneas de código en un lenguaje de programación, provee una API ("*Interfaz de Programación de Aplicaciones"*) que contiene las funciones y módulos que pueden ser utilizados por otro software. Esta API permite no sólo la codificación de los scripts que componen la aplicación, sino que permite personalizar la herramienta y ampliar su funcionalidad. La API provee dos módulos importantes para su uso: BGE (*"Blender Game Engine"*), que contiene la funcionalidad para acceder a los elementos que componen la aplicación que se intenta desarrollar y, BPY (*"Blender Python*"), que permite acceder a los datos, clases y funciones de Blender,

[1] Espacio virtual 3D representativo de un espacio físico particular, creado y diseñado con el propósito de permitir su acceso para recorrerlo, conocerlo y sensibilizarse con el lugar. Este espacio virtual, copia fiel de un espacio real, permitirá a cualquier persona, "anticiparse" al lugar donde desarrollará alguna actividad futura.

[2] Sitio oficial del proyecto: http://robots.linti.unlp.edu.ar

[3] https://www.blender.org/

permitiendo crear nuevos elementos y modificando el entorno mismo provisto por Blender.

Si bien en un primer momento sólo se pensó en utilizar esta herramienta para el modelado de los recursos necesarios, luego se decidió continuar con el desarrollo del software manteniendo el mismo en el mismo entorno.

El modelado de los recursos estuvo a cargo de Máximo Bibiloni, estudiante avanzado de la carrera Licenciatura en Diseño Multimedial, quien colaboró muy estrechamente en el desarrollo de este trabajo.

La figura 1 muestra algunas de las capturas de la aplicación FI3D desarrollada.

Figura 1. Capturas de distintos escenarios de FI3D.

La decisión de utilizar Blender se basó principalmente en dos factores: el primero y más importante es que es software libre y existe una gran comunidad y recursos disponibles. Esto fue fundamental, dado que en todo momento este desarrollo se pensó como base para otras aplicaciones, con lo cual, tanto los recursos como el software deberían estar disponibles para ser extendidos ya sea por el mismo grupo de trabajo como por personas externas al mismo. El segundo factor tuvo que ver con que, el lenguaje de scripting utilizado por la herramienta es Python. El peso que tuvo el lenguaje de programación, en la elección de la herramienta, se debió a que las desarrolladoras a cargo de FI3D tienen experiencia en el mismo y son docentes desde hace años de la cátedra de Seminario de Lenguaje - opción Python. Además, cabe destacar que la herramienta Blender ha sido utilizada en la cátedra como recurso, para el desarrollo de los trabajos finales de los alumnos.

Algo de destacar sobre Blender es que existe un plugin para web mediante el cual es posible visualizar las aplicaciones en algunos navegadores y es posible generar aplicaciones que se ejecuten en dispositivos móviles. Si bien estas dos características no fueron determinantes para su elección, la posibilidad de exportar los objetos

creados a otras plataformas si lo fue, dado que se pretende continuar este desarrollo en un entorno multi usuario[4].

3 Propuestas educativas

FI3D simplemente permite realizar un recorrido virtual sin ningún objetivo específico. A través de un único avatar se puede ir recorriendo los distintos espacios tanto de la planta baja como de los pisos superiores[5].

Como se mencionó anteriormente, FI3D fue usado como base para plantear dos propuestas en las cuales se debe recorrer el lugar pero con dos objetivos y estrategias específicas:

- **TIVU Virtual**, un juego serio que plantea el uso de espacios anticipatorios;
- **ProBOTs 3D**, un entorno de aprendizaje en 3D para enseñar programación a niños y jóvenes con herramientas libres.

Actualmente ambas propuestas se encuentran en etapa final de su implementación.

3.1 TIVU Virtual

Tivu Virtual es un juego serio [1] cuyos destinatarios son los nuevos ingresantes a la Facultad de Informática de la UNLP. El mismo representa un lugar físico particular (la facultad), al cual por un lado, se lo podrá acceder para recorrerlo y aprender la ubicación de las dependencias más importantes de la entidad y los trámites administrativos que se realizan en ellas. Pero, por el otro lado, se lo podrá acceder para comprobar si se ha logrado familiarizar tanto con el lugar como con las dependencias y sus funciones.

A este tipo de entorno se lo define como un "*espacio anticipatori*o", dado que es un espacio físico particular, creado y diseñado para poder accederlo y recorrerlo con el fin de familiarizarse con el lugar. Es un medio por el cual las personas pueden, en forma virtual, "anticiparse" y conocer un lugar particular al cual se supone irán. Esto permite que a la hora de llegar a él, en forma real, se lo haga sin los temores e incertidumbres que muchas veces provoca el enfrentarse a lo desconocido [2, 3].

Preocupados por la integración de los nuevos ingresantes, muchas universidades y facultades de todo el mundo implementan talleres o cursos orientados a favorecer la socialización de los mismos a las entidades. La mayoría de ellos trabajan la integración académico-disciplinar, ya que los jóvenes deben incorporar y asimilar los conocimientos, habilidades y valores propios del currículo de la disciplina, haciéndolo a través de distintas prácticas que se desarrollan en aulas, laboratorios y talleres. Este proceso se denomina "*alfabetización académica*" y según la definición de la autora Paula Carlino [4], dicho concepto se refiere al "*proceso de enseñanza que puede (o no) ponerse en marcha para favorecer el acceso de los estudiantes a las diferentes culturas escritas de la disciplina*". Si bien es necesario este tipo de sensibililización,

[4] En un principio se piensa exportar el trabajo a OpenSimulator

[5] En esta versión del software no está habilitado el acceso a los ascensores del edificio, sino que siempre se utiliza las escaleras para trasladarse de un piso a otro.

los alumnos ingresantes también necesitan adaptación en el lo que se refiere a los aspectos administrativos de la organización. Esta temática muchas veces no es tratada por las entidades y son temas sensibles que deben ser considerados, dado que también influyen en la adaptación a la organización. El autor Alain Coulon [5], se refiere a este proceso con el nombre de "*afiliación institucional*". Según el autor, la incorporación de los jóvenes a la vida universitaria es el inicio de un proceso de aprendizaje que él mismo denomina: "*aprender el oficio de estudiante*" y el proceso de afiliación institucional forma parte de ese camino inicial que el alumno debe recorrer.

Por otra parte, TIVU Virtual fue concebido como un juego serio. El concepto de "*juegos serios*" o "*serious games*" fue introducido formalmente por Clark C. Abt. [6] en donde los definía como juegos que tienen un propósito educativo explícito y cuidadosamente planeado y, que no están pensados únicamente para ser jugados por diversión. A modo de ejemplo, un tipo de juego que caía en esta categoría eran aquellos que simulaban eventos de la primera guerra mundial, y que recreaban las estrategias de guerra en el aula de clases. Igualmente, se hace notar que esto no implica que los juegos serios no sean divertidos ni atractivos para aquellos que los juegan.

Los juegos serios basados en software como videojuegos han tenido un gran auge en la última década y han sido y son utilizados en varios campos de aplicación. En salud existen juegos que se han utilizado con pacientes con diferentes capacidades tanto físicas como intelectuales; en educación se han utilizado este tipo de juegos en los procesos de enseñanza-aprendizaje en los diferentes niveles educativos (educación primaria, secundaria y universitaria); en el campo científico, se están desarrollando juegos o plataformas para resolver problemas o para ayudar en investigaciones; en la industria, se utilizan este tipo de juegos para capacitación, entrenamiento, reclutamiento, gestión, etc.

TivuVirtual es un juego serio que permite a los alumnos ingresantes a la Facultad de Informática de la UNLP, familiarizarse con el lugar físico en donde pasarán gran parte de sus días, y con los procesos administrativos más comunes que realizarán a lo largo de sus vidas universitarias. El juego tendrá un valor agregado, y será la posibilidad de brindar al alumno información complementaria de autorregulación que le permita organizar mejor su vida universitaria.

3.2 ProBOTs 3D

ProBots 3D es un entorno que permite realizar las actividades prácticas de un curso elemental de programación en lenguaje Python que utiliza robots. Este curso se enmarca en el proyecto "*Programando con Robots y Software Libre*" en el cual se trabaja desde hace ya varios años [7]. Si bien los conceptos generales son los mismos que se imparten en forma presencial en el proyecto, las actividades prácticas se deben adecuar para que puedan ser desarrolladas dentro del mundo virtual basado en FI3D.

En el proyecto "*Programando con Robots y Software Libre*" los talleres se dictan en forma presencial, en alguna de las aulas de la facultad, y se utilizan los robots Múltiplo N6, adquiridos a la empresa RobotGroup, para realizar las distintas actividades. El mismo está basado tanto en hardware como en software libre. La

figura 2 muestra una imagen del robot real con el que se trabaja y el recurso 3D modelado en Blender.

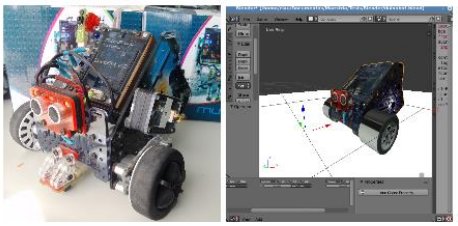


Figura 2. Robot Múltiplo N6 real y Múltiplo N6 modelado en Blender.

Este proyecto tiene una gran aceptación e impacto, pero el principal inconveniente es la disponibilidad de los robots físicos. Para intentar resolver este inconveniente, en una primer instancia se desarrolló un simulador 2D [7] cuyo código se encuentra disponible en github[6]. Si bien el trabajo con el simulador es muy distinto a trabajar con el robot real, permite contar con una herramienta disponible para cualquiera que así lo desee, y establecer la misma como un elemento de motivación para preparar a los estudiantes en la enseñanza de la programación con Python e intercalar algunas actividades con el robot real.

Al momento de proponer el desarrollo de ProBots 3D se plantearon dos objetivos principales:

- Por un lado, brindar, de forma similar que el simulador 2D, un entorno de trabajo que no requiera de la adquisición de elementos físicos adicionales, como lo son los robots. En este aspecto, mantiene las mismas ventajas y desventajas en su uso que con el simulador 2D;
- Por el otro, y el más destacado en nuestra investigación, es analizar el impacto que tiene la inclusión de un entorno más motivador, en este caso un entorno 3D, para el estudiante.

ProBots 3D se encuentra en etapa final de su desarrollo. Los alumnos accederán a la facultad virtual y podrán realizar las actividades prácticas del taller de programación de Python en la biblioteca, ubicada en la planta baja del edificio. Los estudiantes deberán ingresar a las instalaciones, ubicar la biblioteca y en este lugar completar las lecciones. Cada lección está relacionada a los distintos conceptos que se imparten en el curso. El estudiante podrá manipular un robot virtual e ingresar las instrucciones para resolver las consignas y, en los encuentros presenciales probar sus programas con los robots reales.

La forma en que se planteó esta propuesta y los distintos mecanismos de interacción formulados, permitirá utilizar ProBots 3D con niños más pequeños para enseñar los primeros pasos de la programación en la escuela primaria. Esta propuesta

[6] https://github.com/Robots-Linti/pilas-robots

fue aceptada en varias instituciones en las cuales se trabaja con Scratch[7] o LightBot[8] y se formuló un proyecto con las mismas para el año 2016.

4 Conclusiones

FI3D fue pensado y realizado con el objetivo de poder ofrecer a los futuros ingresantes de la Facultad de Informática de la UNLP un espacio anticipatorio, que les permita sensibilizarse con el lugar donde iniciarán sus estudios universitarios.

Por ser software libre pudo ser utilizado como base para dos propuestas de tesis de maestría: una que se enmcarca en las acciones relacionadas al proceso de ingreso a la facultad (TivuVirtual), y otra que propone una contribución al proyecto de extensión "Programando con Robots y software Libre", incorporando un nuevo recurso que permite encarar actividades para enseñar programación a niños y jóvenes de edad escolar.

La importancia de hacer uso de software libre no sólo permite poder reutilizarlo para desarrollar nuevas aplicaciones, sino que también permite escalarlos y modificarlos para que las mismas puedan ser adaptadas y ajustadas a otras necesidades. Teniendo en cuenta esto y, pensando que el escenario virtual de las instalaciones de la facultad podría servir como base para propuestas de trabajos de alumnos, se prefirió desarrollar el trabajo con Blender. Si bien la herramienta utilizada presentó algunas limitaciones, se pensó continuar con ella en esta etapa de desarrollo y luego, en una segunda etapa, migrar a otra herramienta de software libre. Si bien existen otras herramientas que son privativas, muy populares y que podrían ser más eficientes a la hora de poner en producción el software, el mismo no podría ser reutilizado para otros contextos.

Actualmente ambos desarrollos están en la etapa de finalización de la construcción de sus respectivos prototipos y se estima que a finales del año actual puedan ser presentados.

References

1. Diego G. Vilches Antão: Juegos Serios, evaluación de tecnologías y ámbitos de aplicación. Tesis de Especialista en Ingeniería de Software. 2014.
2. Oihab Allal-Chérif: Using serious games to recruit, integrate and train your employees: an exploratory studiy of practices. PhD in Management Siciences.
3. Michael, D. y Chen, S: Serious Games: Games That Educate, Train, and Inform. Course Technology PTR, 2005.
4. Paula Carlino: Red de Revistas Científicas de América Latina, el Caribe, España y Portugal. Sistema de Información Científica. “Alfabetización académica diez años después”. Revista Mexicana de Investigación Educativa, vol. 18, núm. 57, 2013, pp. 355-381 Consejo Mexicano de Investigación Educativa, A.C. Distrito Federal, México. Disponible http://www.redalyc.org/articulo.oa?id=1402577400

[7] https://scratch.mit.edu/
[8] http://lightbot.com/

5. Coulon, A.: Le Metier d'Estudiant. L'entree dans la vie universitaire. Ed. Económicas. Paris. 2005
6. Abt, C. Serious Games. Viking Press, 1970.
7. Javier Díaz, Claudia Banchoff, Sofía Martin, Fernando López: Aprendiendo a programar con juegos y robots. Publicado en el "VII Congreso de Tecnología en Educación y Educación en Tecnología. TE&ET 2012" en la Ciudad de Pergamino, los días 11 y 12 de Junio de 2012. ISBN 978-987-28186-0-9
8. Javier Diaz, Claudia Banchoff, Sofía Martin, Andrea Gómez del Mónaco y Einar Lanfranco: Robots a pilas. Presentado en CISL: Conferencia Internacional de Software Libre en Argentina. CABA. Argentina. Octubre de 2013.

Definición de Metadatos Educativos para Repositorios de Instituciones de Educación Superior

Carlos Giorgetti[1], Lucila Romero[1] y Milagros Gutierrez[2],

[1]Facultad de Ingeniería y Ciencias Hídricas – UNL, cgiorgetti@fich.unl.edu.ar, lucila.rb@gmail.com. [2]Facultad Regional Santa Fe – UTN, milagros.gutierrez@gmail.com.

Abstract. En la era de las redes y tecnologías de la información, favorecer la comunicación abierta de los resultados de la investigación científica y de la producción académica de las universidades se ha convertido en un compromiso de científicos, instituciones, docentes, bibliotecarios y otros administradores de la información. La nueva situación requiere plantear políticas y mecanismos de comunicación diferentes, en los que los flujos de difusión sean más rápidos y eficaces. A raíz de esto, el movimiento para el Acceso Abierto ha desarrollado, en muy poco tiempo, un alto nivel de madurez en la renovación de los modelos de comunicación científica y el planteamiento de alternativas sostenibles a los modelos existentes. En este contexto, un nuevo desafío emerge: la necesidad de describir correctamente a los recursos almacenados en los repositorios de manera que puedan ser descubiertos por motores de búsquedas y, de esta manera, facilitar su localización. Para afrontarlo es necesario tener en cuenta los diferentes tipos de Objetos Digitales Educativos (ODE) que pueden ser compartidos y la variedad de estándares propuestos para la descripción de los mismos. Esta diversidad afecta sensiblemente la interoperabilidad de la información y particularmente su búsqueda, recuperación y reuso. Este trabajo propone un conjunto de colecciones y de metadatos adecuados para cada una de ellas, para la descripción de la producción científico-tecnológica y académica de las Instituciones de Educación Superior que deben ser utilizados en los repositorios para la identificación precisa de los ODE, obtenidos como resultado del análisis y comparación entre los estándares de definición de metadatos más utilizados LOM (Learning Object Metadata), DC (Dublin Core) y, considerando las directivas del Sistema Nacional de Repositorios Digitales (SNRD).

Keywords: Objeto digital educativo, acceso abierto, sistema nacional de repositorios digitales, Instituciones de educación superior, metadatos.

1 Introducción

En el año 2011, el Ministerio de Ciencia Tecnología e Innovación de Argentina en el marco de las políticas que impulsa su Secretaría de Articulación Científico-Tecnológica (SACT), crea el Sistema Nacional de Repositorios Digitales (SNRD)[1],

[1] http://repositorios.mincyt.gob.ar/recursos.php

con el propósito de impulsar la conformación de una red interoperable de repositorios digitales, distribuidos físicamente y coordinados a través de la Web. El principal objetivo de esta iniciativa es aumentar la visibilidad y el impacto de la producción científica y tecnológica, mediante el acceso gratuito para leer, descargar, copiar, distribuir, imprimir, buscar y enlazar los textos científicos y usarlos con fines legítimos a la investigación y la educación (Ley 26899)[2].

En este trabajo se pone énfasis en el aspecto tecnológico de este tipo de repositorios, de manera de aportar una solución al problema de la descripción de los recursos almacenados en ellos para que puedan ser compartidos y reutilizados. Estos recursos se denominan objetos digitales educativos (ODE) ya que su finalidad última, en un contexto de educación superior, es que estén disponibles para el proceso de enseñanza aprendizaje [1]. Su descripción adecuada asegura la supervivencia, cuando es apropiada, para ser interpretada por máquinas. Es decir, cuando se deposita material en un repositorio y se quiere que el mismo esté disponible para la comunidad, es fundamental que la descripción que se haga del mismo no sólo pueda ser entendida por lectores humanos sino que el desafío es la interpretación que los sistemas automáticos puedan hacer de ellos.

Para proveer interoperabilidad en las operaciones de acceso y búsqueda se requiere información estandarizada. De esta forma surgen diversos estándares de metadatos para la descripción de ODE. Diferentes comunidades, instituciones educativas, utilizan estos estándares adecuándolos a sus necesidades. Algunos utilizan estándares de metadatos que no fueron pensados para la descripción de ODE sino para describir recursos web en general (por ej. DC[3]), encontrándose con la dificultad de no poder describir características educativas asociadas al ODE. Por otro lado, existen otros estándares que si bien fueron diseñados para describir ODE, como el caso de LOM [2], propone una gran cantidad de metadatos que difícilmente sean provistos con la información correcta en el momento de la carga en el repositorio. Se hace necesario entonces contar con un conjunto de metadatos lo suficientemente expresivo como para abarcar todas las características de un ODE y a la vez lo suficientemente práctico como para facilitar su carga.

Se propone la descripción de ODE en colecciones, identificando para cada una el conjunto de metadatos apropiados. Para ello se consideraron los estándares más comúnmente usados como son DC y LOM; a su vez se tuvieron en cuenta las directivas del SNRD para proponer la manera en que los metadatos deben ser cargados y la ponderación de los mismos como optativos y obligatorios. Así, esta propuesta pretende ser una recomendación a ser considerada por las instituciones educativas de nivel superior principalmente en nuestro país, para la descripción de sus recursos en los repositorios institucionales (RI).

Este trabajo está organizado de la siguiente manera. En el punto 2 se presentan conceptos preliminares para el desarrollo del trabajo: estándares de metadatos y RI de acceso abierto (AA), en el punto 3 se presenta una propuesta de descripción para materiales disponibles en RI de AA. En la sección 4 se presentan las conclusiones y trabajos futuros.

---

[2] http://www.infoleg.gob.ar/infolegInternet/verNorma.do?id=223459

[3] http://dublincore.org/

2 Preliminares

2.1 Estándares de metadatos

Un estándar de metadatos tiene como objetivo alcanzar acuerdos respecto de las características de un recurso. Particularmente en el contexto de educación basada en tecnologías, e-learning, el uso de estándares de metadatos facilita dotar al mismo de las destrezas propuestas por el Elearning Consortium [3], [4]: (i) Accesibilidad, para localizar materiales instruccionales independientemente de cuál sea su ubicación física. (ii) Adaptabilidad, para describir el contenido y pueda ajustarse a las necesidades de los usuarios. (iii) Asequibilidad, para acrecentar la eficiencia y productividad en relación al tiempo y los costos. (iv) Durabilidad, para disminuir el desarrollo de nuevas versiones en función de los cambios tecnológicos. (v) Capacidad de gestión, para realizar el seguimiento del proceso de aprendizaje. (vi) Reusabilidad, para adaptar los ODE a distintos sistemas con el mismo estándar.

Existen varios modelos de metadatos, estándares, entre los que se encuentran:

- Dublin Core (DC): Es simple pero no específico para recursos educativos, por esta razón no es completamente compatible con la descripción de los mismos, aumentado por la dificultad de pretender describir objetos de tan diversa índole.
- Learning Object Metadata (LOM): es específico para describir objetos de aprendizaje, pero es muy amplio, ya que tiene 72 descriptores de los recursos, haciendo que la tarea de descripción sea un proceso complejo [2].
- IMS CP: La especificación Intelligent Manufacturing System Content Packaging define un estándar para el formato digital de los paquetes de contenidos educativos. Define una notación pedagógicamente neutral que permite crear diseños educativos reutilizables [5].
- MIMETA: este modelo consta de dos partes, un conjunto de elementos de descripción general y uno de descripción de las características educativas de los recursos. Es un modelo de 21 elementos de dispar importancia [6].
- MetaRED (Metadatos para Recursos Educativos Digitales): propone como un esquema de metadatos elementos de DC y LOM agrupado en 4 categorías, 3 subcategorías y 22 elementos [7].
- CanCore: asegura la interoperabilidad de metadatos y recursos entre los usuarios canadienses. Su foco principal es la localización de recursos. Recomienda la utilización de los elementos de LOM relacionados con el intercambio e interoperabilidad de recursos [8].
- SCORM: Propone un perfil de aplicación para definir ambientes de e-learning basado en diferentes estándares y especificaciones como IEEE LOM, IMS [4].

Cada estándar o especificación varía la forma y estructura de los metadatos. Casi todos utilizan XML (eXtensible Markup Language) como lenguaje de marcado, asegurando con ello la independencia del medio y la interoperabilidad [9].

Uno de los problemas actuales es que la descripción del recurso tiene información imperfecta, incompleta, imprecisa y/o poco confiable [10]. La preocupación de los especialistas por describir correctamente un recurso digital se basa en la utilidad que tiene tanto para el que necesita usar el recurso (usuario o cliente) como también para el que desea disponerlo para su distribución (autor). Esto está en relación al esfuerzo

para ubicar el recurso dentro del repositorio y también para catalogar, indexar, buscar y cargar el material en el contenedor.

Actualmente no existe mucha experiencia en el diseño de repositorios de ODE, y por lo tanto se está en las primeras fases de definición de las pautas que orientan la construcción de los mismos [11].

2.2 Repositorios de Acceso Abierto

En la era de las redes y tecnologías de la información, favorecer la comunicación abierta de los resultados de la investigación científica y de la producción académica se ha convertido en un compromiso de científicos, instituciones, docentes, bibliotecarios y otros administradores de la información.

En este contexto, "el movimiento para el Acceso Abierto (AA) ha desarrollado en muy poco tiempo un alto nivel de madurez en la renovación de los modelos de comunicación científica y el planteamiento de alternativas sostenibles a los modelos existentes" [12]. AA es el término utilizado para describir la libre y permanente disponibilidad en Internet de la producción científica y académica, permitiendo a cualquier usuario su lectura, descarga, copia, impresión, distribución o cualquier otro uso legal de la misma, sin ninguna barrera financiera, técnica o de cualquier tipo, reservando a los autores el control sobre la integridad de sus trabajos y el derecho de ser adecuadamente reconocidos y citados [13].

Un aspecto fundamental más que plantea el movimiento para el AA es que los documentos deben estar depositados en un repositorio que permita la interoperabilidad en base al cumplimiento de una serie de protocolos para que proveedores de servicios y motores de búsqueda puedan recolectar los metadatos de estos repositorios, creando así las condiciones para que los productos de la investigación estén universalmente disponibles. Los repositorios son la tecnología que posibilita un mejor acceso a los ODE [14], aunque se debe mencionar que existen otros modos de acceso como los entornos virtuales de aprendizaje (EVA), revistas abiertas, sitios web temáticos, wikis, comunidades virtuales, redes sociales, entre otras [15].

Los repositorios ofrecen varias ayudas a los ODE disponibles dentro de esta modalidad: preservación y reutilización de contenido [16], acceso permanente y mayor visibilidad [17], y facilidad de la búsqueda y recuperación mediante el uso de los metadatos, que los identifican y permiten su intercambio entre diferentes repositorios y EVA [18] y diferentes métodos [19].

En Argentina se dio origen a la Ley 26899: Creación de Repositorios Digitales Institucionales de AA, Propios o Compartidos. La ley exige que los organismos e instituciones públicas con financiamiento del Estado, desarrollen repositorios digitales institucionales de AA en los que se depositará la producción científico-tecnológica que sea resultado de la realización de actividades de investigación y que atraviese un proceso de evaluación de calidad, haya sido ésta publicada o no. Asimismo, las obliga a establecer políticas para el acceso público a datos primarios de investigación a través de repositorios digitales institucionales de AA, como así también políticas institucionales para su gestión y preservación a largo plazo.

2.3 Descripción de los ODE en Repositorios de Acceso Abierto

La descripción del material almacenado en los repositorios se realiza a través de un conjunto de atributos que constituyen los metadatos, estos son datos estructurados que describen las características de los recursos educativos, esto es, proveen información del ODE. También puede considerarse que un metadato es una pieza de información que está unida al núcleo de contenido del objeto al cual describe. El conjunto de metadatos que describe al objeto debe ser sólido conteniendo información relativa a distintos aspectos del objeto, como por ejemplo ciclo de vida, requisitos técnicos, especificaciones educativas, derechos de autor, clasificación entre otros [8]. Al realizar una búsqueda de ODE en un repositorio, lo que se busca es la información contenida en los objetos. Las operaciones de acceso y búsqueda requieren que la información esté lo más estandarizada posible, que es el motivo fundamental de la aparición de los estándares en relación a los metadatos de descripción de estos objetos de manera de favorecer la interoperabilidad de la información.

3 Propuesta de descripción para ODE en Repositorios de AA

El proyecto "Hacia el desarrollo y uso de Repositorios de AA para los Objetos Digitales Educativos en el contexto de las universidades públicas en el centro-este de Argentina" (PICTO-CIN N°143), fue llevado a cabo por la Universidad Nacional de Rosario, la Universidad Nacional del Litoral y la Universidad Tecnológica Nacional (Regional Santa Fe y Regional Rosario). Su objetivo fue proporcionar un primer modelo teórico para la implementación de RI en estas organizaciones, junto a la metodología para su adaptación. Además propuso una tecnología experimental para los repositorios de acceso abierto, para contribuir en diferentes aspectos de las organizaciones educativas [20]:

- Pedagógico-organizacional: referido a los aspectos tecnológicos y sociales del desarrollo de estrategias socio-culturales para promover la adopción de repositorios institucionales.
- Tecnológico: mediante el diseño y desarrollo de herramientas informáticas de código abierto a través de procesos técnicos y procedimientos relacionados con los metadatos, almacenamiento de información, extracción automática de metadatos, diseño de interfaces, accesibilidad e interoperabilidad.

3.1 Colecciones definidas para un repositorio en instituciones de educación superior

Una de las primeras actividades del proyecto fue definir las colecciones (categorías) en las que estará clasificado el material del repositorio, es decir los ODE que lo integrarán, las que se definieron como sigue:

- Libros
- Capítulo de libros
- Revistas
- Artículo en Revista con referato

- Comunicaciones en Congresos, Jornadas, Conferencias/Póster
- Tesis de Posgrado: Maestría/Doctorado.
- Tesis de Grado: Proyectos finales de carrera/Tesinas/Tesis
- Producción en Investigación: Proyecto de Investigación/Informe de Avance /Informes Técnicos/Documento de Trabajo/Software/Póster
- Producción en tecnología: Patente/marca /modelo industrial/modelo de utilidad / Informe técnico/Documento de trabajo/Software/Póster
- Producción en extensión: Convenios/Actas Acuerdo/Informes Técnicos/ Documento de Trabajo/Software
- Producción en arte: Película/documental/Videograbación/Fotografía/Sonido/ Música/Software/Póster
- Producción de material didáctico
- Reseña: Revisión de artículo/Revisión literaria
- Datos primarios: Conjunto de datos/Plano/ Mapa/Imagen Satelital/Radiografía
- Otros

Cada una de estas colecciones debe ser descripta a través de un conjunto de metadatos específicos y adecuados cumpliendo con las destrezas enunciadas, permitiendo su localización en forma económica y eficiente.

3.2 Selección de metadatos para la descripción adecuada de ODE

Teniendo en cuenta la clasificación de los ODE en las diferentes colecciones, se realizó un estudio de los estándares de metadatos para determinar su relevancia para la descripción de los mismos.

Se analizaron los estándares LOM y DC. En primer lugar se consideraron los metadatos del estándar DC dado que dicho estándar posee metadatos generales para la descripción de recursos. Se seleccionaron metadatos como dc:creator (autor del recurso), dc:publisher (editor del recurso), entre otros. El prefijo dc: indica el conjunto de términos del estándar Dublin Core.

Luego se realizó un proceso de correspondencia entre los estándares DC y LOM considerando el aspecto sintáctico de los metadatos. Es decir, se buscaron coincidencias exactas en el nombre de los metadatos. El prefijo lom: hace referencia a los elementos de dicho estándar. Algunos ejemplos de elementos equivalentes desde su sintaxis son los siguientes [21]:

- dc:title y lom:general.title. Ambos metadatos describen el título. En el caso de DC describe el título de un recurso web y en el caso de LOM describe el título de un ODE.
- dc:MediaTypeOrExtent.format y lom:technical.format. Ambos metadatos describen el formato técnico de un ODE.

En forma paralela se realizó un proceso de comparación considerando los aspectos semánticos de ambos estándares. En este caso se analizaron y compararon metadatos en cuanto a su significado aunque no coincidieran en su nombre. Algunas de las correspondencias identificadas son:

- dc:contributor y lom:lifecycle.contribute.role.author ambos términos representan el diseñador o autor de un ODE.

• dc:RightsStatement y lom:Rights, ambos metadatos permiten describir los derechos de autor de un recurso/ODE.

• dc:subject y lom:general.keyword. De acuerdo con Dublin Core, subject se utiliza para identificar los elementos básicos en un recurso, mientras que la palabra clave se utiliza para identificar las frases que describen el tema del ODE

• dc:relation y lom:relation.resource. Ambos se refieren a otros recursos identificados con los que se relaciona el recurso descripto por el metadato.

Cabe aclarar que el estándar LOM contempla más de 70 metadatos que enriquecen la descripción de ODE. Si bien todos los metadatos son importantes es necesario tener en cuenta el tiempo que demanda para el autor de una producción la carga de esos metadatos al momento de la incorporación de la misma en un repositorio. Por lo tanto, como paso siguiente del análisis, se realizó una selección de los metadatos del estándar considerados más relevantes para la localización de los mismos. Se hizo énfasis en la categoría *Educational* dado que es la que describe las producciones desde una perspectiva pedagógica y, por lo tanto, se consideró relevante para favorecer el trabajo colaborativo de los docentes. La categoría *Educational* considera descriptores como *Audience*, *Context*, *Difficulty*, *IntendedEndUserRole*, *InteractivityLevel*, *InteractivityType*, *TypicalAgeRange* y *TypicalLearningTime*. Estos metadatos describen la audiencia destinataria del ODE, su contexto educativo, el nivel de dificultad del mismo, el rol pretendido del usuario final y el tipo y nivel de interactividad que demanda. Es decir, un profesor, al momento de considerar la posibilidad de reutilizar un material didáctico (apunte de cátedra, presentación de una clase, una evaluación) desarrollado y compartido por otro docente, tiene la posibilidad de conocer para qué tipo de alumnos fue pensado dicho material (nivel secundario, educación superior, posgrado, etc.), el nivel de dificultad que representa para ese contexto educativo entre otras características que lo pueden ayudar a considerar la posibilidad de reutilizar el mismo para su propia situación educativa.

Finalmente, cada metadato fue clasificado como Obligatorio / Útil / Deseable, indicando el grado de obligatoriedad en la carga del metadato para un autor o responsable de la incorporación de una producción a un repositorio. Si el metadato es calificado como Obligatorio no se puede continuar la carga de la producción en el repositorio si un metadato de este tipo está ausente. Si es Útil, el ODE puede ser almacenado en el repositorio pero no será público hasta que el autor o responsable de la carga no complete la carga total de todos los metadatos útiles. Si es Deseable, se trata de un metadato opcional y su ausencia no provoca una interrupción de la carga o publicación de la producción en el repositorio.

La clasificación en estos tres tipos fue pensada por dos motivos fundamentales: primero no permitir la carga de una producción sin completar los metadatos indispensables para que luego esa producción pueda ser localizada y recuperada por otros usuarios del repositorio; segundo, para simplificar la carga de los metadatos estableciendo prioridades entre los mismos y permitiéndoles a los autores la posibilidad de completar los descriptores con mayor comodidad una vez que la producción ya se encuentre disponible en el repositorio

Cabe aclarar que se estudió de manera especial a cada una de las colecciones definidas, encontrando particularidades en su descripción. Por ejemplo, para un libro el metadato que refiere al contexto educativo es de carga opcional, pero para un material educativo es un metadato obligatorio.

En la tabla I se muestran los metadatos seleccionados para la descripción de la colección libro. Los metadatos se encuentran en la columna *Término* en el idioma original del estándar (inglés) pero también poseen una etiqueta que traduce al español su nombre mostrada en la columna *Etiqueta*. Para cada metadato se identifica el esquema al que pertenece, es decir, se especifica si se encuentra en el estándar LOM, en Dublin Core o en ambos usando el prefijo correspondiente.

Como puede observarse en la tabla, el metadato *title* se encuentra presente en ambos estándares y tiene carácter obligatorio, es decir, si no se ingresa un valor para ese metadato no puede finalizarse la carga del libro.

La columna *Repetible* indica si el metadato tiene un valor simple o puede contener múltiples valores. Por ejemplo el metadato correspondiente al *autor* se define como repetible, dado que un libro puede tener varios autores.

Tabla 1. Selección de metadatos para la descripción de la colección Libro.

Esquema	Término	Carácter	Repetible	Etiqueta
dc/lom	Title	obligatorio	SI	Título
dc/lom	creator/life cycle.contribute.role.author	obligatorio	SI	Autor/(es)
dc/lom	contributor/life cycle.contribute	opcional	SI	colaborador
dc/lom	description/general.description	obligatorio	NO	Resumen
dc	Description	opcional	SI	Filiación
dc/lom	subject/general.keyword	obligatorio	SI	Palabras Clave
dc/lom	publisher/life cycle.contribute.role. Publisher	obligatorio	NO	Editor
dc/lom	date/life cycle.contribute.date	obligatorio	NO	Fecha de publicación
dc/lom	type/educational.learningResourceType	opcional	NO	Tipo
dc/lom	format/technical.format	obligatorio	NO	Formato
dc/lom	language/general.language	obligatorio	NO	Idioma
lom	educational.typical age range	opcional	SI	Rango de edad
dc/lom	identifier/general.identifier.entity	obligatorio	SI	ISBN
dc/lom	rights/right.description	obligatorio	NO	Titular de los Derechos
dc	Rights	opcional	NO	Contenido de los Derechos (texto)
dc/lom	relation/relation	opcional	SI	Relación
lom	Audience	opcional	NO	Audiencia
lom	educational.context	opcional	NO	Contexto
lom	educational.difficulty	opcional	NO	Dificultad
lom	educational.interactivity	opcional	NO	Tipo de interactividad
dc/lom	source/relation.resource	opcional	NO	Fuente

4 Conclusiones y trabajos futuros

El análisis presentado sirve como guía para desarrolladores e implementadores de repositorios institucionales de acceso abierto que buscan mejorar la localización de las producciones de dichos repositorios.

Se presentaron los principales conceptos a tener en cuenta en el desarrollo de un RI y principalmente se abordó el problema de la utilización de metadatos para la descripción de ODE en dichos repositorios. Se presentaron las principales dificultades que deben ser consideradas en el momento de decidir qué metadatos utilizar para la

correcta descripción de los recursos de acuerdo a los objetivos que cada comunidad se plantee. Se puso de manifiesto que existen diferentes estándares y la necesidad de considerar metadatos provenientes de diferentes estándares, también se realizó una comparación entre dos de los estándares más ampliamente usados DC y LOM poniendo de manifiesto equivalencias tanto sintácticas como semánticas.

También se presentaron las colecciones que deben ser consideradas en un repositorio para Universidades Públicas en la República Argentina. Cada una de estas colecciones tiene características particulares que obligan a utilizar diferentes descriptores. Para cada una de ellas entonces se elaboró una descripción completa de los metadatos que mejor describen a los ODE de la colección indicando su obligatoriedad, formas de introducir los valores de estos metadatos, estructuras que deben tener de acuerdo a los lineamientos del SNRD [22], etiquetas a utilizar y estándar al que pertenece.

El desarrollo presentado es útil para docentes e investigadores que desean reutilizar material de enseñanza disponible en repositorios institucionales ya que su correcta descripción al momento de la carga favorece la localización y recuperación. También es útil para los docentes e investigadores al momento de la carga de la producción en el repositorio, ya que esta tarea que consume gran cantidad de tiempo se facilita a través de la selección y ponderación de los metadatos identificando como obligatorios únicamente a aquellos descriptores imprescindibles para la posterior localización.

Como trabajo futuro se prevé la incorporación de la selección y mapeo realizados a las plataformas de software para repositorios DSpace mediante la utilización de tecnologías semánticas.

Referencias

1. Asociación Española de Normalización y Certificación.www.aenor.es. UNE 71361. Perfil de aplicación LOM-ES para etiquetado normalizado de Objetos Digitales Educativos (ODE).
2. IEEE 1484.12.1-2002. 15 July 2002. Draft Standard for Learning Object Metadata. Sponsored by the Learning Technology Standards Committee of the IEEE. Disponible en: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf, consultado el 25/04/2012.
3. Elearning Consortium.: Making sense of learning specifications and standards: A Decision Maker's Guide to their Adoption. Industry Report. The MASIE Center. Marzo de 2002. Saratoga Springs, NY. Center. http://www.masie.com/standards/S3_Guide.pdf
4. SCORM: Sharable Content Object Reference Model v1.3. 3rd. Ed. Advanced Distributed Learning (ADL). Octubre, 2006.
5. IMS Content Packaging. http://www.imsglobal.org/content/packaging. Consultado el 10/9/2013.
6. Marzal García-Quismondo, M. A., Calzada Prado, J. & Cuevas Cerveró, A. Desarrollo de un esquema de metadatos para la descripción de recursos educativos: El perfil de aplicación MIMETA. Revista Española de Documentación Científica N° 29, 4, octubre-diciembre, 551-571, 2006. ISSN 0210-0614.
7. Fager, J.; Beguerié, G. y Alonso, G. Diseño de Metadatos para Descripción de Recursos Educativos Digitales. Escuela Universitaria de Bibliotecología y Ciencias Afines de la Universidad de la República. Consultado: 1/11/2011.
8. Friesen, N.; Roberts, A. & Fisher, S. CanCore: Metadata for Learning Objects. Canadian Journal of Learning and Technology. Volume 28(3) Fall / automne, 2002.

9. Berlanga, A., García Peñalvo, F.J. Introducción a los Estándares y Especificaciones para Ambientes e-learning. In F.J. Garcia and M.N. Moreno (Eds.), Proceedings of Tendencias en el Desarrollo de Aplicaciones Web. Jornadas Desarrollo Web (Salamanca, Spain, November 20-27), Ediciones Universidad de Salamanca. 2004.
10. Cortes, J., Becerra, C., & Astudillo, H. Mejoramiento de Descripciones de Objetos de Aprendizaje Basado en Comunidades de Expertos. Primer MoodleMoot Uruguay. 12 al 13 de octubre de 2011 Montevideo, Uruguay.
11. Ruiz Delgado, José. Patrones de Diseño en el e-Learning. Universidad de Alcalá. Disponible en: http://www.slideshare.net/jmruizforem/patrones-de-diseo-en-e-learning. Consultado: 22/04/2012.
12. Julio Alonso Arévalo, Imma Subirats Coll y María Luisa Martínez Conde. Informe APEI sobre acceso abierto. Gijón: Asociación Profesional de Especialistas en Información, 2008. Informe APEI 2 – 2008. D. L.: AS-06633-2008. ISBN: 978-84-691-7725-9
13. Santos-Hermosa, Gema; Ferran-Ferrer, Núria; Abadal, Ernest. Recursos educativos abiertos: repositorios y uso. El profesional de la información, 2012, marzo-abril, v. 21, n. 2, pp. 136-145.
14. Wiley, D. Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. En D. A. Wiley (Ed.): The Instructional Use of Learning Objects: Online Version. Consultado el 31/10/2011 en WWW: http://reusability.org/read/chapters/wiley.doc, http://www.eubca.edu.uy/disenio_de_metadatos_para_descripcion_de_recursos_educativos_digitales.pdf
15. D'Antoni, Susan. "The Virtual University, models and messages. Lessons from case studies". The virtual university and e-learning. Unesco-IIEP, 2006. http://www.unesco.org/iiep/virtualuniversity/home.php
16. Bueno de la Fuente, G. Modelo de Repositorio Institucional de contenido educativo (RICE): la gestión de materiales digitales de docencia y aprendizaje en la biblioteca universitaria. Tesis doctoral. Universidad Carlos III Madrid. Departamento de Bibliotecología y Documentación. Getafe, abril, 2010. http://e-archivo.uc3m.es/handle/10016/9154
17. Confederation of Open Access Repositories - COAR. El caso de Interoperabilidad para Repositorios de Acceso Abierto. Grupo de trabajo 2: Interoperabilidad del Repositorio. Julio 201, versión 1.0. http://coar-repositories.org
18. Unesco. World conference on higher education: the new dynamics of higher education and research for societal change and development. Paris, 5-8 July 2009.
19. Guédon, Jean-Claude. El acceso abierto y la división entre ciencia "principal" y "periférica". Crítica y Emancipación, (6): 135-180, segundo semestre de 2011.
20. San Martín, P; Guarnieri, G, y Bongiovani, P. Propuesta sociotecnológica para el desarrollo de repositorios de acceso abierto adecuados al contexto universitario argentino. e-Ciencias de la Información, Vol. 4, Núm. 2. ISSN electrónico: 1659-4142 (2014). http://revistas.ucr.ac.cr/index.php/eciencias/article/view/15131/14426.
21. Euzenat, J., Shvaiko, P. (2007). Ontology matching. Springer-Verlag Berlín. ISBN 978-3-540-49611-3.
22. Sistema Nacional de Repositorios Digitales. Directrices para proveedores de contenido del Sistema Nacional de Repositorios Digitales del Ministerio de Ciencia, Tecnología e Innovación Productiva. Biblioteca Electrónica de Ciencia y Tecnología. Buenos Aires, Abril de 2012

Uso de un EVA en un contexto de enseñanza presencial

Escudero Silvia Susana[1] Marazzo José Luis[1] Peri Jorge Alberto[1]

[1] Universidad Nacional de Luján- Dpto. de Ciencias Básicas
Rutas 5 y 7 – Luján- Buenos Aires
sescudero@unlu.edu.ar marazzojoseluis@gmail.com jp28580@yahoo.com

Resumen

Los cambios tecnológicos repercuten también en la universidad donde la mayoría de los estudiantes pertenecen a una generación de nativos digitales, para dar respuesta a esta demanda, en este trabajo se describe una experiencia donde se utiliza un entorno virtual de aprendizaje como complemento en el dictado de la asignatura Programación y Métodos Numéricos de la carrera Ingeniería Industrial.
Se muestra una nueva forma de presentar los contenidos y se destaca las ventajas del trabajo colaborativo en la enseñanza de programación.

Palabras Claves: Entorno Virtual de Aprendizaje, trabajo colaborativo.

Introducción

La evolución tecnológica ha generado importantes repercusiones en todas las áreas de la sociedad, y la educación superior no está ajena a este cambio. En la universidad la mayoría de los estudiantes pertenecen, desde el punto de vista digital, a una generación que es denominada de diferente forma según autor que se trate.

Para **Prensky** [1] representan la primera generación que creció con esta tecnología y como resultado de ello no piensan ni procesan la información de la misma manera que sus predecesores, ya que sus patrones de pensamiento han cambiado. A estos «nuevos» estudiantes los denomina ***Nativos Digitales***, hablantes nativos de un lenguaje digital, mientras que al resto los define como ***Inmigrantes Digitales***, personas que aunque puedan llegar a adaptarse y aprender a usar estas tecnologías, no dejan de ser inmigrantes en un mundo digital, manteniendo su «acento» que les diferencia.

Tapscott [2] utiliza el término ***Generación Net*** para denominar a esa generación nacida a principios de los años 80, y por tanto «bañada en bits», caracterizados especialmente por su curiosidad, su capacidad de adaptación y su especial interés por los contextos tecnológicos. Describe, además, en su obra diez importantes características de esta generación, como por ejemplo la independencia y autonomía, la curiosidad e investigación, la suspicacia frente a los intereses corporativos, la firmeza y definición de sus opiniones, la necesidad de inmediatez, entre otras cosas.

En definitiva, y más allá de la posición y denominación que elijamos para definirlos, nuestros estudiantes llegan a la universidad con una cierta alfabetización digital, ya que conocen algunas herramientas TICs y las saben utilizar.

En base a lo expuesto creemos que es necesario modificar las prácticas educativas en las aulas. "*Son las TIC, desde su concepción, diseño y posterior empleo en los procesos de aprendizaje, las que nos ayudan a adecuar la enseñanza a los nuevos escenarios de educación que están apareciendo"* [3].
Para dar respuesta a esta demanda una alternativa son los **Entornos Virtuales de Aprendizaje (EVA)**

Entornos Virtuales de Aprendizaje

Los Entornos Virtuales de Aprendizaje (EVA), son instrumentos de mediación educativa, un recurso que permite obtener un mayor control sobre la circulación de contenidos entre los diferentes agentes que intervienen interactuando en el proceso de aprendizaje [4], que deben haber sido elaborados para un uso intuitivo y sencillo, donde el estudiante sea capaz de encontrar la información que necesita en cada caso de forma rápida y lógica [5].
Hablamos, por tanto, de aplicaciones capaces de transformar la realidad y adecuarla a las necesidades de cada alumno en particular. Se trata de crear espacios que dejan de lado el clásico pupitre, crear ambientes que se abstraen del libro de texto, en definitiva, entornos que se orientan al alumno.

Experiencia

En este trabajo se describe el uso de un entorno virtual en el dictado de la asignatura Programación y Métodos Numéricos, que integra el plan de estudios de la carrera Ingeniería Industrial de la Universidad Nacional de Luján. La experiencia se desarrolló con un grupo de 30 alumnos.
Si bien los entornos virtuales fueron creados para la enseñanza a distancia, aquí se muestra una forma de uso distinta de la tradicional, aplicándola como complemento de la enseñanza presencial, con el propósito de modificar la forma en la que el estudiante aborda el aprendizaje, variando sus estrategias de adquisición de los conocimientos; e iniciándolos en el desarrollo de nuevas formas de trabajo y de organización.
El entorno virtual es usado en cada clase para poner al alcance de los alumnos el material educativo y enriquecerlo con recursos publicados en Internet. También promueve la comunicación fuera de los límites presenciales entre los alumnos**,** y entre estos y el docente. Este sistema permite a los participantes familiarizarse con el uso de las Tecnologías de Información, además da acceso a los materiales de cada clase desde cualquier computador conectado a la red, permitiendo mantener la clase actualizada con las últimas publicaciones de buenas fuentes, pueden compartir puntos de vista con compañeros de clase, y llevar a cabo trabajos en grupo.
El EVA que se utiliza es **e-ducativa,** el mismo es licenciado por la Universidad Nacional de Luján.
El acceso al EVA está limitado sólo a aquellos alumnos que se han inscripto para cursar la asignatura por medio de una clave personal.

El docente es el administrador del espacio encargado de cargar contenidos, actividades o ejercitaciones de la clase, monitorear la asistencia de los alumnos al aula virtual, conocer los resultados de las ejercitaciones y acceder a los archivos de evaluación que el alumno envía al sistema.

Diseño del EVA

El docente además del rol de administrador es el encargado de diseñar el espacio virtual en base a las características del entorno virtual; en el caso de **e-ducativa** la misma está dividida en secciones, el docente puede configurar las mismas de acuerdo a sus preferencias (nombre, color, tipo de letra, orden..).
En esta asignatura se decidió la siguiente configuración:

SECCIONES	CONTENIDOS
GUIA DIDACTICA	Documento de texto independiente del contenido cuya función es informar al estudiante lo necesario sobre el proceso de aprendizaje. La cual está compuesta por el programa de la asignatura, y cronograma de actividades
MATERIALES	Material didáctico, guía de trabajos prácticos
CORREO INTERNO	Medio de comunicación asincrónico entre los alumnos y el docente
INTERCAMBIO	Foro y chat

El material utilizado en las cátedras ha evolucionado y actualmente el soporte principal ya no es el texto o apunte impreso, sino que se adicionan diversos recursos digitales, tales como los provistos por la web 2.0. Además de este cambio en el formato y soporte de los materiales usados en un curso o materia, también ha evolucionado la forma de comunicación e interacción entre los distintos participantes de un curso (docentes, alumnos, tutores, etc.)[6]
Uno de los objetivos de esta experiencia es fomentar el trabajo en equipo, ya que cuando se enseña programación una de las técnicas usadas es la resolución de problemas a través de la computadora. En el curso habitual se plantea un problema, se lo debe interpretar, modelar una solución, seleccionar la estructura de programación más adecuada para solucionarlo, escribir el algoritmo en un lenguaje de programación y luego ejecutar el mismo. Toda esta tarea se realiza en forma individual por parte del alumno.
En este ensayo se privilegió el uso de un elemento importante del EVA: el Foro. En el mismo se propuso a los alumnos resolver, en grupo y con la guía del docente tutor,

ejercicios adicionales fuera del horario de clase. El objetivo de esta actividad fue observar actitudes vinculadas con la inteligencia emocional, como por ejemplo**:**

- Disposición para trabajar en grupo.
- Tolerancia para enfrentarse a situaciones ambiguas.
- Habilidades para la solución de problemas.
- Habilidades de comunicación

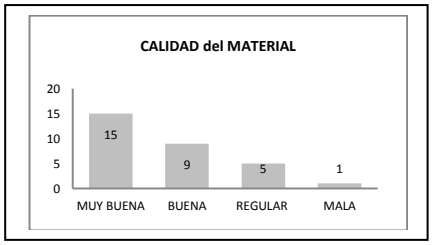
Los alumnos realizaron un trabajo colaborativo en el estudio de un problema buscando soluciones viables, y de esta forma asumieron una mayor participación sobre su aprendizaje.
En este curso se trabajó en el diseño de algoritmos para la resolución de sistemas de ecuaciones lineales. La experiencia indica que algoritmos de este tipo presentan cierto nivel de dificultad a alumnos que están haciendo su primer curso de programación.

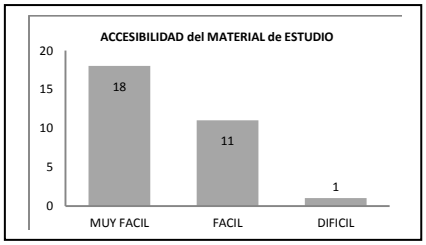
Resultados

Al final de la cursada, a los alumnos se les suministró una encuesta, para recabar su opinión respecto de la misma. En esta primer experiencia, se efectuaron preguntas respecto de dos puntos fundamentales:

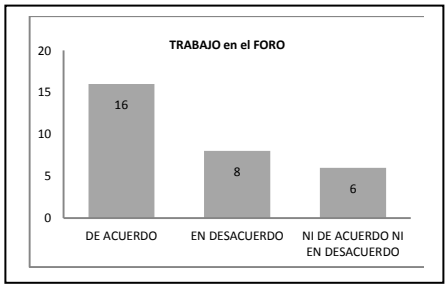
- *Calidad y presentación del material didáctico*
- *Experiencia con el uso del foro*

Con respecto al primer punto, el **95%** de las respuestas valoró positivamente la metodología utilizada.





Con respeto al trabajo del foro, el 85% destacó como muy favorable está modalidad y además consideraron que resultó una buena metodología de trabajo grupal que los ayudó a prepararse para los exámenes.



En las preguntas abiertas manifestaron su satisfacción de disponer en un solo lugar todo el material de la asignatura, y la posibilidad de contar con el mismo aún cuando por alguna causa no hubieran podido asistir a las clases presenciales. Destacaron también la ventaja de poder trabajar en equipo sin necesidad de encontrarse físicamente.

Desde el punto de vista de los docentes, se percibió que fue posible un mejor seguimiento del proceso de aprendizaje y facilitó la realización de los ajustes necesarios para mejorar el mismo.

Trabajo futuro

Se considera que los resultados de esta primer experiencia, modesta en su origen, ameritan la continuidad mediante ensayos más ambiciosos en la misma línea. Además de la confirmación y medición más precisa de las mejoras observadas en esta experiencia en el proceso de aprendizaje, resultaría particularmente interesante la medición de resultados en aspectos vinculados con las actitudes sociales durante los trabajos en equipo a través del foro. Para ello se requerirá el desarrollo de los correspondientes instrumentos de medición, incluido el perfeccionamiento y ampliación de la encuesta de fin de curso.

Conclusiones

En este experimento se trabajó con una forma de aprendizaje, en donde se generan espacios virtuales que facilitan interacciones sociales entre los participantes de estos procesos educativos, independientemente del tiempo y lugar geográfico donde se encuentren. En este caso particular, el EVA se utilizó en el contexto de la enseñanza presencial, y como resultado se observó una interesante fecundación cruzada entre ambos mundos: el uso del EVA potenció el trabajo de los docentes, y por otro lado la presencia física de estos últimos mejoró los resultados de la herramienta informática.

Se observó en la práctica como el Aula Virtual promueve una mayor producción académica e intelectual**,** al establecer un espacio donde los productos de su esfuerzo podrán ser consultados y compartidos.

En un contexto más amplio, se percibe como la evolución tecnológica ha generado importantes repercusiones en todas las áreas de la sociedad, en educación superior no sólo ha conllevado cambios en las instituciones educativas sino que también ha influido en las características del actual estudiante. El reto que debemos asumir desde las universidades es desarrollar estrategias de aprendizaje para esta nueva generación de estudiantes.

Bibliografía

1. Prensky. M. (2001). Digital natives, digital immigrants. En *The Horizon*,.

2. Tapscott, D. (1998). Growing up digital: The rise of the net generation. New York: McGraw-Hill.

3. Mondéjar, J., Mondéjar, J. A. & Vargas, M. (2006). Implantación de la metodología elearning en la docencia universitaria: una experiencia a través del proyecto Campus Virtual. Revista Latinoamericana de Tecnología Educativa.

4. Lara, P., Saigí, F. & Duart, J. M. (2003). Gestión de Información en el Diseño de Contenidos Educativos On-Line. Revista Iberoamericana de Ciencia, Tecnología, Sociedad e Innovación, 6

5. Hassan et al. (2004). Arquitectura de la información en los entornos virtuales de aprendizaje. Aplicación de la técnica Card Sorting y análisis cuantitativo de los resultados. El profesional de la información, 13(2), 93-99

6. Fracchia, C. Plaza, J. Análisis de los materiales educativos incorporados a la plataforma PEDCO. RUEDA. V Seminario Internacional. (2010). Universidad Nacional Del Centro De La Provincia De Buenos Aires .Tandil, Bs. As., Argentina.

Plan de evaluación del material educativo digital EPRA. Propuesta de indagación sobre la motivación intrínseca.

Natali Salazar Mesia[1,2], Gladys Gorga[2], Cecilia Sanz[2].

[1] Beca Tipo A – Facultad de Informática – Universidad Nacinal de La Plata
[2]Instituto de Investigación en Informática LIDI. Facultad de Informática – Universidad Nacional de La Plata
{nsalazar, ggorga, csanz}@lidi.info.unlp.edu.ar

Resumen. En este artículo se describe un plan de evaluación para el material educativo digital EPRA (Enseñando Programación con Realidad Aumentada). EPRA incluye el uso de realidad aumentada para el desarrollo de una serie de actividades con diferentes intenciones didácticas para la enseñanza de conceptos básicos Programación. El plan de evaluación se orienta a analizar las dimensiones de satisfacción, impacto en el aprendizaje y motivación. Se presentan resultados preliminares en relación a las primeras dos dimensiones. Se describe la propuesta específica para la evaluación de la dimensión de motivación, a partir del uso del instrumento: *Intrinsic Motivation Inventory*, y finalmente, se presentan las conclusiones y trabajos futuros.

Palabras Clave: Realidad Aumentada, Enseñanza de programación, Material educativo digital, Motivación, Evaluación.

1 Introducción

La realidad aumentada es una tecnología que complementa la percepción e interacción con el mundo real y permite a la persona vivenciar un entorno real aumentado, con información adicional generada por el ordenador. Posibilita el desarrollo de aplicaciones interactivas que combinan la realidad con información sintética, tal como imágenes 3D, sonidos, videos, textos, sensaciones táctiles, en tiempo real, y de acuerdo al punto de vista de quien está observando la escena.

La información virtual, tiene que estar vinculada especialmente al mundo real, es decir, un objeto virtual, siempre debe aparecer en cierta ubicación relativa al objeto real. La visualización de la escena aumentada (mundo real + sintético) debe hacerse de manera coherente [1] [2].

La enseñanza en el ámbito universitario ha ido evolucionando con la incorporación de diversas tecnologías. A través de las herramientas informáticas tanto los docentes como los alumnos pueden desarrollar una mayor vinculación y motivación para favorecer los procesos de enseñar y aprender. Un ejemplo de esto lo constituye el material educativo EPRA (Enseñando Programación con Realidad Aumentada), que

utiliza realidad aumentada para el desarrollo de una serie de actividades con diferentes intenciones didácticas para la enseñanza de Programación.

Diversos autores indican que la motivación que se genera en los estudiantes y docentes tiene diferentes enfoques de análisis en cuanto a su significación educativa. Señalan que las personas pueden estar motivadas debido a que valoran una actividad o debido a que hay una fuerte presión externa por parte de otras personas como docentes, padres, etc. Existen diferentes técnicas e instrumentos para medir la motivación en los estudiantes y se han realizado diferentes estudios que se analizarán en la siguiente sección [3].

En este artículo se propone la utilización del instrumento *Intrinsic Motivation Inventory* (IMI). Deci describió este instrumento en sus inicios, tomando en cuenta la motivación humana como factor de cambio donde describe tres tipos de motivaciones que pueden presentarse intrínseca, extrínseca y afectiva. IMI ha sido validado por diversos autores tales como Tsigilis y Theodosiou [4] [5].

Este instrumento permite realizar una valoración de interés/disfrute, competencia percibida, esfuerzo, valor/utilidad, sensación de presión y tensión y percepción de elección durante el desarrollo de una actividad, produciendo seis sub-escalas. En particular, la primer sub-escala de interés/disfrute es considerada la medida de motivación intrínseca [6].

IMI ha sido utilizado para medir la motivación en diferentes experiencias de laboratorio y en [6] se describen los diferentes cuestionarios abordados y la forma de calcular los resultados. En este trabajo se retoman estas ideas y se las incorpora al plan de evaluación de EPRA, en donde uno de los ejes es analizar la motivación de los alumnos en el uso de este material educativo.

2 Antecedentes sobre la motivación

La motivación ha sido uno de los conceptos más importantes en el contexto educativo. Se han realizado numerosas evaluaciones para medir la motivación de los estudiantes en determinadas áreas según varios autores [7] [8].

Muchas investigaciones han demostrado que la motivación está relacionada con diversos aspectos como la persistencia, el aprendizaje y el nivel de ejecución. Una de las perspectivas que ha tratado de estudiar la motivación en la educación es la teoría de la autodeterminación que sugiere que la motivación no puede ser entendida desde un punto de vista unidimensional y lo propone desde tres puntos de interés: intrínseca, extrínseca o afectiva que se sitúan en un continuo que va desde la autodeterminación hasta la falta de control [9] [10].

Hay algunos autores que trabajan la motivación autónoma como la formada por la motivación intrínseca y la regulación identificada [11] [12].

El concepto de motivación que se pretende abordar para la evaluación propuesta en este artículo está vinculado con la motivación intrínseca.

Existen trabajos que demuestran una vinculación positiva entre la motivación y los alumnos y esto se observa tanto desde la perspectiva del docente como del alumno. Un ejemplo de este estudio es el que realiza Bieg, Backes, y Mittag [13].

Entre otros antecedentes analizados para medir la motivación, se pueden mencionar:

- La validación de la versión española de "*Échelle de Motivation en Éducation*" a la Escala de Motivación Académica [14] [15].
- El cuestionario "*Motivated Strategies Learning Questionnaire*" [16].
- El estudio basado en el proverbio "*Tell me and I forget, show me and I remember, involve me and I understand*" donde se describe otra forma de enseñanza relacionada a la motivación [17].
- En [18] se describe la utilización de diferentes cuestionarios para medir la motivación. En este estudio se afirma que la elección depende del objetivo que se propone. Entre ellos se pueden mencionar: Cuestionario MAE, MAPE-I, MAPE-II, MAPE-III, MAPEX, AM-1.

Las diversas evaluaciones aquí mencionadas miden diferentes tipos de motivación. En el caso de la propuesta de evaluación para el material EPRA sólo se considerará la motivación intrínseca, la cual indica que el alumno está motivado por la vivencia del proceso, más que por los logros o resultados del mismo [19].

Así, esta propuesta pretende medir la motivación intrínseca de los alumnos en relación al desarrollo de las actividades de RA de EPRA.

3 EPRA (Enseñando Programación con Realidad Aumentada)

En EPRA se incluyen aspectos de realidad aumentada como una estrategia complementaria que puede ayudar al alumno a vivenciar conceptos abstractos que se abordan en el primer año de la carrera. Al mismo tiempo, se busca incentivar la motivación del alumno, y atender a la diversidad de estilos cognitivos, a partir de la utilización de diversos materiales de estudio y actividades. EPRA persigue que el alumno se involucre con nuevos conceptos desde un ámbito más cercano a sus conocimientos previos y a su cotidianeidad a partir del carácter lúdico que propone.

En particular, se aborda la presentación de las estructuras de control iterativas y de selección. Se plantea una serie de actividades vinculadas para que el alumno logre:

- Vivenciar y experimentar el comportamiento de las estructuras de control a partir de situaciones reales y lúdicas.
- Comparar el comportamiento de las estructuras de control, a partir de la visualización inmediata de los efectos que tiene elegir una u otra estructura de control para resolver un determinado problema.
- Decidir qué estructura de control elegir con el fin de obtener un determinado efecto en una escena real, a partir de la inclusión de objetos virtuales.

Así se diseña un sitio web que incluye contenido referencial teórico y actividades basadas en RA para la enseñanza y aprendizaje de Conceptos Básicos de Programación, en particular, para la enseñanza y aprendizaje de las estructuras de control básicas. A continuación se describen algunos aspectos de EPRA.

Presentación general de EPRA. Es un material educativo digital bajo el formato de un sitio web llamado "Enseñando programación básica con Realidad Aumentada". Es accesible desde la web a través de la url: http://163.10.22.174**.**

El material educativo se encuentra organizado en diferentes secciones contenidas en el menú principal, ubicado debajo del encabezado y que se muestra en la figura 1.

Fig. 1. Imagen del sitio web donde se contiene este material educativo. Donde se observa todo su contenido en la parte del menú bajo el encabezado.

Actividades de RA. EPRA plantea el desarrollo de tres tipos de actividades de RA que tienen diferentes objetivos: vivenciar nuevos conceptos, reforzar la comprensión de los temas involucrados, comparar y decidir acerca del uso de las estructuras de control en problemas concretos. Los tipos de actividades diseñadas en cuanto a su función didáctica son: Actividades de Exploración, Actividades de Repaso, y Actividad Integradora.
Para el desarrollo de estas actividades se utilizan marcadores (disponibles dentro de EPRA) que funcionan como condiciones dentro de las estructuras de control a trabajar o como valores finales de la variable índice de la repetición.

Actividades de Exploración. EPRA presenta cuatro actividades de exploración: una para vivenciar la estructura de control de decisión, otra para las estructuras de control de iteración precondicional y postcondicional y otra para la estructura de control de repetición.

Fig. 2. Imagen de las actividades de exploración.

Actividades de Repaso. EPRA contiene tres actividades de repaso para la comparación de las estructuras de control: IF vs WHILE, REPEAT vs WHILE y FOR vs WHILE. Se plantea una consigna que el alumno debe resolver con la elección de una u otra estructura de control.

Fig. 3. Imagen de las diferentes actividades de repaso.

Actividad de Integración. Finalmente la actividad de integración tiene como objetivo que el alumno aplique los conocimientos abordados en las actividades de exploración y repaso. Esta actividad se divide en dos partes. En la primera parte el alumno debe responder tres preguntas sobre aspectos de su persona. La segunda parte se divide en tres etapas. La corrección de cada etapa está relacionada con las respuestas dadas en la primera parte.

Fig. 4. Imagen de la actividad de integración en las diferentes etapas.

4 Plan de Evaluación de EPRA

Se presenta un plan de evaluación para la herramienta EPRA que contempla tres dimensiones:

- Impacto en el aprendizaje de los alumnos
- Satisfacción de los docentes y alumnos
- Motivación de los alumnos

El plan de evaluación establece los objetivos propuestos para cada dimensión, los destinatarios correspondientes y la metodología empleada en cada caso.

1. **Objetivos**

   Al evaluar el impacto en el aprendizaje se pretende medir si el alumno ha consolidado el concepto de estructura de control a partir de la utilización del material EPRA para reforzar los conceptos trabajados en clase y a la vez explorar un material de estudio diferente.

   Al evaluar la satisfacción se persigue medir cuán satisfactorio resultan, para alumnos y docentes, los contenidos teóricos y las actividades de RA de EPRA, la utilidad del material, fortalezas y debilidades encontradas.

   Finalmente, con respecto a la motivación se pretende medir la motivación intrínseca de los alumnos en relación al desarrollo de las actividades de RA.

2. **Destinatarios**
   Este plan de evaluación se lleva a cabo con docentes y alumnos de las cátedras de Conceptos de Algoritmos, Datos y Programas, Programación 1, Programación 2 y del módulo Expresión de Problemas y Algoritmos del curso de Ingreso de la Facultad de Informática – UNLP.
3. **Metodologías**
   Este plan de evaluación utiliza diferentes técnicas e instrumentos.
   La recolección de datos se realiza a partir de observación participante, registro por filmación y juicio de expertos [20]. En el caso de juicio de expertos, los docentes actúan como jueces de EPRA y cada uno brinda su perspectiva tanto en una entrevista como en la realización de una encuesta. Se entiende que su opinión es avalada por los perfiles docentes que tiene cada uno en esta área.
   Para medir el impacto en el aprendizaje se trabaja con un pre-test y un post-test. Con el pre-test se busca analizar el conocimiento previo de los alumnos en relación al tema y con el post-test (efectuado en forma posterior a la utilización de EPRA), se busca analizar el conocimiento del alumno, luego de realizar las actividades propuestas, y consiste en aplicar los conceptos vistos en diferentes situaciones problemáticas que se presentan.
   Para medir la satisfacción se proponen encuestas dirigidas a docentes y alumnos que permiten obtener algunos resultados en relación a diferentes características del material educativo presentado tales como contenido, usabilidad y disponibilidad. Además, se indaga sobre aspectos fuertes y débiles del material en las entrevistas a los docentes.

Finalmente, para medir la motivación se incorpora a este plan de evaluación la utilización de una adaptación del instrumento IMI ya mencionado.

5 Resultados Preliminares

A continuación se detallan algunos resultados preliminares del plan de evaluación presentado que aborda solamente las dimensiones de impacto en el aprendizaje y satisfacción.

Durante la ejecución de este plan de evaluación se llevó un registro de las dificultades encontradas en el desarrollo de cada actividad, los pasos dados para llevar a cabo cada una de las actividades y los aportes planteados por cada participante en relación a la comprensión de las consignas y actividades.

Resultados en relación al impacto en el aprendizaje. Los resultados obtenidos a partir del pre-test y pos-test se reflejan en la figura 6. Se trabajó al momento sobre un total de 11 alumnos.

Preguntas	Pre Test		Post Test	
IF	Bien 60%	Mal 40%	Bien 100%	
WHILE	Bien 75%	Mal 25%	Bien 100%	
FOR	Bien 70%	Mal 30%	Bien 90%	Mal 10%

Fig. 6. Extracto de resultados del pre-test y post-test

Se observa que los alumnos tuvieron varios errores en la evaluación del pre-test de acuerdo a los porcentajes. Sin embargo, los resultados obtenidos en la evaluación del post-test fueron más favorables. Esto constituye sólo un indicio, ya que la muestra es pequeña. Durante este semestre se trabajará con un número mayor de alumnos.

Resultados en relación a la satisfacción de docentes y alumnos. Algunas de las respuestas de las encuestas de docentes y alumnos presentan una valoración de 1 a 5 que indica cuán satisfactoria ha sido cada uno de los tipos de actividades realizadas. Se observa que la clasificación obtiene una valoración mayor o igual a 3 y que en su mayoría las puntuaciones fueron altas tanto en docentes como en alumnos. En la figura 7 se muestra un extracto de los resultados obtenidos, a partir del trabajo con 9 docentes y 11 alumnos.

	Actividades de Exploración		Actividades de Repaso		Actividades de integración	
	Alumno	Docente	Alumno	Docente	Alumno	Docente
3	12,5%	27,27%	25%		25%	27,27%
4	50%	36,36%	25%	81,81%	12,5%	45,45%
5	37,5%	36,36%	50%	18,18%	62,5%	27,27%

Fig. 7. Extracto de resultados obtenidos en las encuestas para docentes y alumnos respecto a la satisfacción de cada tipo de actividad.

Resultado del juicio de expertos. Por otro lado, en el juicio de expertos, uno de los conceptos que se destaca en la evaluación de esta herramienta es la concordancia entre los jueces. Esto se evidencia al calcular el porcentaje de acuerdo. En este caso, es importante destacar que todos los docentes (jueces) incluyeron la motivación como principal aporte de EPRA.

La descripción detallada de las actividades de RA en EPRA presentada en la sección 3 y los resultados obtenidos de las dimensiones evaluadas que se mencionan en esta sección, se encuentran de manera completa en [21] [22].

6 Propuesta de Evaluación de la motivación en EPRA

Para completar el plan de evaluación propuesto en la sección 4, se agrega la evaluación de la dimensión Motivación que describe un conjunto de aspectos que permiten medir la motivación intrínseca de los alumnos. Para ello se utiliza el instrumento IMI ya presentado.

En particular se considera el cuestionario IMI de 22 ítems y cuatro sub-escalas: 1. interés/disfrute, 2. competencia percibida, 3. elección percibida y 4. presión/tensión.

Cada uno de los ítems del cuestionario se evalúa en la escala del 1 al 7 cuya valoración es como se muestra en la figura 8.

1	2	3	4	5	6	7
Nada Cierto			Algo Cierto			Muy Cierto

Fig. 8. Escala de valores para cada ítem del cuestionario propuesto por IMI.

Los ítems están numerados y se clasifican en las sub-escalas mencionadas como se observa en la figura 9. Al momento de realizar la evaluación, se presentan al alumno los 22 ítems, sin clasificación en sub-escalas.

Interés/Disfrute	Competencia Percibida	Elección Percibida	Presión/Tensión
1. Mientras realizaba la actividad pensaba cómo lo disfrutaba	**4.** Creo que soy bastante bueno en la actividad	**3.** Sentí que yo elegí hacer la actividad	**2.** No sentí ninguna presión cuando hice la actividad
5. La actividad me pareció muy interesante	**7.** Si me comparo con otros compañeros, creo que hice muy bien la actividad	**11.** Tuve que hacer la actividad	**6.** Sentí tensión al hacer la actividad
8. Me divirtió hacer la actividad	**12.** Estoy satisfecho en mi desempeño al hacer la actividad	**15.** Sentí que podía hacer lo que quería al hacer la actividad	**9.** Me sentí distendido al hacer la actividad
10. Disfrute mucho hacer la actividad	**16.** Me sentí muy hábil al hacer la actividad	**19.** Sentí que tuve que hacer la actividad	**13.** Me sentí ansioso al hacer la actividad
14. Considere la actividad aburrida	**22.** Después de realizar la actividad durante un rato, me sentí muy competente	**21.** Hice la actividad porque no tuve opción	**18.** Me sentí presionado al hacer la actividad
17. Considere la actividad muy interesante			
20. Describiría la actividad como muy entretenida			

Fig. 9. Ítems clasificados en sub-escalas del cuestionario IMI.

Para evaluar los resultados de cada sub-escala se suman los puntajes correspondientes a cada ítem asociado a esa sub-escala. En el caso del cálculo de la sub-escala de interés/disfrute el puntaje del ítem 8: *me divirtió hacer la actividad* se obtiene restando los ítems 2, 9, 11, 14, 19 y 21 (Ver Fig. 9). Estos son, por ejemplo: *Considere la actividad aburrida* o *Hice la actividad porque no tuve opción*.

Un puntaje alto en la sub-escala tensión/presión, por ejemplo, indica que el alumno sintió más presión o tensión en la realización de la actividad resultando un factor predictivo negativo respecto a la motivación, mientras que puntajes altos en la sub-escala de competencias percibidas indica que el alumno se sintió más competente, proporcionando un factor predictivo positivo. Dado que EPRA trabaja con tres tipos de actividades con diferentes intenciones didácticas, se propone realizar una evaluación de la motivación considerando una subclasificación específica que permita vincularla a los distintos tipos de actividades.

Cada uno de los 22 ítems de IMI se subclasifican para la actividad de exploración, de repaso y de integración. El resultado de cada ítem se evalúa por separado para cada actividad respetando las sub-escalas y el cálculo que expone IMI.

En el caso de puntajes altos en las sub-escalas de interés/disfrute, competencia percibida y elección percibida y puntaje bajo en la sub-escala de presión/tensión reflejan que la herramienta tiene un nivel de motivación satisfactorio. Cualquier otra combinación de puntajes lleva a un análisis más detallado para introducir mejoras en el material relacionados con la sub-escala determinada.

Por ejemplo, un puntaje bajo en la sub-escala interés/disfrute lleva a reconsiderar algunos de los aspectos de visualización y contenidos abordados en la encuesta, entre otros.

7 Conclusiones y trabajos futuros

Se plantea un plan de evaluación que contempla tres dimensiones. De los resultados de la evaluación de las dimensiones de impacto en el aprendizaje y satisfacción, se observa que la motivación es un aporte visualizado tanto por docentes como por alumnos en relación a EPRA. Esto lleva a completar el plan de evaluación incorporando un análisis más formal que atienda otros aspectos específicamente vinculados a la motivación.

Se propone incorporar una adaptación del instrumento IMI para medir la dimensión motivación del plan de evaluación de EPRA, que contempla 22 ítems clasificados en las cuatro sub-escalas: interés/disfrute, competencia percibida, elección percibida y presión/tensión.

Con este instrumento de evaluación se pretende obtener una valoración de la motivación intrínseca que poseen los alumnos cuando se disponen a trabajar con el material educativo EPRA. Una vez que el plan de evaluación se ejecute con una muestra significativa se propone introducir las mejoras pertinentes en el material EPRA en cuanto a su aplicación como recurso complementario en la enseñanza y aprendizaje de los conceptos básicos de la Programación. En particular, para la dimensión motivación se tendrán en cuenta los puntajes alcanzados en cada una de las sub-escalas y se tomarán las decisiones correspondientes en cada caso.

Se considera que el plan de evaluación presentado es un aporte para quienes se propongan evaluar un material educativo digital, y que incluya como objetivo la motivación, aspecto fundamental del aprendizaje.

8 Referencias bibliográficas

1. Milgram Kishino, P.; Takemura, H.; Utsumi, A.; Kishino, F. Augmented Reality: A class of displays on the reality virtuality continuum. Telemanipulator and Telepresence Technologies, 2351, 282-292 (1994)
2. Azuma R. Augmented Reality: Approaches and Technical Challenges, Fundamentals of Wearable Computers and Augmented Reality, W. Barfield, Th. Caudell (eds.), Mahwah, New Jersey, 2001, 27-63 (2001)
3. Ryan, E., Deci, E. La Teoría de la Autodeterminación y la Facilitación de la Motivación Intrínseca, el Desarrollo Social, y el Bienestar. Universidad de Rochester. Revista American Psychologist. Vol. 55, No. 1, 68-78. (2000)
4. Tsigilis, N., Theodosiou A. Temporal stability of the intrinsic Motivation Inventory. Percept Mot Skills. 2003 Aug; 97(1):271-80 (2003)
5. Deci, E. L., Ryan, R. M. Intrinsic motivation and self-determination in human behavior. New York: Plenum (1985)
6. IMI (Intrinsic Motivation Inventory). Recuperado en el 10 de julio de 2015: http://www.selfdeterminationtheory.org/
7. Rodríguez, S., Cabanach, R. G., Piñeiro, I., Valle, A., Núñez, J. C., González-Pienda, J. A. Metas de aproximación, metas de evitación y múltiples metas académicas. Psicothema, 13, 546-550 (2001)
8. De la Torre, C., Godoy, A. Diferencias individuales en las atribuciones causales de los docentes y su influencia en el componente afectivo. Revista Interamericana de Psicología, 38, 217-224 (2004)

9. Balaguer, I., Castillo, I., Duda, J. L. Apoyo a la autonomía, satisfacción de las necesidades, motivación y bienestar en deportistas de competición: un análisis de la teoría de la autodeterminación. Revista de Psicología del Deporte, 17(1) 123-139. Recuperado de http://www.redalyc.org/articulo.oa?id=235119246002 (2008)
10. Deci, E. L., Ryan, R. M. (1991). A motivational approach to self: Integration in personality.In R. Dienstbier (Ed.), Nebraska symposium on motivation: Vol. 38, Perspectives on motivation (pp. 237-288). Lincoln: University of Nebraska Press (1991)
11. Katz, I., Assor, A., & Kanat-Maymon, Y. A projective assessment of autonomous motivation in children: Correlational and experimental evidence. Motivation and Emotion, 32, 109-119 (2008)
12. Deci, E. L., Ryan, R. M. Self-determination theory. En A. W. Kruglanski, P. A. M. Van Lange y E. T. Higgins (Eds.), Handbook of theories social psychology (Vol. 1, pp. 416-437). London: SAGE (2012)
13. Bieg, S., Backes, S., Mittag, W. The role of intrinsic motivation for teaching, teachers' care and autonomy support in students' self-determined motivation. Journal for Educational Research Online, 3(1) (2011)
14. Moreno-Murcia, J. A., Silveira Torregrosa, Y. Hacia una mejor predicción de la percepción de competencia laboral en los universitarios. Universidad Miguel Hernández de Elche (España). Revista de docencia universitaria vol. 13 (1), Enero-Abril 2015, 277-292. ISSN: 1887-4592 (2015)
15. Núñez. J. L., Albo Lucas, J. M., Navarro Izquierdo, J.G. Validación de la Escala de Motivación Educativa (EME) en Paraguay. Universidad de Las Palmas de Gran Canaria, España. Revista Interamericana de Psicología/Interamerican Journal of Psychology - 2006, Vol. 40, Num. 3 pp. 000-000 (2006)
16. Donolo, D., Chiecher, A., Rinaudo, M.C. Estudiantes en entornos tradicionales y a distancia. Perfiles motivacionales y percepciones del contexto. RED, Revista de Educación a Distancia, 10 (2004)
17. Pollock, M. Basic Mechanics: Learning by Teaching – an increase in student motivation (a small scale study with Technology Education students). 35th ASEE/IEEE Frontiers in Education Conference T2F-12.Octubre 19 – 22, 2005, Indianapolis, IN. 0-7803-9077-6/05(2005)
18. Alonso Tapia, J. Evaluación de la motivación en entornos educativos. En Álvarez, M. y Bisquerra, R. (Eds), Manual de orientación y tutoría (Libro Electrónico) pág. 1-39. Ed. Wolters-Praxis. Barcelona (2007)
19. Polanco Hernández, A. La motivación en los estudiantes universitarios. Revista Electrónica "Actualidades Investigativas en Educación", 5(2) 1-13 (2005)
20. Maldonado Mahauad, J. Desarrollo de un Marco de Análisis para la Selección de Metodologías de Diseño de Objetos de Aprendizaje (OA) basado en criterios de calidad para contextos educativos específicos. Tesis presentada para obtener el grado de Magíster en Tecnología Informática Aplicada en Educación. Directores Sanz, C., Fernández-Pampillón. A. Universidad Nacional de La Plata. (2015)
21. Salazar Mesía, N., Gorga, G., Sanz, C. EPRA: Herramienta para la Enseñanza de conceptos básicos de programación utilizando realidad aumentada. X Congreso de Tecnología en educación y Educación en Tecnología. ISBN 978-950-656-154-3 (2015)
22. Salazar Mesía, N., Gorga, G., Sanz, C. EPRA: Herramienta para la Enseñanza de conceptos básicos de programación utilizando realidad aumentada. XXIII Jornada jóvenes investigadores Grupo Montevideo. Universidad Nacional de La Plata. (2015)

Application of Virtual Reality in a Learning Experience

Victor Obrist and Eustaquio Martnez

Polytechnic School, National University of the East
Ciudad del Este, Paraguay
{victor.obrist,ealcidesmartinez}@gmail.com

Abstract. The project is an application that allows users to interact in a virtual environment via web interface, in which models are in three dimensions to simulate different activities. The application focuses on education through the use of virtual reality technology, enriching the student's perception through the interaction with objects in an artificial world, facilitating their learning. For this purpose a website was built on which the application can be executed directly in the web browser with the help of a plugin or downloaded and run as a native application. The user navigates in the virtual environment containing a three-dimensional replica of one of the laboratories of the Polytechnic School U.N.E. The tests consisted of running the model and laboratory simulations. The results obtained from forms show that the use of virtual reality is valid, accepted and helps understanding the context of the simulations.

Keywords: Virtual Reality, Simulation, Computer Graphics.

1 Introduction

New information and communication technologies are emerging with the modern times, one of them is the Virtual Reality, which is widely used in areas of research, simulation, among others that deal with the use of computers for displaying information and give users new ways to make the most of it.

Using this advantage, Virtual Reality is gaining ground in education, getting students to interact and even manipulate information, having rich teaching and enabling the understanding of any area with ease, even the most technical.

1.1 Problematic

Most universities and vocational training centers in developing countries do not have necessary equipment for the education of future professionals, which with luck will have few opportunities to see and operate cutting-edge equipments in the workplace. One option would be to get donations and assemble laboratories to house the equipment, which would generate infrastructure and security costs, plus the possibility that they may suffer some kind of damage. Another option could be to create a "virtual laboratory", which could simulate many types of

laboratories according to the needs, all in one location, with the assurance that if an equipment is damaged in the virtual laboratory, it can be replaced or fixed at the moment of the event, requiring only to restart that specific simulation.

A divergence of this kind of laboratory would use the Internet as a means of dissemination of the class, since having a location for simulation is being tied to schedules, leading the student to have to wait when is available for use after school's hours.

The implementation of the application using Internet solve this problem, being able to have classes even at homes, students attend the simulation with the "virtual" presence of the teacher, and if they have questions outside class time, could access it to better investigate, without having to wait for use. With the creation of these simulation and studies centers, teaching of abstract matters would be benefit, since the manipulation of parameters, changes involving them and their importance in the studied model could be perceived in a better way. Both students and teachers have at their disposal a powerful tool in education, particularly in vocational subjects in technical field.

2 Virtual Reality

Defining Virtual Reality (or simply VR), is difficult. There are probably as many definitions as there are researchers, as its recent and rapid evolution has not allowed a clear definition. Thus it not is surprising that the virtual reality turns out to be relative to different people and in different situations.

Some definitions to follow:

- **Virtual reality** is the experience of telepresence, where telepresence is the sense of presence using a means of communication.
- **Virtual reality** is a way by which human visualize, manipulate and interact with computers and extremely complex data.
- **Virtual reality** is three-dimensional interactive environments and simulations that reproduce real situations.
- **Virtual reality** is a highly interactive environment where users participate through the use of a computer in a virtually real world. It is a three-dimensional computer simulation in which the user is immersed so completely that this reality, artificial origin, appears to be real.

Virtual reality is an interactive three-dimensional computer simulation in which the user feels inserted into an artificial environment, and perceived as real based on incentives over sensory organs [1].

That is why the term "Virtual Reality" is usually associated with almost everything that has to do with three-dimensional images generated by computer and the user interaction with the graphical environment. This implies the existence of a complex electronic system to project 3D visual space and to send and receive signals with information about the user's actions, who, with a system of this kind, feel like is immersed in a "virtual world".

The aim of Virtual Reality is to create an experience that makes the user feel that is immersed in a seemingly actual virtual world; to do this, it uses 3D graphics and sound that surrounds the scenes shown. The user acts like an observer, who moves through the virtual world using suitable devices, such as glasses or electronic gloves.

Virtual Reality exploits every image reproduction techniques and extends them, using them in the environment which the user can examine, manipulate and interact with the exhibits. A virtual world is a mathematical model that describes a "three-dimensional space", within this "space" are contained objects that can represent anything from a simple geometric entity, such as a cube or a sphere, to a complex shape, such as it can be an architectural development, a new physical state of matter or the model of a genetic structure. It is ultimately a step beyond what would be the computer simulation, case really interactive simulation, dynamic and real-time system [2].

3 Virtual Reality in Education

The techniques of virtual reality (multisensory digital simulation) appear in the eyes of many experts as the definitive means of computer input in the processes of education and training. In this sense, education is one of the most promising areas of social use for the dissemination of this emerging medium of communication and digital simulation, which can be considered a perfected form of multimedia.

Interactive multi-sensory techniques such as virtual reality, offer extraordinary possibilities in this regard. In fact, virtual reality coupled with advanced telecommunications network, allows to imagine a teaching environment in which it is possible to experience the presence of the teacher and other fellow students and exchange opinions and materials with them as if we were together without none of the participants have to move from where it previously found at the beginning of the class. In this new context, the traditional role of the teacher changes, no longer a mere transmitter of knowledge more or less valid, to become the instructor about students learning with the help of technology, which is what provides interactive resources learning [3].

Techniques related to virtual reality are very suitable for training in all disciplines requiring skills, because they facilitate the realization of practices in all situations (including, especially, those that may be hazardous in the physical world). A basic development architecture of almost unlimited variety of virtual labs was developed. In them, scientists of many different disciplines are able to penetrate into previously unreachable horizons thanks to the possibility of being there: within a molecule, in the midst of a violent storm or in a distant galaxy [4].

There are several applications that use virtual reality as a tool. Within the field of education, they are from virtual labs to virtual classrooms for behavioral studies. Here are two projects using virtual reality in some way or another.

3.1 AULA Nesplora

AULA is an assessment test that uses virtual reality to help diagnose the disorder Attention Deficit / Hyperactivity Disorder (ADHD) [5]. The AULA system analyzes the behavior of the child in a virtual school class. The test is initially perceived as a game, where you must perform a different task while typical distractions of a classroom are presented.

3.2 Labster. Virtual Biological Laboratory

Labster [6] is an international company dedicated to developing online tools for teaching science globally. Its main product is an easily scalable platform for online teaching of life sciences, which has been shown to significantly increase learning over traditional methods as well as substantially reduce costs [6].

The Labster platform is a 3D virtual learning environment based on a Virtual Laboratory. It Includes 3D animations molecules, questionnaires and support theory that invites the student to an immersive multimedia experience.

4 Method

4.1 Architecture

This work proposed, the development and implementation of an application that uses virtual reality technology as a support tool in teaching using a web interface. To achieve this objective a website with the following structure was created:

- **Main Page:** Contains a welcome and brief introduction about the project. Moreover there are available the download version of the application for different operating systems (Windows, Linux and Mac) the link to the application and a list of contact.
- **Application:** A page where the application of virtual reality lies with instructions to use it.
- **Simulation forms:** Contains questions about the simulations in the application and the application itself. Used to collect data from usability, performance and compliance objectives.
- **SQLite database:** Stores data entered into forms for further processing.

In order to provide portability, the application has an online mode, which uses a supported Web browser with the Unity WebPlayer plugin. Both application and forms run on the user's Web browser where an internet connection is necessary (see Fig. 1).

It also has an offline mode, in which the application download is needed according to the operating system and architecture. Once downloaded and unzipped, it runs as a native application on the operating system, using Internet only for fill the forms (see Fig. 2).

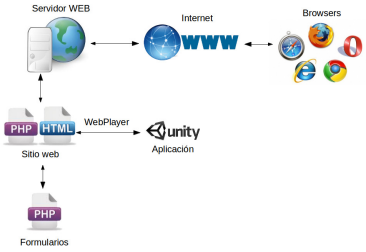


Fig. 1. Online Architecture

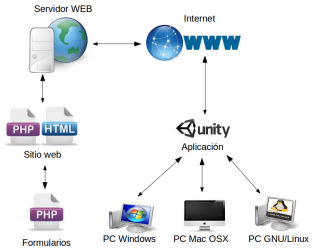


Fig. 2. Offline Architecture

The application development started with 3D modeling of the laboratory building based on photos using Blender 3D, once the modeling process was completed, began the texturing of the photos taken as a reference, making the treatment and adjusting of the them to add to the model. The processing task and adjustment was conducted with the GIMP software. Finally the textured model was imported into the Unity 3D application for creating events and build executables and Web version of the proposal, using the programming language C#.

4.2 Population

The population established close professional authors and students from the Polytechnic School - National University of the East - Paraguay first courses of Computer Sciencie Engineering, totaling 80 participants. Of this group 20 were professionals and 60 students.

5 Tests and results

5.1 Tests

The application went through a stage of testing to determine the efficiency and accuracy of data shown by simulations. Problems were manually proposed. They

experienced variation of the parameters, and with the results verified, algorithm settings where tuned to their optimal point. Tests were performed for Windows and GNU-Linux platform, both 64-bit, as they were available for use. Once this stage ended, testing with general public began.

The website address, `www.vuob.info/simvr` was spread through social networks, email. The use of the application was entirely made by the user, without any assistance in addition to the existing instructions in the Web page. Besides disseminating to professionals, tests were conducted at the campus of the Polytechnic School - National University of the East, in one of the computer labs with groups of students from the Engineering Systems career with the help of the Teacher of the Physics matter. First, the demonstration and use of the application was proceeded, which was connected to the website from internet, showing the use of the application (3) with a real connection. Then the students used the application with the simulations prepared, with the support of the teacher and authors in case of doubt. Finally, via web forms for later analysis they were filled.

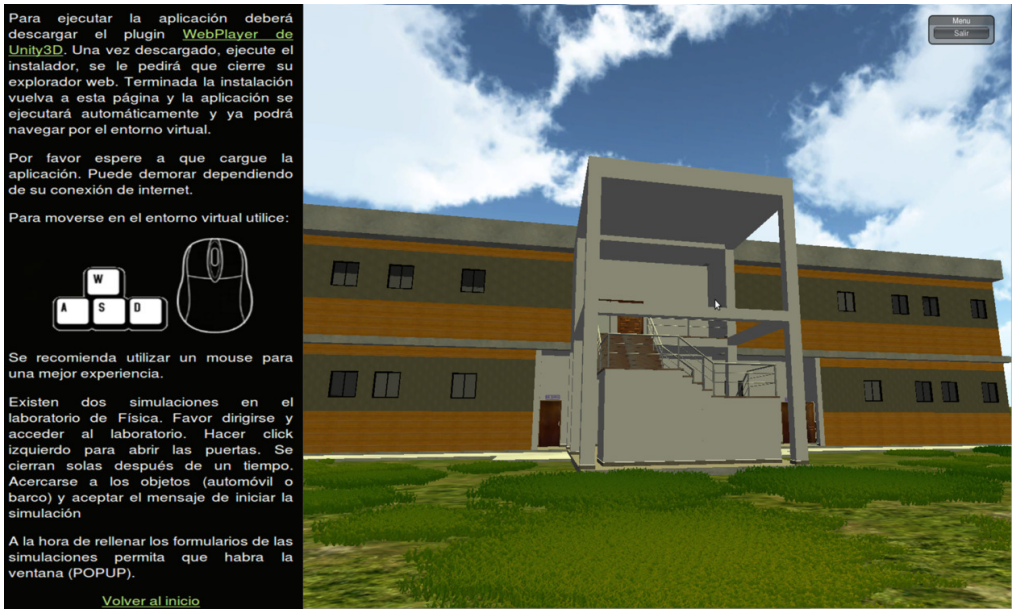


Fig. 3. 3D Aplication on website

The teacher and students showed great interest in incorporating the application in teaching methodology.

5.2 Results

The results obtained with the application are presented. They were extracted from the database and represent the values of the latter form, with respect to the application itself. Each of the indicators under study are detailed.

Degree of Realism The model used in the application is a representation of the laboratory building of the Polytechnic School - National University of the East, being of great importance its resemblance to the actual building, as this aspect is one of the pillars of virtual reality. To do this, textures from photos were used. According to the results, the impact was positive, as in Figure 4 shows that 47.5% of participants responded with the highest note of the evaluation.

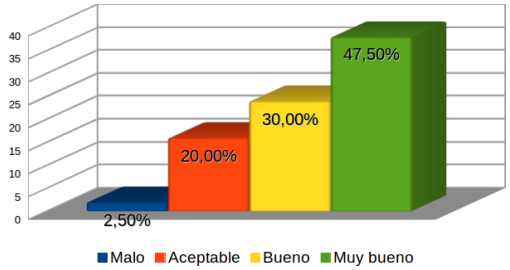


Fig. 4. Degree of Realism

Receipt of Virtual Reality technology The grade of reception of the user is studied with respect to using virtual reality applications. As shown in Figure 5, 95% of the respondents said that they are in favor of the use of this technology in applications.

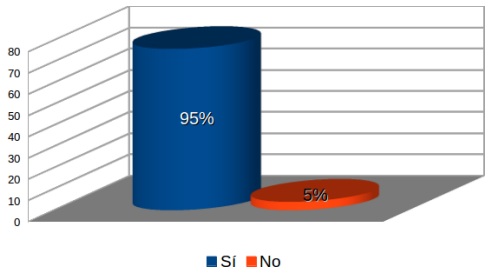


Fig. 5. Receipt of Virtual Reality

Learning using Virtual Reality The results on whether the virtual reality technology helped to some extent to the user better to understand the content presented in the simulations are shown below. As shown, 90% of respondents said that this technology provided some help in understanding the content (see Figure 6).

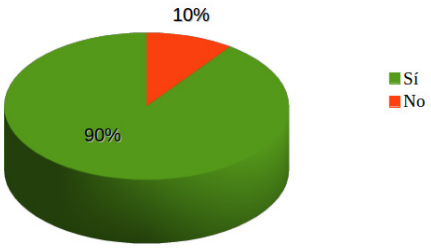


Fig. 6. Learning using Virtual Reality

User Experience with the Application Among the features of virtual reality is the similarity of the movements people make in a virtual environment compared to those made in real life, facilitating their adaptation and immersion. Figure 7 shows that the application was well received and had more positive than negative with respect to its usability.

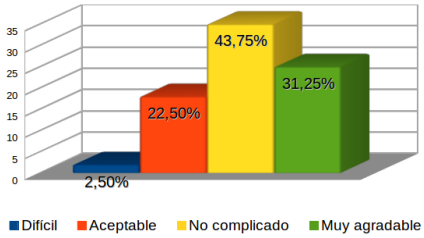


Fig. 7. User Experience

Response Speed Enforcement For the virtual reality experience to be effective, it must respond to real-time user actions. Below it is shown in Figure 8 the response provided by the application as perceived by the user. It is noted, which is in line with expectations, as a normal or quick response provides better immersion in the virtual model.

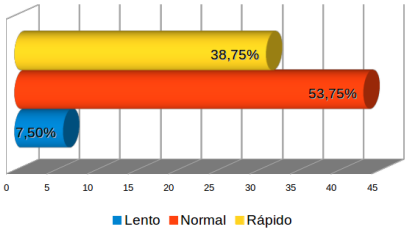


Fig. 8. Speed application response

Portability As a study, the user is asked for his operating system, to demonstrate the application portability. As can be seen in Figure 9, the majority of users use Windows (88.75 %), including its variants: Windows XP, Windows Vista, Windows 7, Windows 8; It is one of the most used systems globally. But it's not the only one, a small but significant percentage of users with GNU / Linux operating system represented by 10 % of the total and another group which is the 1.25 % who used the Mac OSX operating system is observed, which demonstrates the application portability.

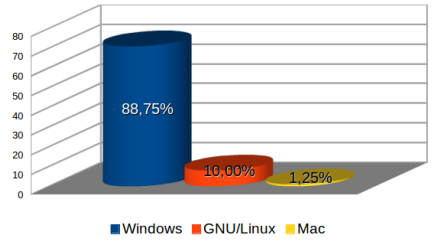


Fig. 9. Application Portability

6 Conclusions and Future Work

The application developed in this work was successful, reaching the main objective, which is to develop and implement a system that allows a user to interact with virtual scenarios in a learning environment.

6.1 Brief review of work

A multiplatform application, both web and desktop for Windows operating systems, GNU / Linux and Mac OSX, allowing people to interact in virtual scenarios that serve as tool in learning different subjects was developed.

Using the Blender application, the laboratory building of the Polytechnic School was modeled, including some furniture and equipment. For the physical appearance were taken pictures and then treated with the Gimp software. The model was exported to Unity3D and two simulations of physical phenomena, the oblique shot and uniform motion were created. With Unity3D the application to the different platforms mentioned above was exported.

In order to show the feasibility of the proposal, several tests with students of the institution and those outside it were made. They showed very satisfactory results about the use of virtual reality technology in conjunction with teaching.

6.2 Major Achievements

This work demonstrates the use of virtual reality in the learning environment. Considering the tests, the main achievements were clarified:

- Using virtual reality technology in a web and multiplatform environment.
- Representation with a high degree of fidelity of the laboratory building of the Polytechnic School - National University of the East.
- Running simulations of physical phenomena and positive feedback of results by students.
- Provide a support tool in the teaching-learning process.

6.3 Future Work

In view the possibilities with the use of virtual reality, it opens to research the following points:

- Development and implementation of a multi user module, enabling the use of avatars thereby enhancing the virtual experience.
- Application development for mobile platforms (Android, iOS, Windows Phone 8.1), extending architectures of execution.
- A comparison of the level of learning using the traditional method and using the application virtual reality.

References

1. Santelices Malfanti I. Parra Márquez J.C., García Alvarado R. Introducción práctica a la realidad virtual. Eds. Universidad del Bío-Bío (2001)
2. Martínez, J., Hilera, J. R., Otón, S. Aplicación de la Realidad Virtual en la enseñanza a través de Internet. http://pendientedemigracion.ucm.es/info/multidoc/multidoc/revista/num8/hilera-oton.html
3. Diego Levis. Realidad Virtual y educación. http://www.diegolevis.com.ar/secciones/Articulos/master_eduvirtual.pdf
4. Realidad Virtual como método de entrenamiento. Realidad Virtual, http://entrenamientovr.wordpress.com/realidad-virtual/
5. AULA. AULA, http://www.aulanesplora.com/
6. Labster. Virtual Lab for Teaching Life Science. http://www.labster.com

Análisis de interacciones y patrones de comportamiento en un curso virtual con Objetos de Aprendizaje

Claudia López de Munain, Silvina Bramati, Zulema Beatriz Rosanigo, Pedro Bramati, Marcela Torrent

Facultad de Ingeniería, Universidad Nacional de la Patagonia San Juan Bosco
Belgrano y Roca – Trelew – Chubut – Argentina
klaucvj@gmail.com, silvina.bramati@gmail.com, brosanigo@yahoo.com.ar, pedrobramati@speedy.com.ar

Resumen. Las propuestas educativas gestionadas por los sistemas de software se pueden estudiar utilizando los registros almacenados de las interacciones y actividades llevadas a cabo en ellas. De esta manera, se obtiene relevante información para mejorar el proceso y el diseño de las mencionadas propuestas. Este artículo presenta los resultados de investigación sobre el análisis de las interacciones y los patrones de comportamiento observados en un curso virtual diseñado con objetos de aprendizaje, complementando el trabajo presentado en CACIC 2014.

Palabras claves: ambiente virtual de enseñanza aprendizaje, objeto de aprendizaje, interacciones, patrones de comportamiento académico.

Abstract. The educational proposals managed by software systems can be studied using the stored records of interactions and activities carried out in them. In this way relevant information to improve the process and the design of these proposals is obtained.
This article presents research results on the analysis of interactions and behavior patterns observed in a virtual course designed with learning object, complementing the work presented in CACIC 2014.

Keywords: virtual learning environment, learning object, interactions, academic behavior patterns.

1 Introducción

Actualmente se utiliza la conceptualización de "EVEA" (Entorno Virtual de Enseñanza y Aprendizaje) para caracterizar en forma amplia a cualquier combinación de propuestas sistematizadas de estudio que prioricen la interacción de los aprendizajes, a partir de algún nivel de virtualidad en relación a variables témporo-

espaciales. Durante los últimos años se ha generalizado su uso en sistemas a distancia (*e-learning*) y semipresenciales (e-*blended learning*), como así también integrado en la enseñanza presencial.

Estos entornos ofrecen una serie de posibilidades para procesos de colaboración, donde el alumnado produce conocimiento de forma activa, formulando ideas que son compartidas y construidas a partir de las respuestas de los demás [1].

Las plataformas tecnológicas utilizadas para esta modalidad de propuestas educativas registran información de todos actores intervinientes: estudiantes, docentes, Institución y cualquier otro usuario que haya participado en la formación. Para los docentes y las instituciones educativas es importante evaluar las actividades e interacciones registradas en los cursos, analizando factores de efectividad y motivación con el fin de mejorar y madurar el proceso enseñanza aprendizaje por medio de plataformas virtuales [2].

El análisis de las interacciones que se producen en el EVEA, tiene por finalidad comprender cómo se produce el proceso de enseñanza y aprendizaje en un espacio virtual colaborativo y a partir de ello, realizar correcciones que permitan mejorar el rendimiento académico de los estudiantes.

A partir de los años noventa se desarrollan una variedad de trabajos que comparten el interés por entender cómo las tecnologías de la información y la comunicación (TIC) pueden facilitar procesos de desarrollo colaborativos en situaciones de enseñanza-aprendizaje, y sobre cómo los entornos de aprendizaje colaborativo pueden mejorar y potenciar la interacción, el trabajo en grupo, y por consiguiente, el resultado del proceso de aprendizaje de los participantes [3, 4].

En base a la experiencia de dictado de varias ediciones de un curso virtual con Objetos de Aprendizaje, se analizaron los datos registrados por el sistema de gestión de aulas virtuales y las encuestas de alumnos, con el fin de identificar patrones de comportamiento de los estudiantes y analizar las interacciones que se producen. En CACIC 2014 se presentó la identificación de los patrones de comportamiento académico acumulado de tres ediciones del curso, en dos momentos diferentes: uno correspondiente a las actividades de los primeros módulos y el otro correspondiente a las actividades de los últimos módulos. En este trabajo se presentan los patrones de comportamiento académico y el análisis de las interacciones que se producen entre los actores participantes en el curso 2013.

2 Objeto de investigación

La finalidad de este estudio es recopilar evidencias acerca de las interacciones y comportamiento de los actores participantes en el curso de posgrado “Diseño y producción de Objetos de Aprendizaje” a efectos de capitalizarlas para introducir mejoras en ediciones futuras.

A continuación se describe el curso y se conceptualizan los tipos de interacciones que ocurren en un EVEA.

2.1 Descripción del curso

El curso está dirigido especialmente a docentes interesados en desarrollar y socializar material didáctico que pueda ser utilizado en diferentes entornos virtuales educativos. Se desarrolla en modalidad virtual durante 12 semanas.

Como herramienta de gestión de cursos virtuales se utilizó MOODLE (Modular Object-Oriented Dynamic Learning Environment). Este entorno facilita la creación de comunidades de aprendizaje en línea permitiendo agilizar y mantener una fluida comunicación con los alumnos durante el cursado.

El modelo pedagógico adoptado se centra en el estudiante, buscando la flexibilidad, la cooperación, la personalización y la interactividad en el proceso de enseñanza y aprendizaje. El estudiante tiene una elección real sobre cuándo, cómo y dónde estudiar, y puede adoptar diferentes caminos y diferentes materiales, algunos de los cuales se encontrarán fuera del espacio formal de formación. La estrategia empleada supone un alumno con cierta autonomía y autogestión, que administra su propio tiempo dentro de un cronograma preestablecido para cada módulo. Se favorece e incentiva además la interacción de los alumnos entre sí y con los tutores mediante la utilización de los foros como herramientas de aprendizaje.

El material didáctico consiste en un conjunto de Objetos de Apendizaje (OA), que se monta en el aula virtual además de algunos artículos seleccionados. Un OA es "un conjunto de recursos reunidos con un propósito educativo, autocontenible e independiente, diseñado y creado en pequeñas unidades digitales que pueden ser adaptadas para maximizar el número de situaciones en que puede ser reutilizado, y que cuenta con una estructura de información externa (metadatos) para facilitar su almacenamiento, identificación y recuperación" [5].

El proceso de enseñanza aprendizaje mediante OA permite aprender construyendo el conocimiento mediante la reflexión, la experimentación, la interacción, la solución de problemas, etc. Son totalmente adecuados para la modalidad virtual con centro en el estudiante.

Cuando los OA son exportados en un formato estandarizado como SCORM, y se utilizan en un EVEA, tienen la capacidad de comunicarse con la plataforma tecnológica que lo sustenta, obtener información del alumno para personalizar el aprendizaje, marcar los recorridos y avances que ha realizado por el alumno.

Los trabajos de fin de módulo se evalúan por los responsables del curso siguiendo una metodología de mejora continua, indicando errores conceptuales si los hubiera y solicitando mejoras en la calidad o completitud de manera de lograr un aprendizaje deseado.

En la práctica la evaluación de los trabajos consiste en establecer una serie de aspectos a observar, evaluar el nivel de cumplimiento y si el nivel establecido como meta no ha sido alcanzado, otorgar una retroalimentación y solicitar las modificaciones correspondientes al alumno, de manera que lo logre.

Esta forma de evaluación se condice con el aprendizaje constructivista, para el cual las evaluaciones también son una instancia de aprendizaje y como tales deben ser utilizadas en pos de lograr que el alumno adquiera el nivel de conocimiento deseado y no solamente para finalizar el proceso de aprendizaje.

2.2 Interacciones

Se entiende por interacción la relación o comunicación entre dos o más personas, y/o los procesos de enseñanza y aprendizaje. Las interacciones involucran la necesidad de conocer, aprender, intercambiar información [6].

Según Vigotsky [7], la interacción es uno de los componentes más importante de cualquier experiencia de aprendizaje. El intercambio de ideas que se realiza mediante interacciones significativas que ocurren entre tutores y estudiantes y entre los mismos estudiantes es un componente clave intelectualmente estimulante de la buena enseñanza. Como manifiesta Monroy [8], las interacciones cumplen funciones en el aprendizaje, relacionadas con el conocimiento que tiene el individuo de sí mismo, de sus procesos cognitivos y metacognitivos. Es el aspecto integrador para combatir el aislamiento del estudiante posibilitando su permanencia y constancia.

En la modalidad virtual se dan fundamentalmente cuatro tipos de interacción [9, 10]:

- *Estudiante-contenido*: Se trata de la interacción de quien estudia y el material educativo que se pone a su disposición. Los contenidos o materiales didácticos planeados para un curso virtual "dialogan" con el estudiante ante la ausencia del docente facilitador. Se da una negociación interna entre los conocimientos existentes en las estructuras cognoscitivas del estudiante y los nuevos contenidos, producto de un tipo de conversación didáctica interna [11]. Esta interacción demanda la disciplina del discente adoptando técnicas de estudios para aprender de forma autónoma, colaborativa y cooperativa.
- *Estudiante-tutor*: se trata de la interacción del estudiante con el docente facilitador. Es un tipo de relación establecida (sincrónica y asincrónica) entre docente y estudiante, y se caracteriza por ser motivadora, afectiva y de interés mutuo. Se refleja en actividades, procesos y mensajes que recíprocamente se generan para aclarar y realimentar conceptos y profundizar en ellos, que mantienen el flujo comunicativo en el proceso formativo.
- *Estudiante-interfaz*: comprende desde las formas de presentación del material de estudio, hasta las características interactivas de los medios de comunicación, manejo de la plataforma educativa, el uso adecuado y eficiente de las herramientas infovirtuales.
- *Estudiante-estudiante*: se trata de la interacción de los estudiantes dentro o fuera del proceso de aprendizaje. Las funciones que cumple esta interacción van desde las sociales hasta las de comparación de grupo, por lo tanto dan medida de la necesidad de esfuerzo y permiten realizar procesos correctos en relación con los resultados de las evaluaciones. Los discentes trascienden las distancias físicas y establecen relaciones amistosas y de interés común que posibilitan la conformación de verdaderas comunidades académicas.

Garrison y Anderson [12] proponen un modelo conceptual de aprendizaje online desde el cual se fomente la interacción entre docentes y estudiantes, la independencia cognitiva y la interdependencia social. Consideran que los aprendizajes serán más profundos y significativos si convergen tres presencias: docente, social y cognitiva.

La presencia docente contempla el diseño curricular y organizativo, se relaciona con el desarrollo cognitivo y el entorno positivo de aprendizaje.

La presencia social incluye todas aquellas interacciones que fomentan la creación de una dinámica grupal, promuevan las relaciones sociales, expresan emociones, y aquellas en las que el grupo de estudiantes se afirme como tal. Es necesaria para establecer un sentimiento de confianza y de pertenencia de una comunidad orientada a la construcción de conocimientos.

Y la presencia cognitiva hace referencia al entorno intelectual, se advierte a través de una serie de cuatro estadios cíclicos que comienza con un evento desencadenante que promueve la exploración, la integración y la resolución, definiendo procesos de pensamiento crítico y creativo [13]. La presencia cognitiva se define como la medida en la que los alumnos son capaces de construir y confirmar significados a través de un discurso sostenido en una comunidad de indagación crítica.

Tabla 1 – Dimensiones, categorías e indicadores según Garrison y Anderson (2005)
Fuente: http://www.revistacomunicar.com/indice/articulo.php?numero=42-2014-09 [21]

PRESENCIA SOCIAL		
Afecto	**Comunicación Abierta**	**Cohesión**
Expresión de emociones Recurrir al humor Expresarse abiertamente	Seguir el hilo Citar los mensajes de otros Referirse explícitamente a los mensajes de otros Hacer preguntas Expresar aprecio Expresar acuerdo	Vocativos Dirigirse o referirse al grupo usando pronombres inclusivos Elementos fáticos, saludos

PRESENCIA COGNITIVA			
Hecho desencadenante	**Exploración**	**Integración**	**Resolución**
Evocativo (inductivo)	Inquisitivo (divergente)	Tentativo (convergente)	Comprometido (deductivo)

PRESENCIA DOCENTE		
Diseño Educativo e organización	**Facilitar el discurso**	**Enseñanza directa**
Fijar el programa de estudios Diseñar métodos Establecer un calendario Emplear el medio de forma efectiva Establecer pautas de conducta y cortesía en la comunicación electrónica (netiquette) Plantear observaciones en el nivel macro del contenido de los cursos	Identificar áreas de acuerdo/desacuerdo Intentar alcanzar un consenso Animar, reconocer o reforzar las contribuciones de los estudiantes Establecer un clima de estudio Extraer opiniones de los participantes, promover el debate Evaluar la eficacia del proceso	Presentar contenidos/cuestiones Centrar el debate en temas específicos Resumir el debate Confirmar lo que se ha entendido mediante la evaluación y el feed-back explicativo Diagnosticar los errores de concepto Inyectar conocimiento desde diferentes fuentes, por ejemplo libros de texto, artículos, Internet, experiencias personales Responder a las preocupaciones técnicas

3 Caso de estudio

Esta investigación analizó las interacciones y determinó patrones de comportamiento académico a través de un entorno virtual del curso sobre Diseño y producción de Objetos de Aprendizaje 2013 en el que participaron 27 alumnos y 3 tutores. Se obtuvieron datos de la participación de los alumnos en las actividades de cada módulo que incluyen lecturas, OA, cuestionarios, entrega de trabajos y participación en foros.

3.1 Metodología de investigación

El método de trabajo es analítico, a través de análisis descriptivo de los datos utilizando las pruebas disponibles en el complemento "Análisis de datos" de Excel.

La fuente de información fueron los estudiantes, tutores y foros de discusión de las aulas virtuales. La recopilación de datos se obtiene del historial del curso almacenado en Moodle y de encuestas realizadas al finalizar el mismo.

Se utilizan en primera instancia los aspectos cuantitativos (número de intervenciones, tiempo de acceso, cantidad de actores) y luego a través de métodos cualitativos se generan categorías para analizar las intervenciones [14, 15, 16]

Para medir la interacción con los contenidos se tiene en cuenta la participación en los foros académicos, el acceso a los materiales de lectura y el seguimiento de la actividad con los OA, a través de los rastros que deja cada SCORM en la plataforma.

En los foros, primero se establecieron las categorías para identificar el tipo de interacción a que responde la intervención, teniendo en cuenta el destinatario, el tipo de mediación, el tipo de participación, el tiempo y pertinencia de respuesta por parte del tutor. Para clasificarlas, se analiza el texto de las intervenciones a nivel de unidad temática, siguiendo la metodología propuesta por Silva y Gros [17] y se evalúan la presencia docente, social y cognitiva en cada intervención, según los indicadores propuestos por Garison y Anderson [12].

Para analizar los ritmos de interacción, se ha tenido en cuenta las categorías de patrones de comportamiento académico de Böhrt y colaboradores [18, 19, 20] que clasificaron el comportamiento en tres categorías y un patrón de comportamiento general, y en cada uno de ellos a su vez se propone tres subcategorías.

La tabla 2 muestra esta clasificación y agrega una columna con las consideraciones que se han tenido en cuenta en este estudio para la categorización.

Tabla 2 - Patrones de comportamiento académico

Categoría	Subcategoría	Consideraciones en este estudio
Inicio	Oportuno	fecha de inicio ± δ
	Demorado	fecha de inicio + δ < inicio < fecha de inicio + 6 δ
	Tardío	inicio < fecha de inicio + 6 δ
Ejecución	Distribuida	Intervenciones uniformes
	Escalonada	Con períodos de pausas intermedios
	Concentrada	Gran pausa inicial, ejecución en la mitad final del período
Finalización	Temprana	Finalización < fecha de fin - 4 δ
	En plazo	Fecha de fin – 4 δ < finalización < fecha de fin – δ
	Al límite	Fecha de fin – δ < finalización < fecha de fin + δ
General	Autorregulado	Inicio oportuno o ligeramente demorado y ejecución distribuida.
	Ejecución inestable	Inicio oportuno y ejecución escalonada.
	Procrastinación	Demora en el inicio, ejecución concentrada y finalización tardía

En las categorías Inicio y Finalización se observa el comportamiento respecto de la fecha inicial y final pautada para cada módulo. Por eso se establece un tiempo δ para distinguir objetivamente cada subcategoría.

El Patrón *Autorregulado* representa al estudiante que inicia tan pronto como puede y demuestra estar a cargo de su propio proceso de aprendizaje distribuyendo las actividades y materiales en el tiempo. En el otro extremo se encuentra el Patrón de *Procrastinación*, el cual responde a un alumno que retrasa el inicio de la tarea tanto como puede, debiendo realizar largas sesiones de intenso trabajo para poder concluir al límite del plazo establecido. Y entre ambos casos, se encuentra el Patrón *Ejecución Inestable* representando al estudiante que tiene un inicio oportuno, pero no logra distribuir sus ejecuciones en el tiempo.

3.2 Resultados

Sobre la cantidad y distribución de intervenciones en los foros:

Se consideraron 5 foros para su análisis, el foro 1 de presentaciones, los foros 3, 4 y 5 de aprendizaje y el foro 5 de consultas, con un total de 267 mensajes que incluyen 68 intervenciones de los tutores y 197 de alumnos. La figura 1 muestra la distribución de intervenciones por foro y tipo de actor.

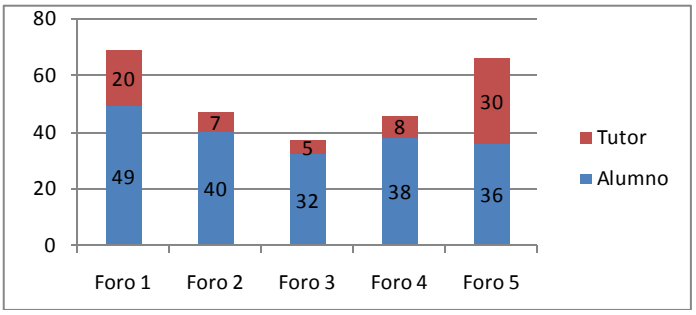


Figura 1: Cantidad de intervenciones por foro

En el foro 1, correspondiente a las presentaciones, el 56% de los participantes intervinieron más de una vez, acotando sobre las intervenciones de otros compañeros. Este porcentaje fue disminuyendo en los restantes foros. La media de intervenciones por alumno en los 5 foros es 7, la mínima es 4 y la máxima es 11, dando un promedio de 1,4 intervenciones por alumno y foro.

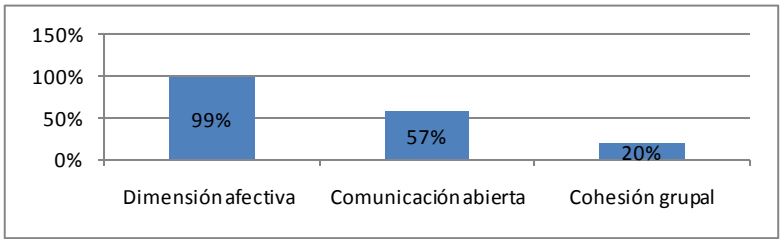


Figura 2: Intervenciones de alumnos con presencia de la dimensión social

Se notó la presencia de la dimensión social en todas las intervenciones, predominando la dimensión afectiva. La figura 2 muestra el porcentaje de su presencia por categoría respecto del total de intervenciones de los alumnos.

Respecto de las interacciones planteadas por Hillman, Willis y Gunawaderna [9]:

Estudiante-contenido: Se pueden diferenciar dos tipos de estudiantes, aquellos que siguen el material de manera exhaustiva, que ingresan a consultar el material en forma periódica y uniforme, que generalmente no tienen mayores dificultades con el contenido, solamente realizan preguntas que contribuyen a enriquecer su aprendizaje. Por otra parte aquellos que ingresan hacia la finalización del módulo y fechas de entrega de trabajos, presentan mayores dificultades en la interpretación del contenido y en ocasiones tienen dudas por no haber analizado todo el material.

*Estudiante-tutor***:** Se observa que los alumnos que inician los módulos en forma oportuna y realizan las actividades uniformemente y autorregulando su aprendizaje, consultan al tutor sobre el contenido con el fin de mejorar el aprendizaje, demostrando haberlo leído y analizado. Aquellos que comienzan en forma tardía generalmente se relacionan con el tutor planteando consultas por no entender la consigna. Se observa que los tutores intervienen oportunamente para alertar cuando se detectan atrasos, alentado a seguir y proponiendo alternativas para que puedan continuar realizando las actividades.

Estudiante-interfaz: el diseño del material utilizando OA propone una interfaz de navegación web a la que la gran mayoría de los cursantes está acostumbrado. En cuanto al EVEA si bien una vez comenzado a utilizar se torna bastante ameno, se observa que las primeras experiencias son más dificultosas. En general los participantes tienden a subutilizar la plataforma de trabajo y no aprovechan todas las ventajas que brinda.

Estudiante-estudiante: Se observa una fluida y enriquecedora relación entre algunos participantes apareciendo intervenciones de tipo social y de colaboración y de discusión respecto de los temas del mismo.

Los resultados de patrones de comportamiento académico correspondientes a las actividades del curso son mostrados en la figura 3.

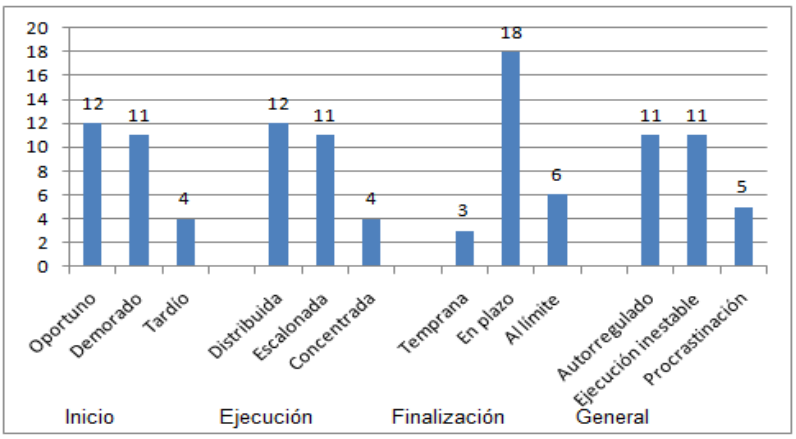


Figure 3. Patrones de comportamiento académico

En el estudio se ha observado que la población que presenta un patrón de comportamiento Autorregulado, con Inicio Oportuno, Ejecución distribuida y Finalización temprana, disminuye en la segunda mitad del curso. En parte, esto es

atribuible a que los alumnos de estos cursos son docentes en actividad y la segunda mitad del curso de posgrado coincide con el período de mayor esfuerzo en su propia actividad laboral.

4 Conclusiones y futuros trabajos

La identificación de los patrones y el análisis de las diferentes interacciones, permite detectar fortalezas y debilidades en el desarrollo del curso (material utilizado, actividades, tutorías, tiempo asignado y uso del aula virtual); y en consecuencia, generar estrategias que consoliden el aprendizaje en el entorno virtual.

El mayor esfuerzo de las intervenciones del tutor se concentra en alentar la participación y favorecer la interacción con los participantes. Las intervenciones de los participantes son mayoritariamente personales y escasamente colaborativas. Al principio del curso la mayoría tiene muy bien controlado el tiempo, y a medida que se avanza, les cuesta más su regulación.

La motivación es un factor que debe ser potenciado por el material, las actividades y la mediación tutorial. Se ha observado que el factor motivacional tiene una relación con el tiempo dedicado al desarrollo de las actividades y por ende en la finalización con éxito del mismo y de las actividades intermedias.

La ejecución concentrada de las tareas en un corto período de tiempo no es deseable porque conlleva hábitos de saturación y probable baja asimilación a largo plazo. Estos casos son fácilmente detectables en el proceso, y es importante que el tutor intervenga para alertar y alentar al alumno a recuperar un ritmo más uniforme.

Las intervenciones en relación al desarrollo del curso y dudas han aparecido en mayor proporción cercanas a las fechas de finalización de actividades. En general, son tipos de interacciones tutor-estudiante y en menos ocasiones, estudiante-estudiante, relativas a preguntas y respuestas para clarificar algún tipo de contenido o consigna.

Mirando hacia el futuro, continuaremos investigando sobre patrones de comportamiento, de usabilidad y valoración de los recursos por parte de los estudiantes y relacionarlos con la metacognición, con el objeto de lograr mejoras en el aprendizaje.

5 Referencias

[1] Resnick, M. (2002). Rethinking Learning in the Digital Age. In G.S. Kirkman, P.K. Cornelius, J.D. Sachs & K. Schwab. (Eds.), The Global Information Technology Report 2001-2002. Readiness for the Networked. (pp. 32-37). World. New York: Oxford University Press. hasp.axesnet.com/contenido/documentos/harvard%20global%20it%20readiness.pdf

[2] López de Munain C., Rosanigo Z.B. & Bramati S. (2014) “La enseñanza virtual con Objetos de Aprendizaje: Interacciones y patrones de comportamiento” XIX Congreso Argentino de Ciencias de la Computación. La Matanza: Univ. Nac. de La Matanza.

[3] Willging, P.A. 2008. “Técnicas para el análisis y visualización de interacciones en ambientes virtuales”. En: Redes. Revista Hispana para el Análisis de Redes Sociales, Vol.14,#6, DOI= http://revistaredes.rediris.es/pdf-vol14/vol14_6.pdf.

[4] Rubia, B., Jorrín, I. & Anguita, R. (2009). Aprendizaje colaborativo y tecnologías de las información y la comunicación. In J. DePablos (Ed.), Tecnología educativa: la formación del profesorado de la era de Internet. (pp. 191-214). Málaga: Aljibe.
[5] Rosanigo, Z. B. (2013) Objetos de Aprendizaje en Gallego G. y Alvarez M. (Coordinadores) Capacitación y gestión del conocimiento a través de la WEB 2.0. Madrid: Dykinson S.L. ISBN: 978-84-9031-406-7. Páginas 177-196
[6] Henao, O. y Zapata, D. (2002). La enseñanza virtual en la educación superior. Bogotá: ICFES.
[7] Vigotsky, L. (1978), El desarrollo de los procesos psicológicos superiores, México, Grijalbo.
[8] Monroy, E. (2007). Metacognición en la educación. [En línea]. http://pizarrondigital. wordpress. com/2007/11/14/metacognicion-en-la-educacion/
[9] Hillman, D. , Willis, D. , & Gunawardena, C. (1994). Learner-interface interaction in distance education: An extension of contemporary models and strategies for practitioners. The American Journal of Distance Education, 8(2), 30-42.
[10] Garrison,D., Anderson, T. & Archer, W. (2010). The First Decade of the Community of Inquiry Framework: A retrospective. The Internet and Higher Education, 13(1-2),5-9.
[11] Holmberg, B. (1986), La educación a distancia, Buenos Aires, Kapeluz.
[12] Garrison,D. & Anderson, T. (2005). El e-learning en el siglo XXI. Barcelona: Octaedro.
[13] Shea, P., Hayes, S., Vickers, & Al. (2010). A Re-examination of the Community of Inquiry Framework: Social Network and Content Analysis. The Internet and Higher Education, 13 (1-2), 10-21.
[14] Taylor, S. y Bodgan R. (1992). Introducción a los métodos cualitativos de investigación, Barcelona: Paidos.
[15] Gil Flores, J. (1994). Análisis de datos cualitativos: Aplicaciones a la investigación educativa. Barcelona: PPU.
[16] Porta, L. y Silva, M. (2003). La investigación cualitativa: El Análisis de Contenido en la investigación educativa. Universidad Nacional de Mar del Plata.
[17] Silva, J. y Gros, B. (2007). Una propuesta para el análisis de interacciones en un espacio virtual de aprendizaje para la formación continua de los docentes. En Sánchez, M. y Revuelta Domínguez, F. (Coords.) Estudio de los comportamientos emocionales en la red. Revista Electrónica Teoría de la Educación: Educación y Cultura en la Sociedad de la Información. Vol. 8, nº 1. Universidad de Salamanca. http://www.usal.es/~teoriaeducacion/rev_numero_08_01/n8_01_silva_gros.pdf
[18] Böhrt, R., Arce, P., y Walker, D. (2007). Patrones de comportamiento académico ante un Programa Computarizado de Enseñanza Personalizada en alumnos de Pregrado de la Universidad Católica Boliviana "San Pablo". Ajayu, 5(1), 110–132.
[19] Böhrt, R., Arce, P., Walker, D., y Romero, C. (2009). Modificación de Patrones de Comportamiento Académico Mediante un Programa de Instrucción Personalizada en Estudiantes Universitarios. Ajayu, 7(1), 1-37.
[20] Böhrt, R., Arce, P., Walker, D., y Romero, C. (2010). Relación entre Patrones de Comportamiento y Rendimiento Académico en Estudiantes Universitarios. Ajayu, 8(2), 157-190
[21] Fernández, M.R. & Valverde, J. (2014). Comunidades de práctica: un modelo de intervención desde el aprendizaje colaborativo en entornos virtuales [A Community of Practice: An Intervention Model based on Computer Supported Collaborative Learning]. Comunicar, 42, 97-105. http://dx.doi.org/10.3916/C42-2014-09

Mundos Virtuales en UNNOBA

Claudia Russo[1], Mónica Sarobe[1], Tamara Ahmad[1], Michel Lombardo[2], Paula Lencina[1], Benjamín Cicerchia[1], Hugo Ramón[1]
[1] Instituto de Investigación y Transferencia en Tecnología (ITT), Escuela de Tecnología, Universidad Nacional del Noroeste de la Provincia de Buenos Aires (UNNOBA)
[2] Becariode la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CIC)

{claudia.russo, monica.sarobe, tamara.ahmad, michel.lombardo, paula.lencina, lucas.cicerchia, hugo.ramon}@itt.unnoba.edu.ar

Abstract. La incorporación de Tecnologías de la Información y Comunicación (TICs) han revolucionado la forma de enseñar y aprender. Teniendo en cuenta que su evolución es cada vez mayor, este trabajo se centra en los avances realizados desde el desarrollo del Entorno Virtual de Enseñanza y Aprendizaje 3D (EV3D) en la Universidad Nacional del Noroeste de la Provincia de Buenos Aires (UNNOBA) y su conexión con el Entorno Virtual de Enseñanza y Aprendizaje (EVEA) ya oficialmente utilizado, UNNOBA Virtual.

Keywords: EVEA**,** EV3D, mundos virtuales, Avatar, OpenSim.

1 Introducción

La evolución de las TICs ha crecido considerablemente en los últimos años, brindando muchas posibilidades en casi todos los aspectos de nuestras vidas, generando un importante cambio en la comunicación e interacción entre personas [1].
Estas tecnologías aplicadas a la educación provocan el nacimiento y crecimiento de nuevos espacios que favorecen los procesos de enseñanza y aprendizaje, en particular en la educación a distancia. Entre estas nuevas tecnologías se puede mencionar los "Sistemas de Gestión de Aprendizaje" o LMS (Learning Management System) [2] que ofrecen diversas herramientas para facilitar su uso en grupos de estudio y permiten controlar las actividades dentro de un proceso de enseñanza y aprendizaje, lo que de otra manera podría tornarse difícil.
Repositorios, salas de chat, foros de discusión, libros de calificaciones, son sólo algunas de las herramientas que ofrecen los LMS. Esto hace que sean un recurso beneficioso para el apoyo y mejora de los procesos de enseñanza. Sin embargo presenta ciertas limitaciones, en cuanto a la representación de algunos contenidos y en el modo de interactuar con ellos [3].
Por su parte los EV3D, son una nueva plataforma que, aunque su principal objetivo no se relaciona con la educación, están tomando un papel muy importante en el área educativa [4]. Éstos integran conceptos utilizados en la Web e ideas del mundo real, para mostrar la información de una forma más llamativa y con muchas más

posibilidades. Simulan espacios físicos, reales o no, que permiten a los usuarios interactuar entre sí en tiempo real.
La UNNOBA cuenta con un EVEA llamado UNNOBA Virtual [5], donde los docentes pueden impartir cursos y extender el aula con todas las potencialidades que el EVEA brinda. Asimismo, en el ámbito de la Escuela de Tecnología de la UNNOBA, y enmarcado en el proyecto de investigación "**Tecnologías exponenciales en contextos de realidades mixtas e interfaces avanzadas**", se desarrolla el EV3D para la UNNOBA. En este contexto se trabajó en el diseño y generación del EV3D y actualmente se continúa trabajando en contenidos específicos para este tipo de entornos; sumado a ello se trabaja en el diseño y creación de avatares pedagógicos que posibiliten enriquecer los procesos de enseñanza y aprendizaje. En este punto y avanzando con la investigación, se intenta unificar las funcionalidades de ambos entornos, dado a que es posible juntar un LMS y un EV3D, con el fin de crear un espacio colaborativo de aprendizaje. Además se trabaja en el diseño de contenidos y actividades específicas acorde a ambos entornos a fin de poder relevar, medir y comparar el impacto de la introducción de un EV3D en las diferentes modalidades de enseñanza presentes en el ámbito de la UNNOBA.
Este trabajo intenta mostrar los avances alcanzados en la conexión del EVEA, tomando como inicio la plataforma MOODLE, con el EV3D.

2. Contexto

Este trabajo de investigación está enmarcado en el proyecto "Tecnologías exponenciales en contextos de realidades mixtas e interfaces avanzadas", aprobado y financiado por la Secretaría de Investigación, Desarrollo y Transferencia de la Universidad Nacional del Noroeste de la Provincia de Buenos Aires (UNNOBA) en el marco de la convocatoria a Subsidios de Investigación Bianuales 2015 (SIB 2015).
Dicho proyecto es continuación del proyecto "El desafío de las nuevas tecnologías de la información y las comunicaciones en los contextos educativos", acreditado en la convocatoria a Subsidios de Investigación Bianuales 2013 (SIB 2013).
El equipo de trabajo, constituido por docentes e investigadores, becarios y alumnos, se desempeña en el Instituto de Investigación y Transferencia en Tecnología (ITT) y trabajan en conjunto con la Escuela de Tecnología de la UNNOBA.

3. Simulación de un EV3D en UNNOBA

3.1 Simulación

Existen varias definiciones que se pueden encontrar sobre simulación, aunque tomaremos una que se adecúa a nuestra temática y es la que provee R. E. Shannon,

quien define a la simulación como "un proceso para diseñar un modelo de un sistema real y llevar a términos experiencias con él, con la finalidad de comprender el comportamiento del sistema o evaluar estrategias -dentro de los límites impuestos por un cierto criterio o un conjunto de ellos- para el funcionamiento del sistema".
Es decir, a través de la simulación se intenta reproducir los rasgos, características y apariencia de un sistema real, pudiendo representar desde conceptos básicos como pueden ser situaciones de la vida cotidiana, juegos, y hasta sistemas complejos; pudiendo realizar de esta manera un estudio del sistema real más económico, sencillo y con menos riesgos. Una vez hecha esta representación se pueden llevar a cabo pruebas y/o modificaciones con el principal objetivo de estudiar y comprender mejor el sistema.

3.2 EV3D y educación

El concepto de EV3D ha ido evolucionando en el tiempo. En sus orígenes éstos se utilizaban con propósitos militares, más concretamente para simulación de vuelo. Con su posterior comercialización a diferentes áreas fueron surgiendo nuevas tecnologías dependiendo de su función. Con los juegos de computadoras apareció un nuevo tipo de EV3D interactivo, conocido como motor de juego, y que con el paso del tiempo fue desarrollándose y obteniendo un alto grado de calidad gráfica [6].
Hoy en día existen varios Mundos Virtuales, que son EV3D de múltiples usuarios simulado en un espacio tridimensional, en la cual las personas pueden interactuar entre sí y con diferentes objetos a través de una representación virtual que se denomina *avatar* [7]. Algunas de las características de estos entornos son:

- Inmersivos: dan la sensación a los usuarios de estar dentro del entorno virtual.
- Interactivos: los usuarios pueden comunicarse, en tiempo real, con otros usuarios mediante voz y texto, e interactuar con los diferentes objetos del entorno virtual. Permitiendo que se potencie la característica de inmersión.
- Personalizables: permiten la construcción o modificación de los escenarios, con sus respectivos objetos.
- Persistentes: el sistema sigue desarrollándose a pesar de que algún o incluso todos los usuarios no estén online. Además, los estados actuales son guardados al momento de cerrar sesión.

Todo esto ofrece una mayor diversidad en los entornos, permitiendo que se puedan realizar ciertas actividades en las cuales en la vida real serían muy costosas o de gran riesgo para la vida humana.
Se debe tener en cuenta que, con el uso de EV3D, es posible recrear espacios virtuales que tengan correspondencia o no con la realidad.
Hace ya unos años que muchas universidades han empezado a utilizar estas herramientas como punto de investigación y desarrollo. Ibañez, Naya y López, en "Mundos virtuales como canal de comunicación entre escuelas y museos"[8], por ejemplo, relatan una experiencia realizada entre escuelas y museos, en la cual el

personal del museo brindaba diferentes charlas informativas a los alumnos de las instituciones involucradas. López-Hernández, en "El campus de la Universidad Carlos III en SecondLife"[9] describe cómo la biblioteca de la Universidad Carlos III utiliza SecondLife para crear un espacio en donde representaron su mismo edificio y brindan a los alumnos distintos lugares para reuniones, para estudio e incluso un lugar para recreos. Makosz, en "Using 3D Virtual Worlds–OpenSim" [10], cuenta sus experiencias como profesor de Inglés como lengua extranjera y Ciencias de la Computación a estudiantes universitarios. Estos son sólo algunos ejemplos de la inmensa cantidad de experiencias que se realizan por las diversas universidades.

3.3 La experiencia en UNNOBA

En UNNOBA, como resultado de la primera instancia de investigación, se creó el EV3D y se replicó el edificio de la escuela de Tecnología de la UNNOBA, tal como se muestra en la Ilustración 1.

Ilustración 1: réplica del edificio Eva Perón (Escuela de Tecnología) UNNOBA, en el EV3D

Además se realizó una experiencia en el uso del EV3D en UNNOBA, en donde se plantean actividades dentro de la asignatura Análisis y Diseño de Sistemas, que se dicta para alumnos de tercer año de las carreras de informática, pudiendo resolverlas colaborativamente dentro del entorno [11]. En la Ilustración 2, se muestra una captura real de la actividad.

Ilustración 2: Captura de pantalla de actividad realizada en el marco de una asignatura con el EV3D

Esta conexión entre tecnología y pedagogía genera un nuevo paradigma educativo, y rompe con algunos aspectos como son el tiempo, la distancia y la presencia del paradigma tradicional. Para el alumno es un nuevo y llamativo espacio en donde las posibilidades brindadas por los entornos son muy grandes. Pero para el profesor, por su parte, es un reto poder hacer uso de estas tecnologías para lograr una clase interesante, lo que lo obliga a buscar nuevas estrategias pedagógicas adecuadas [4].

4. Conexión LMS y EV3D

Como dijimos anteriormente, el objetivo del avance de esta investigación es realizar la unificación del EVEA actualmente utilizado en nuestra universidad (UNNOBA Virtual) con el reciente EV3D creado.
Para eso debimos investigar qué herramientas nos permitirían realizar estas conexiones, identificando de esta manera los elementos de trabajo que se describen a continuación:

4.1 Elementos de trabajo

Moodle[1]

Las plataformas LMS son herramientas que nos permiten organizar material y actividades para ayudar a los procesos de aprendizaje y enseñanza, proveen diferentes herramientas que nos dan la posibilidad de gestionar las matrículas de los alumnos, hacer seguimiento de los procesos educativos, evaluar, comunicarse y muchas más funciones que permitan la organización de cursos de formación a distancia.
Moodle es uno de los LMS más utilizado, se distribuye bajo la licencia GNU GPL y posee una amplia comunidad de usuarios y desarrolladores que hacen que el proyecto siga creciendo.

[1]Sitio oficial del LMS Moodle, https://moodle.org

UNNOBA Virtual, el EVEA utilizado en UNNOBA está basado en la plataforma Moodle.

OpenSimulator

Como se mencionó anteriormente, hoy en día existen una gran variedad de EV3D, pero lo más utilizados en el área de la educación son el conocido SecondLife desarrollado por Linden Lab, OpenSimulator que es una aplicación Open Source y OpenWonderland y tiene sus orígenes en la herramienta Wonderland creada por Sun Microsystems. Este último entorno luego de que la empresa fuera comprada por Oracle, el proyecto fue abandonado y los mismos usuarios decidieron continuar su desarrollo cambiando el nombre a OpenWonderland.
En investigaciones anteriores realizadas por la UNNOBA se decidió utilizar como plataforma a OpenSimulator, también conocido como OpenSim[2], con el cual ya se ha realizado una experiencia [10]. Los principales motivos de su elección fueron debido a algunas características que la diferencia de las restantes herramientas y que hacen que sus funcionalidades se puedan aprovechar mucho más. Algunas de ellas son: que es una plataforma completamente libre y gratuita, que se distribuye bajo la licencia BSD lo cual al ser OpenSource originó que existiese una amplia variedad de programadores que dan soporte a dicho proyecto. Que a pesar de estar programado en el lenguaje C#, lo que provocaría tal vez un inconveniente para ejecutar la aplicación en sistemas operativos que no sean los de Microsoft Windows, existe el proyecto Mono, cuya finalidad es la de permitir correr aplicaciones desarrolladas en C# en plataformas de GNU-Linux.
Además, OpenSim nos ofrece la posibilidad de instalar nuestros propios servidores de forma local, sin depender de terceros y dándonos el control total de la herramienta.
Se descubrió también que, por ejemplo, en SecondLife para algunos aspectos se precisa de una inversión de dinero, lo que provocó un rechazo a la hora de elegir una plataforma. En cambio OpenWorderland era un buen candidato para utilizar, compartía algunas características con OpenSim y tenía por ejemplo algunas ventajas como tener incorporado un módulo de voz, con el cual los usuarios pueden comunicarse en tiempo real. Dicha opción en OpenSim, no viene incluida por defecto, pero se puede configurar para su utilización.
El motivo final por el cual se decidió utilizar la plataforma OpenSim y no a OpenWonderland fue que OpenSim permite extender sus funcionalidades a través de la incorporación de módulos externos, como es el caso de Sloodle, que se describe a continuación.

Sloodle[3]

Sloodle cuyas siglas significa Simulation Linked Object Oriented Dynamic Learning Environment o Simulación Orientada a Objetos Vinculados en Ambiente de Aprendizaje Dinámico, es un proyecto Open Source, cuyo objetivo es unir las

[2]Sitio oficial de OpenSim, http://opensimulator.org/wiki/Main_Page
[3]Sitio oficial de Sloode, http://www.sloodle.org

funciones que provee el Sistema de Gestión de Aprendizaje Moodle con los entornos virtuales multiusuario 3D como lo son SecondLife u OpenSimulator. De esta forma todo el material que fuese creado en Moodle, podrá ser utilizado por los profesores y estudiantes desde el EV3D.
Entre las funcionalidades más útiles que provee Sloodle se pueden mencionar:

- RegistrationBooth: esta herramienta vincula los avatares de cada usuario con sus cuentas correspondientes en Moodle. De esta forma, se puede hacer un seguimiento de cada usuario, como registrar su progreso y su gestión.
- Web-Intercom: posibilita la participación en conversaciones escritas entre los usuario por chat, de forma simultánea en el entorno virtual 3D y una sala de chat de Moodle. A su vez, las discusiones pueden ser guardadas de forma segura en una base de datos de Moodle. Esta herramienta es de gran ayuda, porque si algún alumno no puede tener acceso al mundo virtual en ese momento, por diversas cuestiones, aun así puede interactuar en las charlas que vayan ocurriendo en el entorno.
- Presenter: provee la posibilidad de hacer presentaciones en el EV3D de diapositivas y/o páginas web que se encuentran en Moodle, sin necesidad de convertir o subir los archivos al entorno.
- Barra de herramientas multifuncional: esta herramienta mejora la interfaz del usuario. Permite la actualización de blogs de Moodle desde el EV3D, y además posee una serie de botones para expresar diferentes gestos dentro del aula de clases.
- MetaGloss: permite tener acceso a un Glosario que se encuentre en Moodle.
- QuizSloodle y Drop Box: a través de estos módulos se pueden realizar evaluaciones o que los usuarios presenten trabajos que se guardarán en un Drop Box virtual (almacenamiento virtual), que luego los profesores pueden revisar los detalles de las entregar y hacer sus respectivas correcciones.
- QuizChair: esta herramienta permite a los alumnos resolver un cuestionario de tipo múltiple choice y almacenar las resoluciones en Moodle.

5. Algunas conclusiones

De las primeras pruebas realizadas, se pudo lograr la correcta instalación de los elementos y realizar la conexión de Soodle con una versión de Moodle idéntica a la que se utiliza para el EVEA UNNOBA Virtual. Se pudieron probar algunas características y funcionalidades de la integración.
Asimismo, se trabaja en la definición de avatares pedagógicos, con rol de tutor virtual, para ser utilizados en los EVEAs; y en el diseño de contenidos específicos para UNNOBA Virtual y el EV3D desarrollado.
La propuesta no sólo busca crear un espacio colaborativo de aprendizaje con el diseño de contenido y actividades específicas acorde a ambas plataformas, sino también intentar medir y comparar el impacto de la introducción de una plataforma e-learning con mundos virtuales 3D a diferentes modalidades de enseñanza, totalmente a distancia y semi-presencial.

5. Trabajos futuros

Se diseñarán y generarán contenidos específicos para el EV2D de la universidad (UNNOBA Virtual), actualmente en uso, y para el EV3D propuesto para poder así comparar la experiencia de alumnos y profesores en los ambos entornos.

Se determinará el rol del tutor virtual, mediante el uso de avatares y a su vez se intentará simular dos experiencias dentro del EV3D, una simulando educación totalmente a distancia utilizando personas que no posean trato presencial y otra con personas que si lo posean simulando una educación semi-presencial y así poder extraer conclusiones de ambas experiencias tanto de alumnos y profesores, en el uso del EV3D en las diferentes modalidades de enseñanza.

De dichas experiencias se espera recabar la información necesaria para determinar las ventajas que este aporta a la enseñanza con el fin de unificar ambos entornos, 2D y 3D, y que está pase a ser una nueva herramienta de enseñanza para docente y alumnos en UNNOBA Virtual.

Referencias

[1] Esteve Mon, Francesc Marc; Gisbert Cervera, Mercè. Explorando El Potencial Educativo De Los Entornos Virtuales 3D. Teoría de la Educación. Educación y Cultura en la Sociedad de la Información, vol. 14, núm. 3, septiembre-diciembre, 2013.

[2] Ing. Rambo Alice R. Plataformas de Educación a Distancia. 2009. Disponble en: http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/Educacion__Distancia_Alice_2009.pdf

[3] Francisco A. Lizarralde y Constanza R. Huapaya. Análisis de una Plataforma Virtual 3-D Descentralizada para el Desarrollo de Simulaciones Educativas. 2012.

[4] Javier Alejandro Jimenez Toledo. Modelo virtual inmersivo 3D como estrategia didáctica en la educación. 2012.

[5] Entorno Virtual de Enseñanza y Aprendizaje (EVEA) de la UNNOBA, http://virtual.unnoba.edu.ar

[6] Miguel Lozano, Carlos Calderón. Entornos virtuales 3D clásicos e inteligentes: hacia un nuevo marco de simulación para aplicaciones gráficas 3D interactivas. 2003. Disponible en: http://www.uv.es/~agentes/publicaciones/aepia%5Blozano%5D.pdf

[7] Cecilia Sanz, Alejandra Zangara, Magda Lorena Escobar G. Posibilidades Educativas de SecondLife. Experiencia docente de exploración en el metaverso. Revista Iberoamericana de Educación en Tecnología y Tecnología en Educación N°13, Junio 2014.

[8] Ibáñez, Luis Antonio Hernández; Naya, Viviana Barneche; Lopez, RocioMihura. Mundos virtuales como canal de comunicación entre escuelas y museos. 2012.

[9] López-Hernández, Francisco. El campus de la Universidad Carlos III en SecondLife. El profesional de la información, 2008, noviembre-diciembre, v. 17, n. 6, pp. 657-661. Disponible en: http://www.elprofesionaldelainformacion.com/contenidos/2008/noviembre/08.pdf

[10] Makosz, A. Using 3D Virtual Worlds - OpenSim, Quest Atlantis - to Teach International School Students Computer Science and Human Values. Proceeding of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2011, pp. 2480-89.

[11] Hugo Ramón, Claudia Russo, Mónica Sarobe, Nicolás Alonso, Leonardo Esnaola, Tamara Ahmad, Franco Padovani. El uso de los Entornos Virtuales 3D como una herramienta innovadora en propuestas educativas mediadas con tecnología. TE & ET; no. 12, abril 2014. Disponible en: http://teyet-revista.info.unlp.edu.ar/nuevo/files/No12/TEYET12-art08.pdf

XII Workshop de Ingeniería de Software (WIS)

7429 | Una Propuesta de Implementacion MDD y TDD en el Dominio de Sistemas de Procesamiento Transaccional
Hernán Zbucki (UNLP), Claudia Pons (CAETI)

7449 | Ejecución y monitorización distribuida de procesos de negocios: una implementación con Bonita Open Solution
Leonardo Damian Karabogosian (UNLP), Patricia Bazán (UNLP), José Martinez Garro (UNLP)

7491 | Comportamiento de Métricas para Proyectos de Explotación de Información en PyMEs
Diego Basso (UNLaM), Darío Rodríguez (UNLA), Ramon Garcia Martinez (UNLA)

7492 | Revisión Sistemática Comparativa de Evolución de Métodos de Extracción de Conocimiento para la Web
Juan Manuel Rodriguez (UBA), Hernan Merlino (UNLA), Ramon Garcia Martinez (UNLA)

7500 | Mejora de la mantenibilidad con un modelo de medición de la calidad: resultados en una gran empresa
Emanuel Irrazábal (UNNE)

7515 | Características funcionales avanzadas de los BPMS: Análisis comparativo de herramientas
Anahí Rodriguez (UNLP), Patricia Bazán (UNLP), F. Javier Díaz (UNLP)

7518 | Mejoras a un Modelo Léxico mediante Mapas Conceptuales
Alberto Sebastián (UCA), Graciela Hadad (UB)

7609 | Adopción de BPMN para Modelado CIM y su Transformación hacia Vistas de un modelo PIM
Leopoldo Nahuel (UTN), Leandro Rocca (UTN FRLP), Cecilia Ariste (UTN FRLP), Matías Caputti (UTN FRLP), Iván Zugnoni (UTN FRLP), Julieta Ponisio (UTN FRLP), Lautaro Mendez (UTN), Roxana Giandini (UTN FRLP)

7610 | Técnicas de elicitación de requerimientos
Alejandro Oliveros (UADE), Leandro Antonelli (UNLP)

7624 | Estrategia para la Calificación y Priorización de Nuevos Requerimientos de Software Propietario Utilizando un Registro de Historial de Clientes
Sara Ester De Federico (UTN-FRRo), Mariela Avogradini (UTN-FRRo), Noelia Sincosky (UTN-FRRo), Javier Gago (UTN-FRRo), Diana Moschetti (UTN-FRRo), Lucía Melfi (UTN-FRRo), Silvio Gonnet (UTN)

7660 | SIGC – Herramienta para la administracion de SGC certificado bajo ISO 9001
Ariel Pasini (UNLP), Silvia Esponda (UNLP), Delfina Díaz (UNLP), Julieta Calabrese (UNLP), Maximiliano Swaels (UNLP), Marcos Boracchia (UNLP), Patricia Pesado (UNLP)

XII Workshop de Ingeniería de Software (WIS)

7443 | Umbrales sugeridos para promedios de métricas de diseño de una aplicación en Java
Jorge Ramirez (UNSa), Gustavo D. Gil (UNSa), Carina Jimena Reyes (UNSa)

7448 | Verification and validation of domain specific languages using Alloy
Ana Garis (UNSL), Alejandro Sanchez (UNSL)

7483 | Modelo para aplicaciones sensibles al contexto (MASCO): Formalización de las capas relacionadas con datos tomados desde el contexto
María del Pilar Galvez Díaz (UNJu), Nélida Raquel Cáceres (UNJu)

7504 | Usabilidad con AOP: Comparación de Enfoques y Herramientas
Sandra I. Casas (UNPA), Natalia Trejo (UNPA), Juan Gabriel Enriquez (UNPA), Roberto Farías (UNPA)

7550 | Modelo de Aplicaciones Sensibles al Contexto (MASCO), caso de estudio: Robot móvil recolector de objetos
Evelina Carola Velazquez (UNJu), María del Pilar Galvez Díaz (UNJu), Ariel Guzmán Palomino (UNJu)

7578 | Datasheet Modeler: Una herramienta de soporte para el desarrollo de funcionalidades en Líneas de Productos de Software
Mailen Mancuso (UNCOMA), Agustina Buccella (UNCOMA), Alejandra Cechich (UNCOMA), Maximiliano Arias (UNCOMA), Matias Polla (UNCOMA)

7587 | Adaptability-based Service Behavioral Assessment
Diego Anabalon (UNCOMA), Martín Garriga (UNCOMA), Andrés Pablo Flores (UNCOMA), Alejandra Cechich (UNCOMA)

7602 | Un Análisis comparativo de rendimiento en Aplicaciones Móviles Multiplataforma
Lisandro Delia (UNLP), Nicolás Galdámez (UNLP), Leonardo Corbalán (UNLP), Pablo Thomas (UNLP), Patricia Pesado (UNLP)

7637 | Resolución más eficiente de dependencias Java
Martín Jorge Agüero (UNLP), Luciana Ballejos (UTN-FRSF)

7649 | Características del desarrollo de software en la ciudad de Corrientes
Gladys N. Dapozo (UNNE), Cristina Greiner (UNNE), Emanuel Irrazábal (UNNE), Yanina Medina (UNNE), María de los Angeles Ferraro (UNNE), Berenice Lencina (UNNE)

Una Propuesta de Implementación MDD y TDD en el Dominio de Sistemas de Procesamiento Transaccional

Hernán Zbucki[1], Claudia Pons[2]

[1] Facultad de Informática UNLP
[2] Facultad de Informática, UNLP. Universidad Abierta Interamericana, UAI. CIC (Comisión de Investigaciones Científicas de la Provincia de Bs. As.
{hernan.zbucki}@gmail.com
{cpons}@lifia.info.unlp.edu.ar

Abstracto. La ingeniería de software establece que la construcción de programas debe ser encarada de la misma forma que los ingenieros construyen otros sistemas complejos. Los sistemas de procesamiento transaccional no son la excepción. Para lidiar con algunos de los desafíos de construir estas soluciones, se introduce una propuesta de implementación dual MDD y TDD. Esta propuesta define una base de conceptos comunes, obtenidos del análisis de soluciones preexistentes, y experiencias de los desarrolladores, con el objeto de definir elementos del dominio. En paralelo, se conjugarán estrategias de pruebas unitarias, como simulación de las transacciones entrantes en los sistemas bajo desarrollo. El conjunto de ambas metodologías intentará definir un *framework* orientado al dominio que fomente la reutilización de código, reduciendo los esfuerzos de mantenimiento. Así también, dada la naturaleza gráfica de los modelos, se intentará mejorar la visibilidad del sistema construido, tanto para los colegas de equipo, como para los *stakeholders.*

Palabras clave: Payment Gateways, MDD, TDD, refactoring, patrones.

1 Introducción

Los sistemas de procesamiento de transacciones son unos de los más usados y de los más sofisticados dentro de los sistemas de información de gran escala. Particularmente un *CAS*[1] debe manejar varios tipos de transacciones simultáneas, y en gran volumen por unidad de tiempo, tales como compras, autenticaciones, transferencias, devoluciones, balances, promociones, etc. [1]. Más allá del dominio de aplicación, los puntos más delicados de un sistema de proceso de transacciones (particularmente los de tarjetas de crédito/débito), son la *performance* y la *seguridad*. Respecto a la *performance*, el factor primordial es el tiempo que lleva autorizar y completar una transacción de punta a punta, mientras que en el aspecto de *seguridad* el tema importante es la prevención del fraude y la confidencialidad de la información

[1] CAS: Acrónimo inglés para el término *Card Authorization System*, o Sistema de Autorización de Tarjetas.

financiera. [2] Ambos requerimientos están ligados: Los procesos de verificación, autenticación y criptografía, que hacen a la seguridad de cualquier transacción, reducen el rendimiento y el tiempo de respuesta. Además se suma el efecto del aumento del volumen de transacciones a medida que se expande la red transaccional. Esto genera una situación donde se denota un consumo incremental de recursos computacionales, ya sea el porcentaje de uso de los procesadores de los servidores, o los tiempos de acceso a mecanismos de persistencia.

Otra problemática común en estas soluciones es la *frecuencia de actualización del sistema* (referido a nuevas reglas de negocio, que generan cambios en los algoritmos), y la *capacidad de mantenimiento* de los componentes dentro de ellos. Los gobiernos, los bancos y otras entidades interesadas crean, y/o modifican normas y reglas de negocio recurrentemente, para satisfacer uno o varios objetivos corporativos. La motivación a estos cambios está ligado a la competencia entre compañías y bancos, de modo que son presionadas para ofrecer nuevos servicios o modificar los servicios existentes con frecuencia. [1] Estas situaciones causan constantes revisiones de los sistemas de autorización, aumentando la complejidad de mantenimiento de los mismos.

Los desarrolladores dedicados al dominio en cuestión están bajo un dilema importante ya que deben mantener un equilibrio delicado entre la calidad, la expectativa de vida del procesador transaccional y los costos de producción presupuestados. Para mantener el sistema funcionando, y a la vez, cumpliendo con las nuevas reglas de negocio o de servicio, sufren evoluciones constantes que muchas veces incurren negativamente en la calidad de la solución integral.

Por otro lado, la ingeniería de software establece que la construcción de programas debe ser encarado de la misma forma que los ingenieros construyen otros sistemas complejos, como puentes, edificios, barcos y aviones. La idea básica consiste en observar el sistema de software a construir como un producto complejo y a su proceso de construcción como un trabajo ingenieril. Es decir, un proceso planificado basado en *metodologías formales apoyadas por el uso de herramientas*. [3] Sin embargo, los enfoques actuales de construcción de software no siempre son suficientes para tratar los inconvenientes relacionados a los efectos de cambios de tecnología y a los efectos que surgen de la necesidad de cambios en los requerimientos de forma recurrente. [4]

En este artículo se introduce un trabajo que es parte de un proyecto académico de I/D/I, pero que estuvo en un principio enmarcado en el ámbito de una empresa multinacional (*PointPay Inc., www.pointpay.net*). Inicialmente, esta compañía definió una serie de problemáticas encontradas en su proceso de construcción de sistemas de procesamiento transaccional. Los autores de este trabajo evaluaron entonces los tres requerimientos principales recolectados: la *reusabilidad* de la mayor cantidad de componentes posibles, la *reducción de costos de mantenimiento*, y la *convergencia de criterios* de diseño entre desarrolladores.

Como resultado, dentro del ámbito académico del LIFIA-UNLP, se desarrolló esta propuesta de implementación MDD y TDD orientada específicamente a este dominio. Esta propuesta de implementación dual comprende en primer lugar la definición de un *framework* de elementos de dominio, la posterior formalización del mismo a través de un lenguaje específico de dominio (DSL), la construcción de una herramienta de transformación de las instancias de los metamodelos a código funcional, y la integración de pruebas de unidad (metodología TDD) por tipos de transacción

soportados. De esta forma este proyecto pretende cumplir los objetivos planteados de forma tanto corporativa (requerimientos iniciales) como académica (metodología formal para la solución de los requerimientos planteados).

La organización de este artículo es la siguiente: en la sección 2 se introducirá el estudio realizado sobre el dominio transaccional para refactorizar el código y definir una base de conceptos, que a su vez fueran los pilares fundacionales para implementar un DSL (metodología MDD-MDA) basada en elementos de dominio. En la sección 3 se describe una propuesta de implementación TDD orientada al dominio transaccional, e integrada al modelo MDD que se introdujo en la sección 2. En la sección 4 se describe la evaluación de esta implementación, con los resultados y mediciones obtenidos dentro del ámbito de la compañía. En la sección 5 se presenta un conjunto de artículos de otros autores relacionados al tema, que ayudaron a orientar este trabajo. Por último, en la sección 6 se sumarizan las conclusiones y el eventual aporte de este trabajo sobre el estudio del dominio del procesamiento de transacciones.

2 Propuesta de Refactorización e Implementación MDD

El trabajo concreto realizado hasta la actualidad consta de tres productos: la construcción de una primera versión totalmente funcional de un *framework* (clases, interfaces y métodos para soportar conceptos comunes del dominio), una versión de un DSL que formaliza el *framework* bajo una metodología MDA, y una versión funcional de la herramienta de transformación automática de instancias del metamodelo del DSL a código fuente. Todos los productos fueron desarrollados bajo el IDE *Visual Studio 2012*, con la extensión MDD *Dsl Tools*, para el lenguaje C#, y bajo soporte de *.NET Framework 2.0*.

La construcción del *framework* fue el primer paso de todo el trabajo. Comenzó con el análisis de soluciones transaccionales preexistentes, de la cual la compañía era dueña. Estos programas (alrededor de 10 servicios transaccionales concurrentes, o STC) estaban corriendo en ambientes productivos, contabilizando mensualmente 8 millones de transacciones en promedio. Los STC fueron desarrollados por grupos de programadores empleados por la compañía, disgregados geográficamente en todo el continente americano, sin un concepto de equipo de IT centralizado, sin control de los criterios usados para su construcción, ni del lenguaje de programación que debían usar, y ni de los mecanismos de persistencia que se debían implementar. Los equipos de desarrollo de cada país se autogobernaron por un lapso de cinco años consecutivos, haciendo que las diferencias en los códigos y en los criterios fueran cada vez más pronunciadas. Las tareas previas realizadas para converger a un único criterio, a menos lenguajes y a menos tecnologías nunca pudieron llevarse a cabo con éxito. La causa principal fue que se intentó forzar la migración de la filosofía de todos los equipos de desarrollo a la de un país, es decir eligiendo como modelo a una sola filosofía de uno de los países, sin analizar sus ventajas y desventajas, y sin considerar las ventajas de los demás. Lógicamente esto generó discusiones entre los forzados a migrar sobre los fundamentos para elegir el *criterio modelo* de un país con respecto al suyo.

Durante la fase de análisis de los códigos fuentes preexistentes encontramos una divergencia profunda de criterios de programación, falta de reusabilidad de componentes, múltiples tecnologías implementadas, metodologías informales de testing y varios bugs conocidos que no tenían solución (*memory leaks* por ejemplo; principalmente en los STC que fueron programados con C++; anecdóticamente estos eran obligados a reiniciarse cada madrugada para evitar la acumulación excesiva de memoria). Previo a iniciar nuestro trabajo, se definió que era necesario formar una capa de trabajo o *framework*, y que eventualmente ella sería la candidata a modelo de criterio único para lograr la convergencia y fomentar la reusabilidad.

La primera tarea fue la recolección iterativa de factores comunes y artefactos de software que se repetían entre los distintos STC analizados. Si bien las soluciones variaban dados sus requerimientos específicos, muchos comportamientos eran comunes. Por ejemplo, una de las primeras lecciones aprendidas de este análisis fue conocer el mecanismo de crecimiento (a nivel constructivo) de un sistema transaccional: un procesador de transacciones crece a través *del agregado de nuevas transacciones, o la modificación/sofisticación de transacciones existentes*. Fue clave entender entonces que el concepto de *Transacción* era estelar en todo el modelaje de los STC. Hasta ese entonces, muchos de los STC preexistentes estaban *orientados al flujo de las operaciones* en vez de estar *orientados al tipo de transacción*, sin distinguir de forma coherente los flujos operativos de cada una de las transacciones aceptadas por éstos. Esto causaba que al querer modificar el código de una transacción, se ponía en riesgo la lógica de las otras transacciones (bajo nivel de desacople del código). También se encontraron *God Objects* (generalmente con uno o dos métodos principales) que articulaban el orden de ejecución de todos los pasos que forman una transacción. Si una transacción se diferenciaba de otra en su comportamiento, se usaban condicionales *if-else* que a la larga complicaban las tareas de mantenimiento y/o evolución del código de forma limpia y segura. El *framework* y los STC generados de éste, deberían ser entonces *orientados a la transacción*, en vez de *al flujo de la operación*. De este modo, con la implementación de un patrón del tipo *Strategy*, se podrían definir las transacciones como unidades lógicas de construcción. Es entonces que el primer concepto que se definió en el *framework* fue la clase abstracta *transacción* (*AbstractTransactionHandler*). La clase esta compuesta por métodos que definen los pasos que deberían ejecutarse en una transacción típica. Cada vez que llega una trama de datos a los nuevos STC, se analiza el tipo de transacción, y se instancia la clase homónima correcta que represente al tipo entrante, heredando las características y los pasos comunes definidos en la clase abstracta anterior.

De forma consecutiva, fueron necesarios capitalizar en clases abstractas otros aspectos comunes. En segundo lugar se definieron los *motores de entrada*, que son los encargados de la recepción de requerimientos y del envío de respuestas (datos) hacia los POS (*Point of Sale*) que interactúan con el STC (a través de un puerto TCP). Si la trama de entrada tiene coherencia (cumple con el protocolo) y el tipo de transacción es soportado por el STC, se genera un *thread* para atender el procesamiento de la transacción, y se instancia la clase *transacción* correcta (patrón *Strategy*) según el tipo entrante. Cabe acotar que los *motores de entrada* no son una única clase abstracta, sino una jerarquía de clases abstractas ya que existen varios tipos. Consecutivamente para que los *motores de entrada* reconozcan al tipo de transacción entrante, fue

necesario definir los conceptos de dominio *parser*, *parser stream* y *parser structure*. El primer concepto almacena todas las funcionalidades para entender un protocolo transaccional (por ejemplo ISO8583), el segundo concepto es un envoltorio que almacena la trama de datos entrante o saliente de forma serializada, y el tercer concepto es una estructura de campos con la información deserializada de la trama de datos (*stream*). Los *parsers* pueden serializar y deserializar la información, es decir pueden transformar de *parser stream* (lo que llega o lo que sale por un puerto TCP) a un *parser structure* (información de contexto que nutre a la transacción durante todo su proceso), o viceversa. Por último se definieron los conceptos *contexto transaccional* (*AbstractTransactionContext*), *acceso de datos* (*AbstractTransactionDataSource*) y los *motores de salida* (que son parte de una jerarquía de clases abstractas, como en el caso de los *motores de entrada*). El *contexto transaccional* es un objeto *thread-safe* que acompaña al *thread* de la transacción durante toda su vida y sirve para almacenar datos de cualquier proveniencia que le sean de utilidad a las tareas de procesamiento. También almacena los estados por los que va incurriendo una transacción. Los *accesos a datos* permiten el acceso seguro a una base de datos, a un archivo, o a cualquier mecanismo de persistencia, con el objeto de evitar *memory leaks* e independizar al STC de la lógica de acceso hacia una tecnología de persistencia particular. Y los *motores de salida* son muy similares a los de *entrada* permitiendo comunicar al STC con otros STC, ya sea por una conexión TCP o por un protocolo SOAP (*Web Service*), WCF, XML, etc.

Luego de definir los conceptos comunes de los STC en clases abstractas, se definió en qué orden se ejecutarían los pasos que hacen a la transacción. Se propuso analizar y definir *una secuencia de trabajo genérica de procesamiento*: cada transacción debería poderse resolver en pasos, bajo una secuencia pre-ordenada, y la transacción implementa solo aquellos que sean de utilidad para realizar el correspondiente procesamiento. Es decir, que la *secuencia de trabajo genérica* tiene como objetivo que cualquier transacción pueda realizarse con pasos definidos siempre iguales, pero con posibilidad de sobrecarga para los que desee implementar. Los pasos son los siguientes: *recepción, pre-proceso, reenvío, proceso, devolución, post-proceso y mantenimiento*. La definición de esta secuencia requirió mucho trabajo de revisión de STC preexistentes, y de trabajo iterativo de corrección, de modo de conseguir este conjunto de pasos y el orden que puedan eventualmente satisfacer a múltiples STC desconocidos de antemano. Esta secuencia esta implementada en la clase abstracta *transacción*, y una clase hija solo debe sobrecargar aquellos pasos que necesite personalizar según el requerimiento dado para la transacción real. Más aún, dada la jerarquía interna de clases dentro del *framework*, varios de estos pasos pueden tener comportamientos por defecto que faciliten la construcción del STC. Este concepto de sobrecarga de pasos dentro de una jerarquía de clases, bajo una secuencia de ejecución pre-ordenada se formalizó implementando diferentes técnicas de *Inversión de Control (IoC)* y usando el patrón de diseño *Template Method*. A nivel de código fuente, cada paso se definió como un delegado (puntero a función), permitiendo conectar solo aquellos métodos concretos dentro de las subclases.

Una vez definida la *secuencia genérica de trabajo* y la mayoría de las clases abstractas del *framework*, se procedió a realizar un análisis de refactorización a patrones. De este modo se intentó sacar el mayor provecho de los patrones para que los elementos de dominio del futuro DSL puedan ser lo más sólidos posibles a nivel

arquitectura (escalabilidad), y que puedan brindar las funcionalidades planeadas de una forma eficiente y entendible para otros desarrolladores. Por ejemplo, la única forma de acceder al STC es a través de una clase *façade* a través de sus métodos *Start()* y *Stop()*. Por su parte, tanto esta clase *façade* como los *motores de entrada y salida* se definieron como clases *Singleton* ya que deben ser únicos en todo el contexto de ejecución. Esto se definió así para que todos los *threads*, es decir todas las transacciones, puedan accederlos unívocamente para comunicar requerimientos y respuestas.

Por último se propuso formalizar el conocimiento acumulado en un lenguaje de dominio específico. Utilizando las herramientas *DSL Tools* provistas por el IDE *Visual Studio*, se ha generado un metamodelo con la representación gráfica de todos estos elementos de dominio, con propiedades de dominio que permiten su personalización desde el mismo modelo. Se definieron pre-validaciones y post-validaciones por cada elemento, y las relaciones entre estos, para poder interconectarlos. Así también fue necesario desarrollar una herramienta de transformación que interpretara las instancias del metamodelo del DSL, y las transforma a código fuente. Para codificar esta herramienta se utilizó el lenguaje T4 que provee *Dsl Tools* para comprender las instancias del metamodelo. A futuro, estamos considerando poder transformar los modelos generados por este DSL en otros modelos, como por ejemplo diagramas de clase o de secuencia. La lección aprendida de esta implementación MDA-MDD fue que no es posible automatizar por completo la creación de código funcional: siempre habrá características particulares para una solución que deberán personalizarse en cada caso. Es por esto que la herramienta de transformación, genera una estructura de archivos y carpetas ordenadas por capas, de modo que una clase hija que hereda de algún elemento del *framework*, y que haya sido diseñada por el DSL, quede definida en *clases parciales* dentro de dos archivos, uno con la *componente automática* y otro con la *componente manual*. La *componente automática* estará almacenada en un archivo de código, y cada vez que se modifique la instancia del metamodelo del DSL, este archivo se modificará por la herramienta de transformación para quedar sincronizado con el modelo. En cambio, en la estructura de archivos y carpetas existirá una carpeta *CustomCode*, donde se almacenarán los archivos de código que implementan la *componente manual*. Si la *componente manual* no existe y el modelo de DSL se transforma a código, el archivo con esta componente se creará. Si el archivo ya existe no se sobrescribirá, permitiendo almacenar los cambios que hagan los programadores con las personalizaciones que no se puedan representar desde el modelo.

3 Propuesta de Implementación TDD

La metodología *Test-Driven Development* (TDD), es un enfoque evolucionario para la construcción de software, en la que se fomenta la escritura del código de los casos de prueba previo a la construcción de código productivo que cumpla con esa prueba. Esta metodología se basa en la repetición de ciclos de desarrollo de software cortos que contengan las siguientes prácticas: primero el desarrollador escribe casos de prueba automatizados (que fallan en un principio por ausencia de código funcional) y

que definen la funcionalidad o comportamiento deseado, luego se produce la mínima cantidad de código para cumplir con el caso de prueba, y luego se refactoriza el código generado para cumplir con los estándares de aceptación [7]. Como resultado de implementar TDD, los desarrolladores están más concentrados en definir la interface de pruebas que cumple con los requerimientos, que con la implementación propiamente dicha. Es decir, el desarrollo se ajusta más a un *diseño por contrato* que resulta en código más conciso, más flexible y modularizado. A su vez se reduce el acoplamiento entre clases e interfaces, haciendo al código de las mismas más limpio y puntual en pos de cumplir con los casos de unidad definidos.

Durante la fase de análisis de los STC preexistentes, se detectó que no existían políticas de pruebas de validación bien definidas. Tampoco existían casos de unidad desarrollados en el código que validen los requerimientos de estos sistemas. Favorablemente, la mayoría de las pruebas de validación que se realizaban informalmente consistían en enviar transacciones al STC, analizar los *logs* (las líneas escritas en un archivo de texto en cada paso del procesamiento) y validar las respuestas recibidas del STC. En otras palabras, es una ventaja de los STC que el único punto (o el punto principal) de interacción hacia el exterior sea a través del intercambio de transacciones. Es por ello que se propuso implementar una metodología TDD, orientada al dominio del procesamiento transaccional, basado en el factor común del intercambio de requerimientos y respuestas contra un STC.

Las instancias del metamodelo permiten definir en los *motores de entrada* la propiedad de dominio "*incluir pruebas unitarias"*. Si esta propiedad es *true* la herramienta de transformación a código fuente generará dos casos de prueba unitarios por cada *transacción* atada al *motor de entrada* en la instancia del metamodelo: un *test* para simular la aprobación de la transacción, y otro para simular el rechazo de la misma. Además, el código que implementa los casos de prueba se distribuye también entre una *componente manual* y otra *automática*. En la *componente manual*, la herramienta de transformación genera automáticamente una serie de métodos, particularmente cuatro métodos por cada transacción vinculada al *motor de entrada*. Dos de esos métodos permiten al desarrollador armar una instancia de *parser structure* de modo de simular una trama aprobada, y otra trama rechazada. El archivo que almacena esta *componente manual* se mantendrá persistente más allá de sucesivas automatizaciones de la herramienta de transformación, guardando los cambios hechos por los desarrolladores. En cambio, en la *componente automática*, se generan los métodos de prueba unitarios propiamente dichos, que pueden ser llamados desde herramientas de ejecución de *unit test*, como *NUnit*. Este archivo sí se regenerará por cada proceso de automatización de la herramienta. Las pruebas unitarias hacen llamadas a los métodos de la *componente manual* para poder serializar y enviar requerimientos al STC, y eventualmente recibir respuestas y *deserializarlas* en pos de realizar las validaciones pertinentes. Por último, las validaciones son realizadas solo si se reciben respuestas del STC. Los otros dos métodos de la *componente manual* son los encargados de realizar estas validaciones, siendo uno de ellos el agente validador para el caso de aprobación y el otro para el caso de rechazo. Cada respuesta deserializada es pasada por parámetro a uno de estos métodos (según sea el caso) como una instancia del tipo *parser structure*. Dentro de estos métodos se implementará el código que definirá si la prueba unitaria fue exitosa o fallida.

4 Evaluación de la Implementación MDD y TDD

El proyecto de I/D/I comenzó en Febrero de 2012, y durante todo ese año se realizó el análisis de STC preexistentes, la definición de los elementos comunes, la refactorización de código, y la implementación de las clases abstractas para la consecutiva construcción del marco de trabajo. Durante el primer semestre se realizó la implementación de la metodología MDD y TDD, entregando la primera versión de los productos (el marco de trabajo, el DSL y la herramienta de transformación) en Julio de 2013. Los productos se evaluaron en el ambiente corporativo, de modo de conocer si cumplía con los objetivos iniciales definidos.

Contabilizando desde el último semestre de 2013 hasta el primer semestre de 2014, se realizaron bajo esta implementación desde *scratch* 9 autorizadores STC: 6 de venta de tiempo-aire de telefonía celular, 1 para venta de vales, 1 seguros de robo y 1 de lotería. Además se analizaron 8 STC (escritos en C++ y C#) previos a la implementación dual. Más allá que los STC no son exactamente iguales, si comparamos aquellos con requerimientos similares (cantidad de transacciones parecidas, complejidad similar, etc.), se pueden obtener resultados comparativos de la evaluación entre los servidores transaccionales anteriores y posteriores a dicha implementación. Las evaluaciones son las siguientes:

- Los tiempos de desarrollo disminuyeron entre 45% y 70% en 8 de los 9 casos post-implementación (siendo la excepción el autorizador de vales). Para este último caso, el *motor de salida* debía cumplir con una serie de requisitos imprevistos en la primera versión del *framework*, por lo que se definió un nuevo tipo de *motor de salida*, se implementó, se integró a este *framework* y se representó en el DSL. Ese *costo* puede considerarse *inicial*, dado que para próximas soluciones ya estará listo para ser reutilizado.
- Se logró homologar los conceptos de los desarrolladores en los equipos distribuidos de trabajo. De pasar a trabajar de forma independiente, se logró centralizar el equipo y ahora todos colaboran desarrollando STC para países diferentes al de locación de cada individuo.
- La tasa de errores durante los primeros tres meses (luego de la puesta en marcha) de cada STC, también se redujo. Las incidencias registradas para aquellos STC desarrollados previos al 2013 denotan que fueron necesarios cambios *en caliente* en la mayoría de los casos, al corto plazo de haberse puesto cada STC en producción. En los STC desarrollados post-implementación, hubo veces que fue necesario hacer cambios sobre la marcha al corto plazo pero principalmente por errores de comprensión de un requerimiento en particular. Los errores de código dentro de un requerimiento comprendido se han disminuido casi en su totalidad, dado que los elementos de dominio han sido probados en otras soluciones con anterioridad.
- Se logró aumentar la visibilidad del código: A partir de la implementación realizada, se han hecho reuniones retrospectivas de revisión de arquitectura trimestrales. Tanto los desarrolladores, como los líderes de

equipo de cada país están al tanto de la arquitectura de los STC en producción.

También se encontraron puntos débiles en los STC construidos bajo esta propuesta MDD/TDD. En primer lugar, la implementación del DSL y de la herramienta de transformación está atada en esta primera versión a la tecnología *.NET Framework* de *Microsoft.* Del mismo modo, las clases para accesos a base de datos fueron diseñadas para trabajar contra un motor de DB *SQLServer,* por lo que en la versión actual no hay compatibilidad con otros motores tales como *Oracle*. Desde el lado TDD, la implementación tuvo éxito aunque se aprovechó solo la capacidad de validación de la metodología (con las pruebas unitarias solo se validó el código de las *componentes automáticas y manuales*). Aún no se conjugó la filosofía TFD (*Test-First Development*), que podría ser de utilidad, por ejemplo, para la construcción de la *componente manual*.

5 Trabajos Relacionados

Del conjunto de artículos recolectados durante la primera fase del proyecto, se destacan estos tres trabajos:

El primer trabajo (*Multi-threading technique for authorization of credit card system using .NET and JAVA*) del autor W. Y. Ming [1] propone un STC orientado a la paralelización de tareas (*multi-threading*) para mejorar los tiempos de respuesta en autorizadores financieros. El análisis del autor recae en qué tareas típicas de estos STC pueden paralelizarse dentro de cada hilo de ejecución de cada transacción, por ejemplo, las operaciones criptográficas. Muchas de estas operaciones se hacen con dispositivos de E/S (un HSM conectado al servidor), y la interacción contra estos puede llegar a ser costosa en tiempos de respuesta. Este STC es realizado tanto en plataformas *Java* como en *.NET Framework*. Sin embargo no trabaja sobre la formalización de esta técnica *multithreading* en elementos de dominio reutilizables.

El segundo trabajo (*Lagniappe: Multi-* programming made simple*) [6] hace una propuesta de formalización MDD de un conjunto de elementos de dominio sobre arquitecturas *multi-procesador* o *multi-threading* (*multi-**). El artículo está orientado particularmente a sistemas de alto tráfico como enrutadores GENI, sistemas de consola y sistemas transaccionales en general. No define elementos de dominio *complejos* (como se hizo en nuestro artículo, de modo de representar conceptos comunes), en cambio solo define unos elementos de dominio *reducidos*, que pueden conectarse entre sí para formar elementos de dominio más complejos.

El tercer trabajo (*Domain-Specific Languages for Enterprise Systems*) [8] analiza una arquitectura llamada POETS, para la construcción orientada a eventos en sistemas del tipo ERP. En este artículo se hace una propuesta de modelización MDD sobre POETS, proponiendo una serie de elementos de dominio. El trabajo describe un caso de uso del DSL implementado a modo de evaluación. La forma de exposición de este trabajo (dominio, DSL y evaluación) sirvió como modelo para nuestro artículo.

6 Conclusiones y Aportes

Este artículo presentó un análisis de elementos comunes para el dominio de sistemas transaccionales concurrentes (STC). A partir de este análisis se realizó la organización de todos los conceptos recolectados en una jerarquía de clases abstractas dentro de un *framework*. A su vez este *framework* fue la base para la construcción de un DSL que permita la implementación de una metodología MDD. Por último se desarrolló una herramienta de transformación de instancias del meta-modelo a código funcional, y se integró parcialmente una técnica de casos de unidad (TDD) orientada al dominio bajo análisis.

El aporte de este trabajo es la definición del conjunto de elementos de dominio funcionales basados en factores comunes analizados de diversos STC. El DSL pretende ser útil para la construcción de otros STC, cualquiera fuera el tipo. Bajo la premisa que los elementos de dominio aquí analizados deberían encontrarse en cualquier STC, nuestra visión a futuro es que estos faciliten el diseño de otros servidores transaccionales, y que otros trabajos sobre el tema puedan extender el DSL con nuevos elementos de dominio aun no definidos. Así también se podrían rediseñar aquellos aquí presentados para que representen la realidad de forma más precisa, o funcionen de mejor forma. Por último la implementación TDD aumenta la confiabilidad del sistema a través de la automatización de pruebas unitarias de validación. En este trabajo se aporta un enfoque para implementar las pruebas unitarias como la combinación de transacciones y sus resultados esperados, operando al STC como un objeto cerrado (*black box*) que recibe *entradas* y devuelve *salidas*.

Referencias

1. Ming, W. Y.: Multi-threading technique for authorization of credit card system using .NET and JAVA, Kuala Lumpur, 2007.
2. Kang, K., Lee, J., Kim, B., Kim, M., Seo, C., Yu, S.: Re-engineering a credit card authorization system for maintainability and reusability of components–a case study. In: Reuse of Off-the-Shelf Components, pp. 156--169. Springer Berlin Heidelberg (2006)
3. Pons, C., Giandini, R., Perez, G.: Desarrollo de software dirigido por modelos. Conceptos teóricos y su aplicación práctica. EDULP, La Plata, Argentina (2010)
4. Singh, Y., Sood, M.: Models and Transformations in MDA. In: First International Conference on Computational Intelligence, Communication Systems and Networks. (2009)
5. Bishop, J.: CSharp 3.0 Design Patterns. O' Reilly (2008)
6. Riché, T.L., Lavender, G., Vin, H.M.: Lagniappe: Multi-* programming made simple. Proceedings - International Conference on Computer Communications and Networks, ICCCN, art. no. 4317815, pp. 173-178. (2007)
7. Kent, B.: Test-Driven Development by example. Addison Wesley (2003)
8. Andersen, J.B. , Bahr, P.A. , Henglein, F.A. , Hvitved, T.A.: Domain-specific languages for enterprise systems - Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8802, pp. 73-95. (2014)

Umbrales sugeridos para promedios de métricas de diseño de una aplicación en Java

Jorge Ramirez, Gustavo Gil, Carina Reyes

Universidad Nacional de Salta, Facultad de Ciencias Exactas, Centro de Investigación y Desarrollo en Informática Aplicada (CIDIA)
{ramirezj, gdgil,reyescarina}@cidia.unsa.edu.ar

Resumen. Las métricas de producto de software pueden aportar información relevante para la gestión de proyectos de software. La determinación de umbrales para esas métricas permitiría disponer de información rápida que advierta sobre potenciales problemas en los productos bajo análisis. Sin embargo, no existe amplio consenso en cuanto a estos indicadores. El Software Libre y de Código Abierto (F/OSS por sus siglas en inglés) ofrece la posibilidad de estudiar grandes cantidades de versiones a partir de la libre disponibilidad del código fuente que caracteriza a las licencias bajo las cuales se distribuyen esos productos. Este trabajo propone umbrales para métricas a partir del estudio de las distribuciones de frecuencia del promedio de la cantidad de métodos por clase y de la proporción de referencias a métodos de otras clases respecto del total de métodos definidos en la aplicación.

Palabras Clave: Métricas de Diseño, Software Libre y de Codigo Abierto. Umbrales para métricas.

1 Introducción

La caracterización de una pieza de software con miras a evaluar su potencial adopción o reutilización, requiere de disponer de elementos de juicio suficientes para la toma de decisiones. En particular, al momento de seleccionar un producto para incorporarlo a un nuevo desarrollo, es necesario contar con indicadores que orienten esta determinación.

No existe consenso general en la ingeniería de Software acerca de la vinculación entre métricas y los distintos atributos del software [1]; un ejemplo en este sentido es el estudio empírico presentado por Sjoberg y otros [2] que revela que varias métricas asociadas habitualmente a la mantenibilidad (el índice de mantenibilidad, distintas medidas de tamaño y cohesión) son inconsistentes entre sí. Fowler [3] considera que ninguna combinación de métricas puede rivalizar con la intuición humana informada; compartiendo en general ese punto de vista, no se busca postular aquí relaciones causales sino indicadores que actúen como advertencia de posibles debilidades en el diseño. Lanza y Marinescu [4], en tanto, señalaron que no existe un umbral perfecto, cuya superación signifique necesariamente un cambio cualitativo en el atributo que se pretende evaluar, pero pueden ser indicativos de utilidad en la práctica.

Desde esta perspectiva se aborda la posible determinación de umbrales de métricas de software, es decir, valores que al superarse alerten sobre posibles debilidades en el diseño o sobre riesgo potencial respecto de la mantenibilidad de código.

Para ello en este trabajo se analizan las distribuciones de frecuencias de un conjunto de métricas sobre una muestra de versiones de aplicaciones F/OSS escritas en Java pertenecientes a proyectos que han publicado 7 versiones o más, lo que refleja la evolución de estos proyectos.

El enfoque adoptado aquí se diferencia del que sigue la gran mayoría de los estudios que proponen umbrales, dado que se buscan elementos para evaluar el diseño en su conjunto mediante valores promedio en lugar de determinar valores que sirvan de referencia para módulos o clases particulares.

En concreto, se plantea detectar qué métricas referidas al diseño de una aplicación muestran una distribución de frecuencias tal que sugiera la conveniencia de adoptar determinados umbrales. El trabajo se organiza de la siguiente manera: en el apartado siguiente se ofrece una revisión de trabajos similares; a continuación, se describe la metodología utilizada en la investigación; posteriormente se analizan los datos obtenidos y se proponen umbrales para algunas métricas; a continuación, se comparan estos valores con los sugeridos en otros trabajos. Finalmente, en la sección 5 se exponen las conclusiones y los trabajos futuros.

2 Trabajos Relacionados

Se han propuesto diversos umbrales para métricas de diseño orientado a objetos definidas mediante diversos métodos. Por un lado, algunos trabajos utilizan técnicas estadísticas que intentan relacionar las mediciones obtenidas con atributos asociados a la mantenibilidad o a la propensión a errores de alguna parte de código o un elemento de diseño [5] [6] [7] [8] . Otros, en cambio, se basan en la distribución de los valores de las métricas, buscando revelar cuáles serían cifras atípicas para los indicadores que consideran [9] [10] [11] [4] [12] .

En la mayoría de los estudios las métricas que se abordan son las propuestas por Chidamber y Kemerer [13], y Brito e Abreu y Carapuça [14] entre otras de uso común.

2.1 Estudio de Umbrales en Base a Correlaciones Estadísticas

Los trabajos que se resumen a continuación apuntan a determinar valores de las métricas que pudieran advertir sobre debilidades en un módulo o una clase en concreto.

Benlarbi y otros [5] plantearon la determinación de umbrales para los valores de las métricas orientadas a objetos. En el estudio consideran a los umbrales como valores heurísticos que sirven para determinar rangos de valores deseables y no deseables para métricas de software. Los autores analizaron las métricas propuestas por Chidamber y Kemerer sobre dos sistemas de telecomunicaciones escritos en C++ en los que clasificaron a cada clase como “defectuosa” o “no defectuosa” según si se había reportado al menos una falla durante la operación del sistema.

Los autores concluyeron que el establecimiento de umbrales no aporta ventajas respecto de un modelo continuo.

Shatnawi realizó diversos estudios buscando determinar umbrales a partir de diferentes análisis estadísticos. En uno de ellos [6], emplea la curva ROC para proponer umbrales de un conjunto de métricas CK y de Lorentz. Ese método permite evaluar la pertinencia de un umbral para clasificar en dos categorías (en este caso, en defectuoso y no defectuoso). En un trabajo posterior, el mismo autor [7] empleó regresión logística (como hiciera anteriormente Benlarbi) para estudiar valores aceptables para un conjunto de métricas; los resultados por fuera de esos valores indicarían un mayor riesgo de presencia de defectos. Este autor analizó diferentes versiones del popular IDE Eclipse, realizando posteriormente una comparación con otros dos productos (Rhino y Mozilla); concluye que no pueden generalizarse las cifras postuladas en el primer caso.

El autor también compara los umbrales calculados con los propuestos por Rosenberg [9], concluyendo que éstos son menos seguros para estimar clases defectuosas.

Kaur [8] y otros obtuvieron las métricas CK y otras indicativas de posibles "bad smells" o antipatrones [15] sobre dos versiones del software JFreeChart. Emplearon regresión logística, estimando posibles umbrales en base al Nivel Aceptable de Riesgo (Value for Acceptable Risk Level, VAR) de acuerdo a la metodología elaborada por Bender [16] para analizar posibles "efectos de umbral". De este resumen se desprende que los objetivos buscados principalmente por los trabajos relevados se relacionan con la determinación de métricas que permitan caracterizar piezas individuales de software, más que caracterizar el diseño de las aplicaciones en su conjunto.

2.2 Estudio de Umbrales en Base a Distribuciones de Frecuencia de las Métricas

Rosenberg [9] utilizó histogramas para analizar el efecto de los valores de las métricas de Chidamber y Kemerer [13] sobre la calidad del software. Los valores de los umbrales se basan en el análisis de los histogramas, aunque no se analiza la posible relación entre esos valores y la probabilidad de que módulos presenten errores o deban modificarse.

Erni y Lewerentz [10] analizaron la distribución de métricas orientadas a objetos postulando como potenciales umbrales a la media ± la desviación estándar. Los autores plantean la adopción de uno de esos valores en función de los problemas que plantearía en cada caso cifras demasiado altas o demasiado bajas. Por ejemplo, si para una métrica se considera que un número alto es perjudicial, se adopta $\mu+\sigma$ como umbral (siendo μ la media y σ la desviación estándar).

Chidamber y otros [11] consideraron las distribuciones empíricas de las métricas sugiriendo que el tope para definir un valor como "alto" sería el 80º percentil (es decir, el 80% de las muestras alcanzarían valores por debajo del umbral señalado).

Lanza y Marinescu [4] también propusieron umbrales para métricas comunes partiendo de la distribución de los valores en un grupo de 45 aplicaciones escritas en Java. Clasificaron las medidas según sus frecuencias en bajo, medio, alto y muy alto.

Ferreira y otros [12] estudiaron las distribuciones de las métricas COF (Factor de Acoplamiento), cantidad de métodos públicos, cantidad de campos (atributos) públicos, LCOM (Falta de Cohesión en los métodos) y DIT (Profundidad del árbol de herencia)- Analizaron 40 aplicaciones escritas en Java, computando las métricas con

la herramienta Connecta, y las distribuciones de frecuencia se ajustaron a distribuciones teóricas con la herramienta EasyFit. Los autores dividieron la muestra según diferentes dominios, planteando umbrales distintos para cada uno de ellos.

Recientemente Filó y otros [17] presentaron un estudio de diversas métricas a partir de distribuciones de frecuencia encontradas en una muestra de 111 aplicaciones escritas en Java. A partir de ese análisis proponen umbrales para 17 métricas diferentes. El trabajo muestra similitudes con el que se expone aquí dado que se basa en una muestra públicamente accesible y emplea el análisis de distribución de frecuencia como metodología para detectar posibles umbrales; no obstante, el objetivo de ese estudio es proponer valores referidos a debilidades de clases o paquetes concretos, más que obtener indicadores referidos el diseño en su conjunto.

3 Metodología

Para este trabajo se tuvo en cuenta la infraestructura de investigación propuesta por Gaser y Scacchi [18] para los estudios empíricos sobre F/OSS. A fin de posibilitar la repetibilidad y revisión del trabajo, se utilizaron versiones públicamente accesibles que incluyan su código fuente; las herramientas empleadas siguen el mismo criterio: las mediciones se realizaron con iPlasma [19] y el tratamiento estadístico se efectuó con el entorno R [20].

Se descargó el código fuente de todas las versiones a analizar y se obtuvieron las métricas que se detallan en la tabla 1.

La selección de las métricas apunta a una visión global del software a estudiar; indagando en las características generales del código fuente. Los proyectos analizados tienen en común el haber producido numerosas versiones y que han sido utilizados de manera efectiva por muchos usuarios. Son proyectos "exitosos" en los términos de Crowston [21] y Weiss [22].

Por ello se eligieron las medidas propuestas por Lanza y Marinescu [4], quienes plantean una perspectiva de conjunto para caracterizar el diseño de una aplicación.

Tabla 1. Métricas computadas en el presente trabajo

Métrica	Descripción
CYCLO	Complejidad ciclomática, según la definición de McCabe[22]
LOC	Líneas de código, incluyendo líneas en blanco y comentarios
NOM	Cantidad de métodos y otras operaciones definidas en toda la aplicación
NOC	Cantidad de clases
NOP	Cantidad de paquetes
CALL	Cantidad de invocaciones a operaciones
FOUT	Suma de las clases referenciadas por cada una de las clases

Cada una de estas métricas arroja un valor global dependiente del tamaño del software. Es esperable que una aplicación más grande tenga un mayor número de

clases, mayor cantidad de líneas de código, etc. Lanza y Marinescu plantean entonces la caracterización a partir de las proporciones que se muestran en la tabla 2.

Tabla 2. Proporciones analizadas

Nombre	Descripción	Fórmula
Estructuración de alto nivel	Cantidad de clases por paquete	NOC/NOP
Estructuración de clases	Cantidad de operaciones en relación con le número de clases	NOM/NOC
Estructuración de operaciones	Promedio de la cantidad de líneas de código por operación	LOC/NOM
Complejidad de operación intrínseca	Cantidad de caminos independientes en relación con el tamaño en líneas de código	CYCLO/LOC
Intensidad de acoplamiento	Proporción de los métodos que invocan a cada operación en relación con la cantidad total de operaciones	CALL/NOM
Dispersión de acoplamiento	Proporción de las llamadas de cada clase a métodos de otra respecto del total de llamadas	FOUT/CALL

En este trabajo se computan las métricas mencionadas para 560 versiones de 28 proyectos diferentes escritos en Java. Si bien iPlasma no extrae WMC (métodos ponderados por clase) [13], se puede obtener a partir de la Complejidad Intrínseca, la Estructuración de Operaciones y la Estructuración de Clases. Analizamos la distribución de WMC dada de su amplia utilización.

En total, se computaron 371.934 clases que incluyen 3.907.564 métodos, cubriendo 43.842.838 líneas de código.

4 Análisis de los Datos

La tabla 3 resume las medidas de localización y dispersión [23] para las proporciones estudiadas:

Tabla 3. Medidas localización y dispersión para las métricas computadas

	1º Cuartil	Media	Mediana	3º Cuartil	Desviación estándar	Coef. de Variación	Coef. de asimetría
Complejidad Intrínseca	0,1727	0,2076	0,2053	9,2236	0,041	0,13	0,87
Estructuración de Operaciones	7,973	10,27	10,86	14,310	4,277	0,41	0,89
Estructuración de Clases	9,581	10,91	11,74	13,33	3,453	0,29	0,93
Estructuración de Alto Nivel	7,503	9,98	12,55	17,81	6,965	0,7	1,41
Intensidad de Acoplamiento	0,556	0,588	0,585	0.616	0,168	0,29	17,63
Dispersión de acoplamiento	2,534	2,992	3,024	3,399	0,82	0,27	0,13

Las distribuciones de estas proporciones muestran que la Complejidad Intrínseca, la Intensidad de Acoplamiento, la Estructuración de Clases, y la Dispersión de acoplamiento exhiben menor dispersión. De ellas, la Intensidad de acoplamiento presenta una distribución de frecuencias con una asimetría más marcada.

Se analizan a continuación las gráficas de distribución que se muestran en las figuras 1 a 4; posteriormente, se considera la variación entre deciles, a fin de observar en cuál de ellos se constata mayor distancia entre los valores.

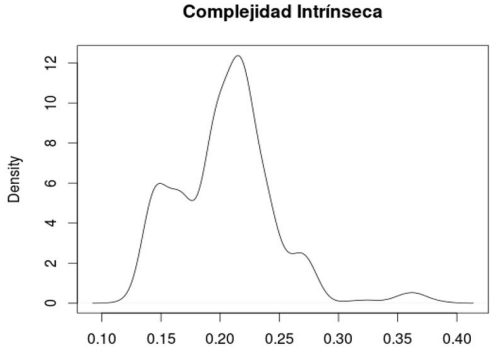


Fig. 1. Distribución de la Complejidad Intrínseca.

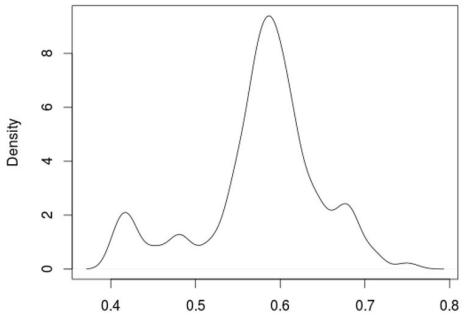


Fig. *2*. *Distribución de la Intensidad de acoplamiento*

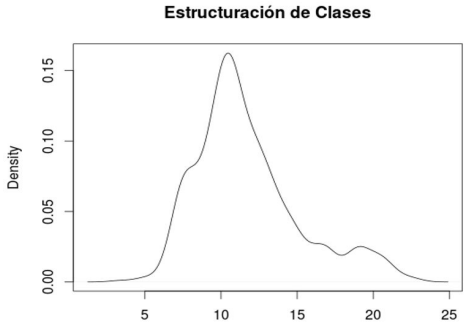


Fig. *3*. *Distribución de la Estructuración de Clases*

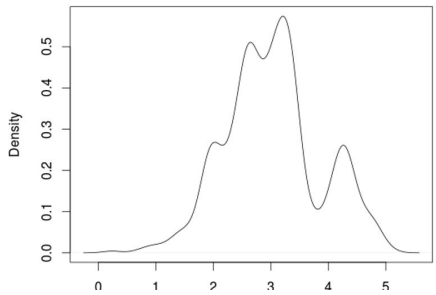


Fig. *4.* *Distribución de la Dispersión de acoplamiento*

Respecto de la Estructuración de clases, la gráfica muestra una alta concentración en torno a la moda. En tanto, la Estructuración de Operaciones exhibe varios picos, con una caída pronunciada en las frecuencias de valores mayores a 20.

Las distribuciones de frecuencias de la Estructuración de Clases y de la Intensidad de Acoplamiento muestran claramente una zona modal a partir de las cuales las frecuencias decrecen en ambos sentidos. Analizando la variación de los deciles podemos proponer umbrales superiores buscando cambios pronunciados entre los valores sucesivos de los mismos, lo que implicaría un descenso rápido de las frecuencias. La tabla 4 resume los deciles para las distribuciones de las métricas consideradas.

Tabla 4. Valores que correspoden a los diferentes deciles para las distribucionesobservadas en las proporciones analizadas.

Decil	C.I.	E.C.	E.O.	D.A.	I.A.	WMC
"0%"	0,121	3,375	3,832	0,234	0,406	7,346
"10%"	0,148	7,721	5,707	1,968	0,468	9,774
"20%"	0,166	9,001	6,876	2,372	0,544	16,233
"30%"	0,191	9,950	8,218	2,603	0,567	18,103
"40%"	0,196	10,458	9,425	2,750	0,575	20,463
"50%"	0,208	10,905	10,272	2,992	0,588	22,000
"60%"	0,216	11,730	11,301	3,172	0,594	23,772
"70%"	0,221	12,639	13,622	3,337	0,607	26,589
"80%"	0,233	14,092	14,820	3,665	0,625	36,062
"90%"	0,251	16,936	15,830	4,270	0,668	45,683
"100%"	0,385	22,773	29,321	5,084	0,759	182

En la Fig. 5 Observamos también la distribución de WMC, calculada a partir de otras métricas según se señaló más arriba.

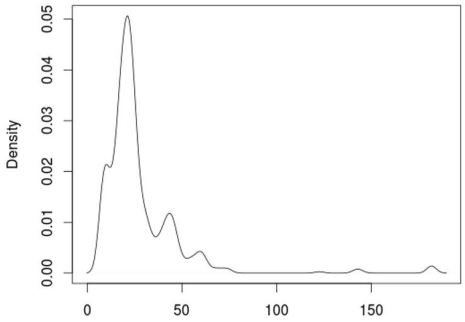


Fig. 5. *Distribución de WMC (Métodos ponderados por clase)*

Respecto de la estructuración de clases, se observa que entre el tercer y el octavo decil (60% de los casos) tienen en promedio entre 9 y 14,09 métodos por clase en promedio. Por otra parte, analizando el histograma (sesgado hacia la izquierda) se observa que la frecuencia de los valores superiores a 13 decrece significativamente, lo que convierte a ese valor en un posible umbral.

En cuanto a la Intensidad de acoplamiento, el decrecimiento más pronunciado comienza con el noveno decil , por lo que un valor superior a 0,668 podría reflejar acoplamiento excesivo.

Si contemplamos WMC, vemos que la curva de distribución de frecuencias tiene un máximo posterior a la moda; recién luego del 9º decil se verifica un descenso constante, por lo que el umbral para esta métrica sería de 45,68.

4.1 Umbrales Propuestos en Otros Trabajos

La tabla 5 muestra los valores propuestos en diversos trabajos para la métrica WMC

Tabla 5. Valores propuestos en otros trabajos como umbral para WMC.

Métricas	Shatnawi [7]	Rosenberg [9]	Lanza y Marinescu [4]	Herbold et al [24]
WMC	20	100	47	100

Como puede observarse, las cifras postuladas son muy disímiles. Cabe señalar, no obstante, que el valor planteado por Lanza y Marinescu es similar al obtenido en este trabajo sobre una muestra mucho mayor.

5 Conclusiones y Trabajos Futuros

Este trabajo adopta dos enfoques novedosos respecto de la búsqueda de umbrales útiles de métricas de diseño orientado a objetos. Por un lado, se buscan referencias para valores promedio, indicativos del diseño general de una aplicación o de un componente de software, en lugar de proponer referencias para clases particulares o módulos particulares.

Por otra parte, se retoma el análisis de distribuciones de frecuencia de los valores de las métricas, buscando topes identificables a partir de las curvas de distribución de frecuencias.

En concreto, una aplicación que presente más de 14 métodos por clase, o un promedio de WMC por encima de 46, muestra una estructuración muy poco común, con clases con demasiados métodos; de esto no se concluye que tal o cual clase se potencialmente defectuosa, sino que el diseño en su conjunto tiende a agrupar demasiadas funciones sobre las clases.

Si bien la métrica compuesta denominada Intensidad de Acoplamiento muestra un comportamiento que sugiere un tope viable, es una medida que pocas herramientas computan. No obstante, puede advertir que el diseño tiende a que las clases requieran de muchas otras para realizar sus tareas.

Teniendo en cuenta estos aspectos, sería conveniente adoptar la métrica Estructuración de Clases (promedio de la cantidad de métodos por clase) por su sencillez, o WMC, que es captada por una amplia cantidad de herramientas. Cabe señalar, sin embargo, que al basarse los umbrales propuestos en valores promedio, éstos pueden no ser relevantes para aplicaciones pequeñas, dadas las características del promedio como medida de tendencia central influenciada por valores extremos.

En futuros trabajos se prevé analizar otras métricas relacionadas con el acoplamiento, y estudiar su comportamiento con relación a las distintas etapas de la evolución de una aplicación. De esta forma, esperamos conocer diversos aspectos de la evolución del software y generar más elementos para una evaluación preliminar de una aplicación F/OSS a partir de la información que se puede obtener de su código fuente.

6 Referencias

1. Bourque, P., Wolff, S., Dupuis, R., Sellami, A., Abran, A.: Lack of Consensus on Measurement in Software Engineering: Investigation of Related Issues. In: 14th International Workshop on Software Measurement (IWSM) IWSM-Metrikon 2004, Konigs Wusterhausen. pp. 321–333. Shaker-Verlag, Magdeburg, Germany (2004).
2. Sjøberg, D.I.K., Anda, B., Mockus, A.: Questioning Software Maintenance Metrics: A Comparative Case Study. In: Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement. pp. 107–110. ACM, New York, NY, USA (2012).
3. Fowler, M., Beck, K., Brant, J., Opdyke, W., Roberts, D.: Refactoring: Improving the Design of Existing Code. Addison-Wesley Professional (1999).
4. Lanza, M., Marinescu, R., Ducasse, S.: Object-Oriented Metrics in Practice. Springer-Verlag New York, Inc., Secaucus, NJ, USA (2005).
5. Benlarbi, S., Emam, K.E., Goel, N., Rai, S.N.: Thresholds for Object-Oriented Measures. In: ISSRE. pp. 24–39. IEEE Computer Society (2005).
6. Shatnawi, R., Li, W., Swain, J., Newman, T.: Finding software metrics threshold values using ROC curves. J. Softw. Maint. Evol. Res. Pract. 22, 1–16 (2010).
7. Shatnawi, R.: A Quantitative Investigation of the Acceptable Risk Levels of Object-Oriented Metrics in Open-Source Systems. IEEE Trans. Softw. Eng. 99, 216–225 (2010).

8. Kaur, S., Singh, S., Kaur, H.: A Quantitative Investigation Of Software Metrics Threshold Values At Acceptable Risk Level. Int. J. Eng. Res. Technol. 2, (2013).
9. Rosenberg, L.: Applying and Interpreting Object Oriented Metrics. Presented at the Software Technology Conference (1998).
10. Erni, K., Lewerentz, C.: Applying Design-metrics to Object-oriented Frameworks. In: Proceedings of the 3rd International Symposium on Software Metrics: From Measurement to Empirical Results. p. 64–. IEEE Computer Society, Washington, DC, USA (1996).
11. Chidamber, S.R., Darcy, D.P., Kemerer, C.F.: Managerial Use of Metrics for Object-Oriented Software: An Exploratory Analysis. IEEE Trans Softw Eng. 24, 629–639 (1998).
12. Ferreira, K.A.M., Bigonha, M.A.S., Bigonha, R.S., Mendes, L.F.O., Almeida, H.C.: Identifying Thresholds for Object-oriented Software Metrics. J Syst Softw. 85, 244–257 (2012).
13. Chidamber, S.R., Kemerer, C.F.: A Metrics Suite for Object Oriented Design. IEEE Trans Softw Eng. 20, 476–493 (1994).
14. Brito e Abreu, F., Carapuca, R.: Object-Oriented Software Engineering: Measuring and Controlling the Development Process. In: Proc. Int'l Conf. Software Quality (QSIC) (1994).
15. Fowler, M., Beck, K.: Bad Smells in Code. In: Refactoring: Improving the design of existing code. pp. 75–88. Addison-Wesley (1999).
16. Bender, R.: Quantitative Risk Assessment in Epidemiological Studies Investigating Threshold Effects. 41, 305–319 (1999).
17. Filó, T., Bigonha, M., Ferreira, K.A.M.: A Catalogue of Thresholds for Object-Oriented Software Metrics. In: SOFTENG 2015. pp. 48–55 (2015).
18. Gasser, L., Scacchi, W.: Towards a Global Research Infrastructure for Multidisciplinary Study of Free/Open Source Software Development. In: Russo, B., Damiani, E., Hissan, S., Lundell, B., and Succi, G. (eds.) Open Source Development, Communities and Quality. pp. 143–158. Springer, Boston (2008).
19. Marinescu, C., Marinescu, R., Mihancea, P.F., Ratiu, D., Wettel, R.: iPlasma: An Integrated Platform for Quality Assessment of Object-Oriented Design. In: ICSM (Industrial and Tool Volume). pp. 77–80 (2005).
20. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2013).
21. Crowston, K., Annabi, H., Howison, J.: Defining Open Source Software Project Success. In: in Proceedings of the 24th International Conference on Information Systems (ICIS 2003). pp. 327–340 (2003).
22. Weiss, D.: Measuring success of open source projects using web search engines. In: First International Conference on Open Source Systems (2005).
23. Devore, J.L.: Probabilidad y estadística para Ingeniería y ciencias. CENGAGE Learning (2005).
24. Herbold, S., Grabowski, J., Waack, S.: Calculation and Optimization of Thresholds for Sets of Software Metrics. Empir. Softw Engg. 16, 812–841 (2011).1. Bourque, P., Wolff, S., Dupuis, R., Sellami, A., Abran, A.: Lack of Consensus on Measurement in Software Engineering: Investigation of Related Issues. In: 14th International Workshop on Software Measurement (IWSM) IWSM-Metrikon 2004, Konigs Wusterhausen. pp. 321–333. Shaker-Verlag, Magdeburg, Germany (2004).

Ejecución y monitorización distribuida de procesos de negocios: una implementación con Bonita Open Solution

Leonardo Karabogosian[1], Patricia Bazán[1], Jose Martinez Garro[1]
[1] Facultad de Informática – Universidad Nacional de La Plata,
50 y 120 – La Plata
leonardo.karabogosian@gmail.com, pbaz@info.unlp.edu.ar,
josemartinezgarro@yahoo.com.ar

Abstract. BPM *(Business Process Management*) ha ido creciendo en relevancia dentro de las organizaciones y ha madurado tanto desde el punto de vista metodológico y como de la tecnología que la sustenta. Por otra parte, el avance tecnológico tanto en términos de comunicación como en poder de cómputo, han hecho que *Cloud Computing* sea una opción potencial para la reducción de costos y mejoras en el procesamiento. La combinación de las tecnologías asociadas a *Cloud Computing* y BPM modifica aspectos tanto de diseño como de ejecución de los procesos de negocios. Los ambientes distribuidos en el contexto de los procesos favorecen el rendimiento y proponen incorporar el concepto de descomposición de procesos, permitiendo que los mismos se ejecuten tanto en un entorno cloud como en uno embebido. Si bien la descomposición de procesos es un tema abordado en los últimos años, la monitorización de dichos procesos no ha sido demasiado explorado aún. Este trabajo propone una implementación de una arquitectura para un sistema de monitorización de procesos distribuidos utilizando Bonita Open Solution como motor de procesos, su API y el uso de conectores personalizados.

Keywords: BPM, BPMS, cloud computing, monitorización de procesos, descomposición de procesos, Bonita OS

1. Introducción

Las organizaciones actuales apuestan fuertemente a contar con procesos de negocio cada vez más eficientes y eficaces. La utilización de Sistemas de Gestión de Procesos de Negocio (BPMS - *Business Process Management Systems*) para identificar y diseñar, ejecutar, monitorear y optimizar los procesos de negocio ha sido de gran ayuda para poder disminuir los costos, aumentar la productividad, mejorar los servicios a los clientes, crear un marco de organización y coordinación de actividades para el personal de la empresa u organización.
Otra de las tecnologías que actualmente está teniendo un interés masivo en las empresas y organizaciones es la computación en la nube o *Cloud Computing*. Con este nuevo paradigma, las empresas buscan proporcionar servicios de computación bajo demanda con una alta fiabilidad, escalabilidad y disponibilidad en un entorno

distribuido, donde el poder de cómputo, almacenamiento de datos y servicios se subcontratan a terceros y se ponen a disposición de las empresas y clientes, lo que ocasiona que los riesgos económicos y técnicos disminuyan.
Indefectiblemente, hace unos años, estas dos potentes herramientas en el área de IT, hicieron que se comience con el estudio e implementación de BPMS en la "nube". Esta combinación de técnicas de *clouding* y BPM ofrece un enfoque flexible y ágil, junto con las ventajas de ambos paradigmas. La gran capacidad computacional de los sistemas en el *cloud* y el "pago por uso" en lugar de enfrentar grandes inversiones en software y hardware, son dos grandes ventajas de la combinación de estas técnicas. Sin embargo, al utilizar un BPMS en la nube, se pierde el control sobre los datos sensibles del negocio, lo que conlleva a tomar un riesgo muy grande para las empresas de hoy en día.
Actualmente podemos encontrar BPMS que se encuentren en servidores locales o en servidores localizados en el *cloud* brindando Software como Servicio (SaaS – *Software as a Service*). La descomposición de procesos de negocios antes mencionada, permite que un proceso se ejecute en ambos ambientes, dependiendo de la lógica y modelo adoptado por el desarrollador del proceso al momento de la descomposición. Sin embargo, una vez que se realiza la descomposición del proceso, la monitorización y seguimiento del proceso de negocio original en este sistema híbrido se dificulta, haciendo que se deba recolectar toda la información de cada uno de los servidores en donde se encuentran las partes del proceso.
Si bien el estudio de la descomposición de procesos de negocios se encuentra en una etapa bastante avanzada, y podemos encontrar mucha información relacionada, el estudio de la integración de estos procesos descompuestos para la monitorización y seguimiento de los procesos de negocio distribuidos es un campo aún poco explorado.
La motivación de este trabajo se basa en favorecer el seguimiento de procesos descompuestos en un sistema distribuido y cubrir de este modo la etapa de monitorización y optimización del ciclo de vida de los procesos de negocios. Para ello se propone una implementación de una arquitectura para un sistema de monitorización de procesos distribuidos utilizando *Bonita Open Solution* [6] como motor de procesos, su API y el uso de conectores personalizados.
El sistema de monitorización se basará en las siguientes etapas:

- Implementación de un conector en Bonita OS para la ejecución de un proceso de negocio descompuesto entre distintos servidores distribuidos.
- Investigación de la API REST de Bonita Open Solution para la instanciación de procesos remotos y el acceso a las propiedades de procesos ya instanciados y desplegados en el motor remoto.
- Confección de una base de datos local para la persistencia de la información relacionada a los procesos de negocios instanciados de forma local junto a la relación que tiene con los procesos de negocio remotos.
- Implementación de una aplicación Web para monitorización y seguimiento de procesos de negocios distribuidos. Utilización de Web Services y API REST para la integración de los procesos descompuestos y distribuidos, y la visualización unificada del proceso global para el correcto seguimiento del proceso de negocio.

Este trabajo se organiza de la siguiente manera: en la Sección 2 se plantea el enfoque de BPMS en un entorno distribuid. En la sección 3 se introducen conceptos de descomposición de procesos. En la Sección 4 se abordan aspectos propios de la

monitorización distribuida de procesos. En la Sección 5 se presenta la solución propuesta y su aplicación a un caso de estudio en la Sección 6. Finalmente, en la Sección 7 se describen las conclusiones.

2. BPMS en sistemas distribuidos

Si bien los BPMS basados en un entorno *cloud* pueden ayudar a las pequeñas y medianas organizaciones a mejorar la eficiencia de los procesos y reducir sus gastos de infraestructura, todavía existen algunas barreras al utilizar este servicio.
Los dos temas principales de estudio en la adopción y del uso de un BPMS en el *cloud* son:

- **Protección de la privacidad**: algunos datos gestionados por los procesos de negocios pueden ser confidenciales y las organizaciones no están dispuestas a correr el riesgo de un posible robo de información. Si bien los servicios de *cloud* se encuentra ocultos detrás de barreras de seguridad provistas por la infraestructura de red y tienen un avanzado mecanismo para mantener la división de información entre sus clientes, los dueños de la información sensible temen perder el control de estos datos, o puede ocurrir que por reglas contractuales con sus clientes no se les permita almacenar información de éstos en otros lugares que no sean servidores propios.
- **Actividades que no son altamente computacionales**: por otro lado, la eficiencia y efectividad de las actividades que no son altamente computacionales puede disminuir debido a que la transferencia de los datos puede tomar más tiempo que el procesamiento mismo de los datos. Además, los costos de la actividad pueden incrementarse debido a que la transferencia de los datos es uno de los elementos de facturación en un sistema de *cloud computing*.

Hoy en día, en la mayoría de las soluciones donde se utilizan BPMS, el motor de proceso, las actividades y los datos de los procesos se encuentran alojados en un mismo sitio, ya sea en un sistema embebido o en un ambiente de *cloud computing*, donde presentan las ventajas e inconvenientes de cada uno de estos. Para poder utilizar los principales beneficios de ambos sitios, esto es, privacidad de datos en un sistema embebido, y poder de cómputo y ubicuidad relacionado a un sistema de cloud, se ha investigado en [4] un modelo de distribución, denominado PAD (Proceso-Actividad-Datos), en el que el arquitecto de procesos de negocio puede separar uno de estos procesos de acuerdo a los beneficios que brindan cada una de estas ubicaciones. Los ambientes distribuidos en el contexto de los procesos de negocio favorecen el rendimiento y proponen incorporar el concepto de descomposición de procesos, permitiendo que los mismos se ejecuten tanto en un entorno *cloud* o embebido [1][2][3][4].

3. Descomposición de procesos

El modelo de distribución de procesos llamado PAD (Proceso-Actividad-Datos), describe las posibilidades de separación de los componentes de un BPMS entre nodos

cloud y embebidos alternativamente. El modelo PAD define cuatro posibilidades de distribución:

1. El primer patrón de solución es el enfoque tradicional de BPM donde todos los elementos están alojados en el usuario final.
2. El segundo patrón es útil cuando el usuario ya tiene un sistema BPM, pero las actividades con intensidad de cómputo se localizan en el *cloud* para incrementar su performance.
3. El tercer patrón es útil para los usuarios que aún no poseen un sistema BPM; en este caso podrían adoptar un esquema de *cloud* de manera de pago por uso, donde las actividades sin intensidad de cómputo y los datos sensibles se pueden localizar en el usuario final.
4. El cuarto patrón es el modelo basado en *cloud* donde todos los elementos se localizan en la nube.

Es posible generalizar la distribución de procesos e identificar un patrón en el cual el motor de procesos, las actividades y los datos se despliegan tanto en la nube como en el usuario final (en la Figura 1 se puede observar este caso).
Esta solución presenta dos beneficios potenciales:

- El motor de procesos regula el flujo de control y el flujo de datos. Una actividad recibe datos del motor de procesos y luego de su ejecución los datos que son producidos se pasan de nuevo al motor de procesos.
- Cuando la nube no se encuentra accesible, los usuarios pueden ejecutar los procesos de negocio en forma completa en el sistema embebido hasta que el primero vuelva a estar disponible.

Para observar la diferencia entre un motor de procesos situado en un solo lugar, como por ejemplo en el patrón 2 del esquema de distribución PAD, y motores de procesos situados en ambos sitios, veamos el siguiente ejemplo.
Consideremos que una secuencia de actividades se ubica en la nube, mientras que el motor de procesos se despliega en el usuario final. Cada actividad utiliza los datos de salida de la actividad previa como entrada. Los datos no se envían directamente de una actividad a la otra sino que son enviados al motor de proceso primero (Figura 1).
Debido a que la transferencia de datos es uno de los factores de facturación en el modelo de *cloud computing*, estas situaciones pueden volverse más caras cuando se transfieren grandes cantidades de datos entre actividades.
Para evitar este problema se puede agregar un motor de procesos al *cloud*, el cual regula el flujo de control y el flujo de datos entre las actividades situadas en él (Figura 1b). Cuando una secuencia de actividades se localiza en el *cloud*, los datos se regulan por el motor de procesos en el *cloud*, lo cual reduce la cantidad de datos a ser transferidos entre este y el sistema embebido.

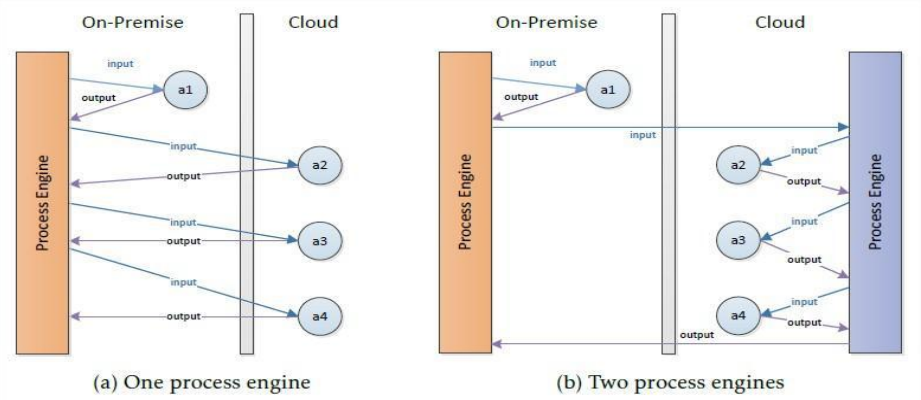


Figura 1. Datos enviados entre actividades coordinadas por motores de procesos [2]

Para poder correr un mismo proceso de negocio en dos motores de proceso separados, el mismo debe ser dividido en dos procesos individuales. Puede llegar a ser conveniente para los usuarios del BPMS tomar una lista de distribución del proceso de negocio y sus actividades, la cual puede ser automáticamente transformada en dos procesos de negocio, uno en el *cloud* y otro en el sistema embebido [1][2][3].

Una aproximación posible para manejar la descomposición del proceso es identificar la estructura y la semántica del mismo. Esto significa que se deben identificar aquellas actividades del proceso de negocio en donde la ejecución de las mismas dan la pauta de atomicidad, y que su ejecución, ya sea en el sistema embebido o en el *cloud*, no modificaría el comportamiento del proceso de negocio original. Una estrategia para detectar aquellas actividades atómicas es tratar de observar las dependencias de control y de datos que tienen con otras actividades y con el motor de procesos en el cuál se ejecutan. Por ejemplo, si existe una actividad que utiliza información confidencial que obtiene del sistema embebido, lo más probable es que no sea recomendable ejecutar la actividad en el motor de procesos situado en el *cloud*, ya que correríamos peligro de exponer tales datos. Al identificar las dependencias de control y de datos, se pueden investigar las consecuencias de mover ciertas actividades del sistema embebido al *cloud* y viceversa.

Cuando se conocen las consecuencias de la distribución de actividades, se puede crear una transformación de modelo en la cual un proceso de negocio y una lista con marcas se usan para crear dos procesos individuales, uno para el *cloud* y otro para el usuario final. Además, se puede generar una descripción de la coreografía para describir la comunicación entre ambos procesos de negocio utilizando algún lenguaje estándar.

4. Monitorización de procesos de negocio distribuidos

En las soluciones orientadas a procesos de negocios existen dos objetivos principales. Primero, la capacidad de proveer flexibilidad en los procesos de negocio, permitiendo una rápida reacción a los cambios del mercado. Y segundo, proveer una buena visualización de los procesos y permitir una optimización eficiente de estos.

Para poder optimizar los procesos de negocios tenemos que saber qué es lo que está funcionando mal o es ineficiente, por lo tanto, debemos comenzar a medir los procesos. Es aquí donde se puede utilizar una herramienta de monitorización llamada Monitorización de Actividades Empresariales (*BAM - Business Activity Monitoring*). Esta herramienta que se encuentra dentro de muchos BPMS nos permite medir varios aspectos de los procesos de negocio tales como tiempo, costos, rendimiento, estados y resultados de operaciones, procesos y transacciones.
Las personas encargadas de optimizar los procesos de negocio tienen que tener en cuenta varias tareas tales como identificar aquellas actividades que están teniendo poca eficiencia (ya sea en tiempo, costo o calidad); determinar los factores internos y externos que afectan la decisión de optimizar un proceso y como éstos influyen con el dueño del proceso, los dueños de las tareas y las áreas específicas de la organización; aplicar reingeniería, buenas prácticas o rediseño del proceso; definir nuevas medidas de rendimiento del nuevo proceso (*KPI - Key Performance Indicator*), realizar simulaciones y monitorearlos para confirmar que ha sido mejorado.
BAM provee acceso a información de los procesos en tiempo de ejecución, permite un análisis en tiempo real de los procesos de negocio, muestra los cuellos de botella en las tareas, mide el tiempo de cada tarea y provee herramientas para visualizar toda esa información [5]. Además, se utiliza para asegurar que los procesos de negocio funcionan como es esperado, simplificar la información compleja relacionada a los procesos y mostrarla oportunamente.

5. Solución propuesta

La solución que se propone para la ejecución y monitorización de los procesos de negocio distribuidos puede separarse en tres grandes fases: 1) el diseño del proceso de negocio distribuido, 2) la ejecución de cada uno de los subprocesos en los servidores en los que se encuentran, y 3) la recolección de la información perteneciente a los subprocesos o a las instancias, que servirán para realizar la monitorización.
El concepto de "subproceso" en la notación BPMN 2.0 tiene un significado diferente al que se utilizará en este trabajo. Mientras que en la bibliografía actual se utiliza el término para referirse a una actividad compuesta que es incluida dentro de un proceso y que contiene un conjunto de actividades, compuertas, eventos y flujos de secuencia [7], en esta sección se utilizará dicho término para referirse a las partes del proceso de negocio original que ha sido dividido a raíz de la descomposición del proceso.
El diseño del proceso de negocio distribuido, al igual que en el diseño de los procesos de negocio que se conocen comúnmente, es la fase previa para la ejecución de los procesos de negocio. Sin embargo, para realizar la ejecución de procesos de negocio de forma distribuida es necesario descomponerlos para crear los subprocesos que serán situados en los distintos motores de procesos que se encuentran en los servidores distribuidos, de acuerdo a si se desea aprovechar las características de pago por uso y la facilidad en la escalabilidad de sistemas basados en *cloud*, o la privacidad de los datos de los sistemas embebidos.
Como producto del proceso de negocio descompuesto se tendrán dos o más subprocesos que se deberán ejecutar en distintos motores de procesos encadenando el

flujo de control y de datos de las actividades pertenecientes al proceso original. Así, al finalizar una instancia en un servidor, esta inicia automáticamente una nueva instancia de la partición del proceso siguiente en el servidor que corresponda de acuerdo a la arquitectura de distribución. Para esto, cada servidor del esquema distribuido debe ser capaz de comunicarse con el servidor siguiente para poder iniciar instancias y continuar con la ejecución del proceso original. [1][3]

La ejecución del proceso de negocio distribuido comenzará con la generación de una instancia de uno de los subprocesos, y luego éste será el encargado de iniciar la ejecución del subproceso siguiente, o de los diferentes subprocesos en caso de que existiera más de uno. El seguimiento secuencial de la ejecución de los subprocesos, se implementa con un conector de Bonita que accede a la API REST del motor de procesos remoto. El conector se desarrolla para realizar dos tareas, la primera es la de instanciar procesos en motores de procesos remotos de Bonita, y la segunda, la de guardar en una base de datos local el identificador de las instancias generadas en otro motor, como así también el identificador de la nueva instancia en el motor de procesos remoto. Con estos datos, la aplicación de monitorización será capaz de interactuar con los diferentes servidores para recolectar la información perteneciente a las instancias ejecutadas en otros motores de procesos.

Para lograr un mejor seguimiento en la ejecución, el conector también genera datos de salida que sirven para controlar que se haya ejecutado correctamente o, y si la ejecución no fue exitosa, determinar el motivo de la falla. El uso de estos datos de salida servirá para controlar el flujo de trabajo del proceso, haciendo que la instancia no finalice hasta que se complete la instanciación del proceso remoto.

La fase de recolección de la información se lleva a cabo una vez que el proceso de negocio es ejecutado o está siendo ejecutado en los diferentes motores de procesos. Dada una instancia iniciada en un servidor de la arquitectura, debemos ser capaces de obtener, no solo los datos propios de la misma, sino también de cualquier instancia que esta haya desencadenado en otro servidor. Para poder realizar esto, será necesario asociar las distintas instancias de proceso iniciadas de manera encadenada, a los fines de poder recuperar información sobre las mismas accediendo a los distintos servidores intervinientes. [1][3]

Utilizando la API REST de Bonita y haciendo uso de la información almacenada en la base de datos, se puede obtener la definición de un proceso de negocio que se encuentra instalado dentro del motor de procesos, como así también la información de las instancias desplegadas dentro del motor de procesos. A través de las peticiones que realicemos a la API REST y con las respuestas obtenidas en formato XML, es posible desarrollar una aplicación web para recolectar la información referida al estado de los procesos y sus actividades en todo el camino de ejecución de los subprocesos de negocio en los diferentes servidores, unificar estos datos y visualizar al proceso de negocio distribuido como si se tratase de un proceso único. De esta manera, al unificar la información obtenida se le puede brindar al analista de procesos una visión integradora del proceso original, con el fin de monitorear las instancias, obtener valores de desempeño y determinar posibles fallas o mejoras que deban realizarse.

Si bien se ha hecho hincapié en la combinación del uso de un sistema *cloud* y uno embebido, en la solución propuesta se generaliza este modelo para lograr una visión

en la cual es posible adaptar la ejecución de procesos descompuestos a cualquier esquema de servidores distribuidos.
La arquitectura del sistema para la ejecución y monitorización de procesos de negocio distribuidos estará basada principalmente en tres componentes, estos son:

- El motor de procesos de Bonita Open Solution [6], que se ejecuta sobre un servidor Tomcat
- La aplicación de monitorización distribuida, desarrollada en PHP y que se ejecuta en un servidor Apache
- Un servidor de base de datos (MySQL) para almacenar la información de las instancias de los procesos de negocio

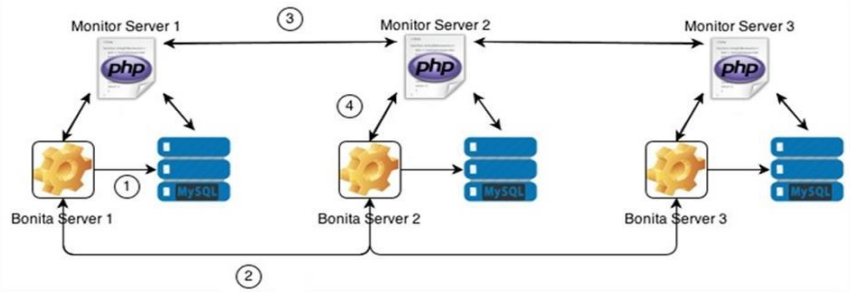


Figura 2. Arquitectura del Sistema de Ejecución y Monitorización con 3 servidores

6. Caso de estudio

El caso de estudio se basa en el Diseño 3D de un producto industrial, en donde el tiempo de procesamiento del modelo 3D del producto está dado por los detalles que contiene el mismo, y es por esto que el tiempo para efectuar la conversión del modelo 3D al diseño 3D depende del poder de procesamiento de la infraestructura de hardware.
El modelado es una técnica que se utiliza para ir dando forma a objetos. Por lo general, el modelo visual suele ser el modelo 3D que los diseñadores manejan, dejando las fórmulas a procesos computacionales. El proceso de transformación de un modelo en 3D hacia una imagen 3D es llamado renderización (rendering).
El propósito del caso de estudio, es utilizar el modelo de infraestructura como servicio de la nube, para realizar el renderizado del modelo 3D, siguiendo sus pasos a través de un proceso de negocio. Actualmente existe un servicio que brinda la empresa Autodesk llamado “Autodesk 360 Rendering”, en el cuál, al utilizar algunos de sus productos tales como AutoCAD, 3ds Max o Revit, le permite al usuario subir a la nube sus modelos y renderizarlos en ella.[8]
En este marco, suponemos una empresa que se compone de una Gerencia General, una oficina técnica, un departamento de mercadotecnia, un departamento de ingeniería, y un equipo de diseño, en donde la descomposición del proceso de negocio original (Figura 3) se hará en 3 servidores con sus respectivos BPMS de la siguiente manera: el primer servidor (en rojo), exclusivo de la Gerencia General, es el que ejecuta las actividades que manejan los datos sensibles de la empresa, alojado detrás

de un firewall o cuenta con una base de datos con acceso restringido, para asegurar la seguridad y confidencialidad de los proyectos; el segundo servidor (en verde), contiene las actividades que realizan los diferentes equipos de Ingenieros, de Mercadotecnia, de Diseñadores y la Oficina Técnica, para llevar a cabo los proyectos; y el tercer servidor (en azul), va a estar situado en la nube, ejecuta las actividades de renderizado de los modelos 3D.

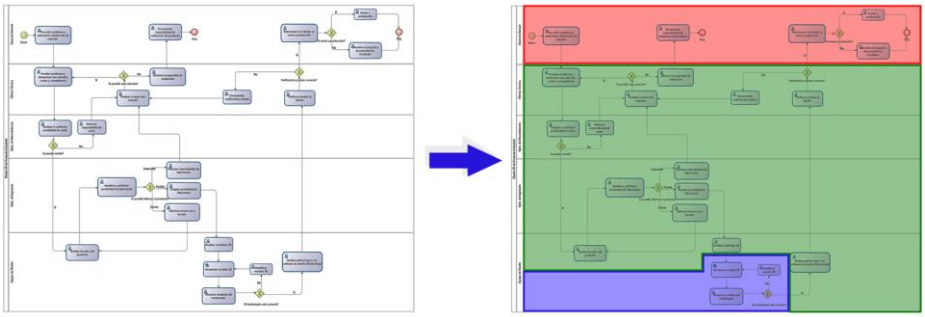


Figura 3. Descomposición del proceso de negocio original y división de servidores

El proceso de negocio original funciona de la siguiente manera: en lo más alto de la pirámide organizacional se encuentra la Gerencia General, encargada de plantear los problemas a solucionar, y tomar las decisiones de fondo de la empresa como así también de inicio y finalización de cada uno de los proyectos. Luego, se encuentra la Oficina Técnica que se encarga de analizar el problema y determinar posibles soluciones de acuerdo a factores como costos, competencia, entre otros, y funciona como nexo entre la Gerencia General y los equipos de especialistas. El Equipo de Mercadotecnia se encarga de analizar si el producto es capaz de comercializarse en el mercado, o si es necesario realizar cambios en las especificaciones que se han definido. El Equipo de Diseño realiza los bocetos del producto como una aproximación a la idea de modelo que tendrá dicho producto. El Equipo de Ingenieros verifica que es posible la fabricación del producto contenido en el boceto y se encarga de aprobar, enviar modificaciones o rechazar tales bocetos. En caso de que sean aprobados, los diseñadores deben realizar el modelo en 3D del producto, utilizando programas especializados basados en herramientas CAD (*Computer-Aided Design*), para enviar a renderizar al servidor que se encuentra en la nube y obtener una imagen real de producto final. La Oficina Técnica se encarga de realizar las pruebas necesarias y la aprueba de acuerdo a las especificaciones requeridas por la Gerencia General, que es la encargada de rechazar o aprobar el proyecto, y enviarlo a producción o al cliente que lo solicitó.

Este proceso de negocio es un claro ejemplo de la conjunción de las diferentes características tratadas en el punto 2, donde podemos ver que en el servidor 1 protegemos la privacidad de los datos de la empresa, mientras que en el servidor 3, situado en la nube, aprovechamos la capacidad de procesamiento otorgada por el servicio de infraestructura (*IaaS – Infraestructure as a Service*) que nos brinda *cloud computing*.

7. Conclusiones

Actualmente, las organizaciones que utilizan sistemas basados en BPMS como tecnología para conducir los procesos de negocios que modelan la actividad de dichas organizaciones, tienen sistemas embebidos con arquitecturas centralizadas donde deben estar sostenidas por una infraestructura muy costosa que a medida que pasa el tiempo se va dañando, teniendo que reparar los componentes, o va quedando obsoleta, en cuyo caso se debe reemplazar por una nueva seguramente más costosa que la anterior.

A partir de la descomposición de procesos de negocio, ejecución distribuida y con despliegue en la nube surgen las siguientes ventajas:

- Se aumenta la disponibilidad de las aplicaciones que interactúan a través del proceso.
- Se logra la integración de elementos dispersos en una arquitectura a través de servidores BPM
- Ubicuidad de los servidores, permitiendo la ejecución de los procesos en áreas más cercanas al lugar de los responsables de las tareas.
- Infraestructuras elásticas y adaptables a la necesidad del negocio, y con pago sólo por el uso, evitando grandes costos en infraestructuras propias.

Por otra parte, la distribución conlleva pérdida de la visión global del proceso tanto para la monitorización como para transparencia de la distribución en ejecución, siendo ambos aspectos resueltos por la propuesta de este trabajo.

Referencias

1. Karabogosian L., Bazán P., Martinez Garro J. “Ejecución y monitoreo de procesos de negocios distribuidos entre diferentes motores de Bonita OS”. WICC 2014 XVI Workshop de Investigadores en Ciencias de la Computación. 2014
2. Evert F. Duipmans. "Business Process Management in the cloud with data and activity distribution". Faculty of electrical engineering, mathematics and computer science software engineering. University of Twente. EWI/SE - 2012-002. November 2012.
3. J. Martinez Garro, P. Bazán. "Monitoreo de procesos distribuidos en el cloud: Una propuesta arquitectónica". Jornadas Chilenas de Computación 2013.
4. Han YB, Sun JY, Wang GL. “A cloud-based BPM architecture with user-end distribution of non-compute-intensive activities and sensitive data”. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 25(6): 1157–1167 Nov. 2010. DOI 10.1007/s11390-010-1092-5
5. Jiri Kolar. “Business Activity Monitoring”. Masaryk University Faculty of Informatics. Brno, spring 2009.
6. Bonitasoft - Open Source Workflow & BPM software. http://es.bonitasoft.com
7. Object Management Group, Inc. (OMG). "Business Process Model and Notation (BPMN) Version 2.0". January 2011
8. Brian Budge. Autodesk 360 Rendering - Scalable and Robust Rendering in the Cloud. http://www.highperformancegraphics.org/previous/www_2012/media/Hot3D/HPG2012_Hot3D_Autodesk.pdf

Comportamiento de Métricas para Proyectos de Explotación de Información en PyMEs

Diego Basso[1,2], Darío Rodríguez[2], Ramón García-Martínez[2]

1. Departamento de Ingeniería e Investigaciones Tecnológicas.
Universidad Nacional de La Matanza
2. Laboratorio de Investigación y Desarrollo en Ingeniería de Explotación de Información.
Grupo de Investigación en Sistemas de Información. Universidad Nacional de Lanús.
diebasso@yahoo.com.ar, drodrigu@unla.edu.ar, rgm1960@yahoo.com

Resumen. En este trabajo se presenta un estudio del comportamiento de las métricas propuestas para proyectos de explotación de información, la cual considera características y parámetros identificados para proyectos pequeños, de aplicación en las empresas PyMEs. Para ello, se realiza una introducción sobre la categorización definida para las métricas consideradas y el modelo de proceso de desarrollo utilizado como referencia, se delimita el problema presentando el diseño experimental y los resultados obtenidos para finalizar con la puntualización de algunas conclusiones.

Palabras claves: Explotación de Información. Métricas. Comportamiento de Metricas. Ingeniería de Proyectos de Explotación de Información.

1. Introducción

El uso de métricas es una característica importante de todas las disciplinas de ingeniería. Dentro de un marco de trabajo ingenieril, las métricas permiten cuantificar aspectos específicos de un proceso, de un producto o de un proyecto. En este sentido, recolectar métricas constituye el primer paso para saber cómo controlar y mejorar el proceso de desarrollo de software [Fenton y Bieman, 2014].

Los proyectos de explotación de información deben considerar la aplicación de una metodología de desarrollo [Chapman et al., 1999; Pyle, 2003; García-Martínez et al., 2011] que incluya entre sus actividades el registro de métricas, que permitan medir y controlar el avance del proyecto y evaluar su calidad. En esta disciplina de la Ingeniería, los proyectos presentan características diferentes respecto a los proyectos de desarrollo de software tradicional y esa diferencia está en la naturaleza del producto resultante [Marbán, 2003; Marbán et al., 2008]. Esto implica que las etapas habituales y métricas definidas para un proyecto de desarrollo clásico no sean totalmente apropiadas para un proyecto de explotación información [Basso et al, 2013; Pytel et al., 2015]. Dentro de este campo de conocimiento, se han propuesto y desarrollado distintas herramientas, entre las que se incluyen un modelo de proceso [Martins et al., 2014], modelos de estimación de esfuerzo [Pytel et al., 2015], y una propuesta de métricas para proyectos de explotación de información [Basso, 2014].

Esta comunicación presenta el estudio del comportamiento de algunas de las métricas propuestas en [Basso, 2014]. Se indica la categorización definida para las métricas (sección 2); luego se delimita el problema (sección 3), se detallan los materiales y métodos usados en el estudio (sección 4) y se presentan los resultados obtenidos (sección 5). El trabajo finaliza con la exposición de conclusiones preliminares obtenidas (sección 6).

2. Métricas para Proyectos de Explotación de Información

Se establece la clasificación que se indica en la Figura 1, para las métricas propuestas en [Basso et al. 2013; Basso, 2014]. Esta clasificación se formula a partir de: (a) las características y consideraciones presentadas en [Marbán, 2003; Marbán et al., 2008] para el modelo de estimación DMCoMo, (b) los parámetros establecidos en [Pytel, 2014] para Proyectos Pequeños, (c) las indicadas en [Pytel et al., 2015] para el Método de Estimación de Esfuerzo propuesto para PyMEs, y (d) los subprocesos y tareas propuestas por [Martins et al., 2014] para el Modelo de Proceso de Desarrollo para Proyectos de Explotación de Información.

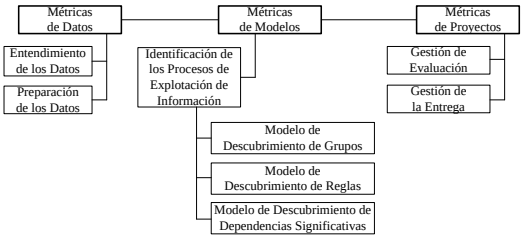


Fig. 1. Clasificación de las Métricas para Proyectos de Explotación de Información

Asimismo, se menciona que las métricas del subproceso de Identificación de los Procesos de Explotación de Información se agrupan según su tarea de descubrimiento y aplicación en un proyecto, contemplando los procesos de explotación de información definidos en [Britos, 2008; García-Martínez et al., 2013], los cuales se basan en la utilización de sistemas inteligentes [García-Martínez et al., 2003].

3. Delimitación del Problema

En [García-Martínez et al., 2011] se señala la necesidad de disponer de métricas asociadas al proceso de desarrollo de un proyecto de explotación de información para PyMEs, que permitiera suministrar información relevante a tiempo y establecer objetivos de mejora en los procesos y productos, con el fin de garantizar la calidad de estos proyectos. En este contexto, el objetivo de este trabajo es analizar el comportamiento de las métricas propuestas de Datos, Modelos y Proyectos, con especial focalización en proyectos de explotación de información de tamaño pequeño, que son los que usualmente requieren las PyMEs [García-Martínez et al., 2011].

4. Materiales y Métodos

Para construir el conocimiento asociado al presente proyecto de investigación, se siguió un enfoque de investigación clásico [Creswell, 2002; Rosas y Riveros, 1985] con énfasis en la producción de tecnologías [Sábato y Mackenzie, 1982], identificando métodos y materiales necesarios para desarrollar el mismo.
El método utilizado para generar los estudios del comportamiento de métricas es el de generación de valores de variables independientes del proyecto por el Método de Monte Carlo [Cohen, 1988].
Los materiales usados para la investigación que se reporta son las métricas que se proponen en [Basso, 2014] y se derivan de los trabajos [Marbán, 2003; Marbán et al., 2008; Martins et al., 2014; Pytel, 2014; Pytel et al., 2015].
El protocolo utilizado para la experimentación fue el siguiente:

Paso 1: Desarrollo de un banco de pruebas donde se generan los datos de distintos proyectos y modelos de explotación de información con los valores de las variables experimentales independientes (métricas básicas) y se aplican las variables experimentales dependientes (métricas derivadas).

Paso 2: Integrar estadísticamente la información obtenida generando los gráficos y tablas auxiliares que se consideren necesarias.

Paso 3: Interpretar los resultados experimentales obtenidos y formular, como conclusión, una regla de comportamiento general de la métrica.

5. Resultados Obtenidos

En esta sección se presentan los resultados obtenidos del estudio experimental de comportamiento de las métricas de Datos (sección 5.1), métricas de Modelos (sección 5.2) y métricas de Proyectos (sección 5.3).

5.1. Métricas de Datos

En esta sección se presentan los resultados experimentales del comportamiento estudiado para las métricas de Densidad de Valores Nulos, Nivel de Compleción de la Tabla, Grado de Corrección de Datos de la Tabla y Grado de Utilidad de Atributos.

Densidad de Valores Nulos y *Nivel de Compleción de la Tabla*:

Métricas	**Referencia**
$DVN(T) = \frac{NVN(T)}{NR(T) * NA(T)}$ $\quad NCT(T) = 1 - DVN(T)$	– Nro. de valores nulos o faltantes - NVN (T) – Nro. de registros de la tabla - NR (T) – Nro. de atributos de la tabla - NA (T)

Del estudio del comportamiento de las métricas analizadas se concluyó que: al aumentar el número de valores nulos en las tablas, la densidad de valores nulos tiende a 1 y el nivel de compleción a 0. Lo contrario ocurre al disminuir el número de valores nulos. En base a este análisis y a lo establecido en el trabajo de [Pytel, 2014], se sugiere que el valor de la métrica DVN (T) sea inferior al 15% de los datos,

mientras que la métrica NCT (T) sea superior al 85%. Esto contribuye a considerar datos con una calidad aceptable y facilitar el análisis posterior de los mismos, permitiendo de esta manera una adecuada aplicación de los tipos de modelado para descubrimiento de conocimiento. La misma conclusión es aplicable si se analiza la densidad de valores nulos por atributo o registro de la tabla.

Grado de Corrección de Datos de la Tabla:

Métrica	Referencia
$GCD(T) = 1 - \frac{NVE(T) + NVN(T)}{NR(T) * NA(T)}$	– Nro. de valores erróneos - NVE (T) – Nro. de valores nulos o faltantes - NVN (T) – Nro. de registros de la tabla - NR (T) – Nro. de atributos de la tabla - NA (T)

Se consideró que cuando la medida de la métrica toma un valor ALTO (un valor superior a 0.70) significa que los datos de la tabla son útiles para el proyecto, mientras que un valor por debajo, considerado BAJO, indica que deben analizarse los datos con mayor precisión y evaluar la relevancia o representatividad de la tabla para el proyecto.

Del estudio del comportamiento de la métrica analizada se concluyó que: al aumentar el número de valores erróneos y nulos en las tablas por encima del 30% de los datos totales, el grado de corrección de los datos de la tabla GCD (T) tiende a ser BAJO. Al igual que con el número de valores nulos NVN (T), se sugiere que el número de valores erróneos NVE (T) en una tabla sea inferior al 15% de los datos disponibles, para lograr que la misma posea un ALTO grado de corrección (mayoría de datos útiles). La misma conclusión es aplicable por cada atributo o registro de la tabla, permitiendo identificar los atributos y registros con más valores erróneos.

Grado de Utilidad de Atributos:

Métrica	Referencia
$GUA(T) = \frac{NA(T) - (NO_UTILES(T) + 0,5 * NAUD(T))}{NA(T)}$ $NA(T) = NASE(T) + NAUD(T) + NANC(T) + NANS(T)$ $NO_UTILES(T) = NANC(T) + NANS(T)$	– Nro. de atributos útiles sin errores – NASE (T) – Nro. de atributos útiles con defectos – NAUD (T) – Nro. de atributos no correctos – NANC (T) – Nro. de atributos no significativos – NANS (T)

Se consideró que cuando la medida de la métrica toma un valor ALTO (un valor superior a 0.70) significa que la mayoría de los atributos son de utilidad y significativos para el proyecto y no se requieren demasiadas correcciones; cuando el valor es MEDIO (un valor entre 0.30 y 0.70) implica que los atributos son aceptablemente usables en el proyecto pero requieren correcciones en los datos, mientras que cuando toma un valor BAJO (un valor inferior a 0.30) significa que los atributos de la tabla no son útiles y representativos para el proyecto, pudiéndose descartar la misma. En la Figura 2 se muestra la proporción por tipo de atributos necesarios en una tabla, para alcanzar los distintos grados de utilidad de la métrica en un proyecto de explotación de información para PyMEs.

La métrica y los parámetros propuestos alcanzan una precisión promedio del 70% en la valoración de la utilidad de una tabla. Del estudio del comportamiento de la métrica analizada, se concluyó que la misma alcanza sus diferentes umbrales de referencia cuando se producen los siguientes comportamientos en las métricas básicas:

- El grado de utilidad de los atributos es ALTO, cuando existe un mínimo de 80% de atributos útiles. Este indicador representa una cantidad de atributos útiles sin

errores, asociado a la métrica NASE (T), mayor o igual al 70% y una cantidad de atributos útiles con defectos, asociado a la métrica NAUD (T), menor al 30%. A su vez, la cantidad de atributos no útiles, asociado a las métricas NANC (T) y NANS (T), no debe superar el 20% del total de atributos. Las conclusiones obtenidas indican que es preciso disponer de muchos atributos útiles y pocos o muy pocos atributos no útiles para que la tabla sea representativa en el proyecto.

- El grado de utilidad de los atributos es MEDIO, cuando la cantidad de atributos útiles es del 40 al 80%. Este resultado implica una cantidad de atributos útiles sin errores, asociado a la métrica NASE (T) entre el 30 y 70%, y una cantidad de atributos útiles con defectos, asociado a la métrica NAUD (T), menor al 50%. A su vez, la cantidad de atributos no útiles, asociado a las métricas NANC (T) y NANS (T), es del 20 al 60% del total de atributos. Las conclusiones obtenidas indican que si la cantidad de atributos útiles en la tabla es de normal a muchos y los atributos no útiles se incrementan en el rango de pocos a bastantes, la utilidad de dichos atributos cae a un grado medio, requiriéndose algunas correcciones en los datos.
- El grado de utilidad de los atributos es BAJO, cuando existe un máximo de 40% de atributos útiles. Este indicador constituye una cantidad de atributos útiles sin errores, asociado a la métrica NASE (T), menor al 30% y una cantidad de atributos útiles con defectos, asociado a la métrica NAUD (T), menor o igual al 20%. A su vez, la cantidad de atributos no útiles, asociado a las métricas NANC (T) y NANS (T), es al menos el 60% del total de atributos. Las conclusiones obtenidas indican que si se tienen muy pocos o una cantidad normal de atributos útiles, pero bastantes o muchos atributos no útiles, la tabla deja de ser relevante para el proyecto.

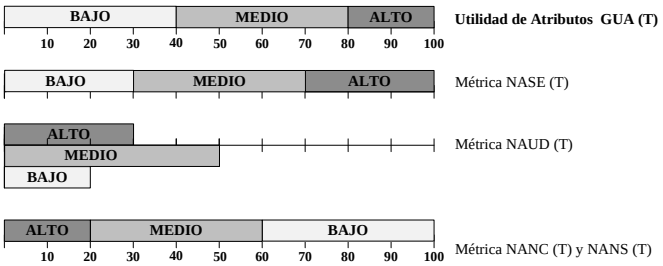


Fig. 2. Proporción de atributos útiles y no útiles según rango en métrica GUA (T)

5.2. Métricas de Modelos

En esta sección se presentan los resultados experimentales del comportamiento estudiado para las métricas de Precisión del Modelo, Cobertura de una Regla y Grado de Incidencia de un Atributo.

Precisión del Modelo:

Métrica	Referencia
$PDM(C_i) = \frac{NCVC_i}{NCVC_i + NCFC_i}$	– Nro. de casos pertenecientes a la clase C_i (o grupo) correctamente clasificados por el modelo en esa misma clase - $NCVC_i$. – Nro. de casos incorrectamente clasificados por el modelo como clase C_i y que pertenecen a otras clases - $NCFC_i$

Se consideró que cuando la medida de la métrica toma un valor ALTO (un valor superior 0.70) significa que el modelo posee un buen nivel de efectividad para clasificar casos en su clase o grupo; cuando el valor es MEDIO (un valor entre 0.50 y 0.70) implica que el modelo puede confundir los casos clasificados a cada clase, convirtiéndose en un modelo inestable y poco eficiente, mientras que cuando toma un valor BAJO (un valor inferior a 0.50) significa que el modelo confunde las clases a las que pertenecen los casos clasificados, resultando un modelo poco efectivo.
La métrica y los parámetros propuestos alcanzan una exactitud promedio del 70% en la valorización del grado de precisión de los modelos en un proyecto. Asimismo, el comportamiento de la métrica comprende a los modelos de explotación de información cuya exactitud de clasificación es superior al 70%. Del estudio del comportamiento de la métrica analizada, se concluyó que alcanza sus diferentes umbrales cuando se producen los siguientes comportamientos en las métricas básicas:

- El grado de precisión de un modelo es ALTO, cuando el número de casos clasificados correctamente en cualquiera de las clases o grupos, asociado a la métrica $NCVC_i$, está entre el 11 y 70% del total de casos (representado del 20 al 70% de casos), el número de casos clasificados incorrectamente en esa misma clase, asociado a la métrica $NCFC_i$, es menor al 10% y la exactitud del modelo para generalizar, predecir y clasificar correctamente nuevos casos es superior al 80%.
- El grado de precisión de un modelo es MEDIO, cuando el número de casos clasificados correctamente en cualquiera de las clases o grupos está entre el 1 y 40% del total de casos (representado del 10 al 40% de casos), el número de casos clasificados incorrectamente en esa misma clase está entre el 1 y 20% (representado del 10 al 20% de casos) y la exactitud del modelo está entre 70 y 90%.
- El grado de precisión de un modelo es BAJO, cuando el número de casos clasificados correctamente en cualquiera de las clases o grupos es menor al 10% del total de casos, el número de casos clasificados incorrectamente en esa misma clase está entre el 11 y 30% (representado del 20 al 30% de casos) y la exactitud del modelo está entre 70 y 80%.

Las conclusiones obtenidas muestran que si la precisión de un modelo de explotación de información es aceptable (con valores ALTOS), el mismo resulta efectivo y puede ser utilizado para predecir y clasificar nuevos casos del mismo dominio del proyecto, de los que desconoce su clase o grupo. En cambio, si la precisión alcanza niveles MEDIOS o BAJOS, aumenta el riesgo que el modelo se vuelva inestable y comience a confundir las clases o grupos durante la clasificación de los casos, convirtiéndose en un modelo poco efectivo y confiable.

Cobertura de una Regla:

Métrica	**Referencia**
$COBER(R) = \frac{NCCNS(R)}{NCE(C)}$ Valor Mín. Umbral: $COBER(R) > \frac{100}{NRGL(C)}$ [%]	– Nro. de casos que satisfacen la aplicación de la regla de comportamiento o de pertenencia de una clase o grupo - NCCNS (R) – Nro. de casos por clase para entrenar el modelo - NCE (C). – Nro. de reglas descubiertas por el modelo por cada clase o grupo - NRGL (C)

Se propuso un valor mínimo de umbral de cobertura de una regla, a partir del número de reglas de pertenencia obtenidas para una clase o grupo. La utilidad de esta métrica

en un proyecto que aplica un modelo de Descubrimiento de Reglas [García-Martínez et al., 2013], está en que aquellas reglas con mayor cobertura son las más representativas y útiles del modelo para definir el comportamiento de una clase o grupo. Por consiguiente, representan las reglas que mayor credibilidad e interés le otorgan al modelo construido para clasificar nuevos casos a una clase o grupo.
Del estudio del comportamiento de la métrica analizada se concluyó que: cuando se incrementa el número de casos a los que se le puede aplicar una regla de pertenencia a una clase o grupo, asociada a la métrica NCCNS (R), respecto del total de casos de esa misma clase, la métrica COBER (R) aumenta su valor. Mientras que, cuando el número de casos a los que se puede aplicar la regla disminuye, también lo hace la métrica de cobertura de la regla. Por otra parte, obtener un bajo número de reglas para una clase o grupo, permite identificar más fácilmente las características representativas del comportamiento de una clase o grupo del modelo de explotación de información. A su vez, provoca un aumento en el valor de umbral de cobertura, la necesidad de una mayor cantidad de casos a cubrir con la regla, dándole credibilidad e interés a ésta para clasificar nuevos casos a una clase o grupo. Si el número de reglas es alto, el valor de umbral para la métrica de cobertura es bajo, menor es la cantidad de casos cubiertos con una regla y más dificultoso y complejo resulta identificar el comportamiento asociado a una clase o grupo del modelo. En consecuencia, es deseable que la cantidad de reglas que definen el comportamiento de una clase o grupo sea siempre la más baja posible.

Grado de Incidencia de Atributos:

Métrica	Referencia
$GINC(A_{vi}) = \frac{NCAT(A_{vi})}{NCE(C)}$	– Nro. de casos cubiertos por una clase o grupo del atributo clase, cuando un atributo significativo A toma el valor V_i - NCAT (A_{vi}) – Nro. de casos por clase para entrenar el modelo - NCE (C).

Del estudio del comportamiento de la métrica analizada se concluyó que: al aumentar el número de casos cubiertos por alguno de los valores V_i de un atributo significativo, para un valor específico del atributo clase, la métrica GINC (A_{vi}) aumenta para ese valor V_i, mientras que para los demás valores del atributo significativo disminuye. Asimismo, todo valor V_i, asociado a un atributo significativo de un modelo, cuyo grado de incidencia sea máximo en una clase o grupo de un atributo clase, representa las características o factores que más influencia tienen sobre un determinado resultado para el proyecto, según el problema de explotación de información planteado. Cabe mencionar que, cuando esta métrica toma valores muy similares en dos o más valores del atributo significativo, implica que no hay una única característica (en ese mismo atributo) que sea más relevante sobre el resultado del atributo clase.

5.3. Métricas de Proyectos

En esta sección se presentan los resultados experimentales del estudio del comportamiento realizado para la métrica de Éxito de Resultado del Proceso de Desarrollo de un proyecto de explotación de información.

Éxito de Resultado del Proceso de Desarrollo:

Métrica	Referencia
$EPD(P) = \frac{\sum NMI_i * Peso(i)}{NMOD}$	– Nro. de modelos de explotación de información cuyo interés *i* y utilidad de resultados es Alto, Medio o Bajo - NMI_i – Valor de ponderación asignado a cada modelo de acuerdo al interés de los resultados obtenidos - Peso(i) – Nro. de modelos construidos para el proyecto - NMOD

Se consideró que cuando la medida de la métrica toma un valor ALTO (un valor superior a 0.80) significa que el proceso de desarrollo aplicado es correcto y los resultados alcanzados cumplen con los criterios de éxito del proyecto; cuando el valor es MEDIO (un valor entre 0.61 y 0.80) implica que los resultados cumplen parcialmente con los criterios de éxito y las tareas del proceso deben ser revisadas y ajustadas, mientras que cuando toma un valor BAJO (un valor inferior a 0.60) significa que los resultados obtenidos no responden a los criterios de éxito del proyecto. Por otra parte, por cada modelo se estableció un criterio de valoración (muy bajo, bajo, medio y alto) y un valor de ponderación (de 0 a 1), por nivel de satisfacción de resultados e interés del conocimiento descubierto para el proyecto. En la Figura 3 se muestra el número estimado de modelos necesarios, junto al grado de interés esperado de resultados, para alcanzar los diferentes niveles de éxito de la métrica en un proyecto de explotación de información para PyMEs.

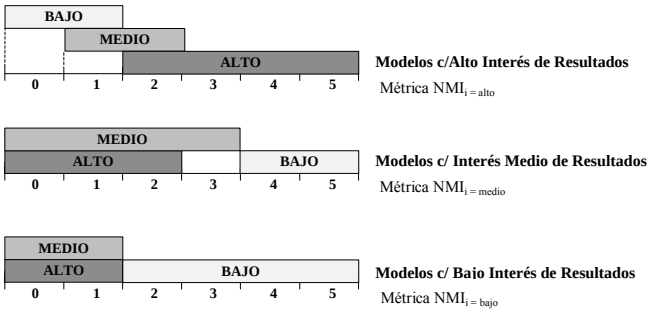


Fig. 3. Número de modelos por nivel de interés de resultados s/rango de la métrica EPD (P)

La métrica y los parámetros propuestos alcanzan una exactitud promedio del 70% en la valorización el éxito del proceso de desarrollo en un proyecto. Del estudio del comportamiento de la métrica analizada alcanza sus diferentes umbrales cuando se producen los siguientes comportamientos en las métricas básicas:

- El éxito de resultados del proceso de desarrollo es ALTO, cuando el número de modelos que presentan un alto interés para el usuario, por el conocimiento descubierto y su utilidad, asociado a la métrica NMIi, es mayor o igual a 2. A su vez, el número de modelos que presentan un interés medio es menor o igual a 2, mientras que el número de modelos con bajo interés es a lo sumo 1. Estas conclusiones indican que el proceso de desarrollo aplicado es correcto y se cumplen satisfactoriamente los criterios de éxito definidos para el proyecto. Cuanto mayor sea el número de modelos con este nivel de interés, más se acerca la medida de la métrica a su valor óptimo.
- El éxito de resultados del proceso de desarrollo es MEDIO, cuando el número de modelos que presentan un alto interés para el usuario, asociado a la métrica NMIi,

es 1 ó 2. A su vez, el número de modelos que presentan un interés medio es menor o igual a 3, mientras que el número de modelos con bajo interés para el usuario es a lo sumo 1. Estas conclusiones significan que se cumplen parcialmente con los criterios de éxito definidos para el proyecto. Esto se debe a que si bien se obtienen patrones de conocimiento interesante de los modelos, algunos expresan resultados que no son totalmente útiles como se presentan. En consecuencia, se necesita revisar el proceso de explotación de información utilizado y ajustar algunas tareas del proceso de desarrollo.

- El éxito de resultados del proceso de desarrollo es BAJO, cuando el número de modelos que presentan un alto interés para el usuario, asociado a la métrica NMIi, es a lo sumo 1. Mientras que el número de modelos con bajo interés es mayor o igual a 2. Estas conclusiones indican que no se alcanzaron los criterios de éxito. En consecuencia, resulta necesario revisar las tareas del proceso de desarrollo, especialmente los subprocesos de: (a) entendimiento de los datos, (b) preparación de los datos y (c) identificación de los procesos de explotación de información.

6. Conclusiones

Se definieron tres categorías generales de métricas aplicables a proyectos de explotación de información: *Métricas de Datos*, *Métricas de Modelos* y *Métricas de Proyectos*. En cada una de estas categorías se incluyeron algunas de las métricas propuestas, que cubren el Modelo de Proceso de Desarrollo con las tareas definidas en [Martins et al., 2014] para este tipo de proyectos. Además, se estudió de manera analítica, mediante simulación por Monte Carlo, el comportamiento de las métricas, con particular énfasis en su utilización en empresas PyMEs y proyectos pequeños.
Las conclusiones obtenidas para algunas métricas se basaron en valores numéricos asignados a los términos lingüísticos “Alto”, “Medio” y “Bajo”, a partir de la propia experiencia. En este contexto surge como futura línea de trabajo estudiar si ajustando estos valores, se obtiene un comportamiento más estable de las métricas cuando caen en estos rangos. Por otra parte, con algunas métricas se observaron comportamientos un poco imprecisos, ya que dependían de los valores utilizados en la simulación.
Se prevé validar las métricas propuestas de manera empírica, en el marco de los proyectos de explotación de información recopilados en [Pytel, 2014]. A su vez, se estima validar y ajustar los parámetros de las métricas de manera empírica, utilizando proyectos de explotación de información reales, a fin de evaluar su viabilidad.

7. Financiamiento

Las investigaciones que se reportan en este artículo han sido financiadas parcialmente por el Proyecto de Investigación 33A205 de la Secretaria de Ciencia y Técnica de la Universidad Nacional de Lanús (Argentina).

8. Referencias

Basso, D., 2014. *Propuesta de Métricas para Proyectos de Explotación de Información*. Revista Latinoamericana de Ingeniería de Software, 2(4): 157-218, ISSN 2314-2642.

Basso, D., Rodríguez, D., García-Martínez, R., 2013. *Propuesta de Métricas para Proyectos de Explotación de Información*. Workshop de Bases de Datos y Minería de Datos. Proceedings XIX Congreso Argentino de Ciencias de la Computación. Pag. 983-992. ISBN 978-987-23963-1-2.

Britos, P., 2008. *Procesos de Explotación de Información basados en Sistemas Inteligentes*. Tesis Doctoral. Universidad Nacional de La Plata. Facultad de Informática. Argentina.

Chapman, P., Clinton, J., Keber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R. 1999, *CRISP-DM 1.0 Step by step BIguide*. Edited by SPSS.

Cohen, J. 1988. *Statistical Power Analysis for the Behavioral Sciences* (2da edición). Lawrence Earlbaum Associates Publishers. ISBN 0-8058-0283-5.

Creswell, J. 2002. *Educational research: Planning, conducting, and evaluating quantitative*. Prentice Hall.

Fenton, N., & Bieman, J. 2014. *Software metrics: a rigorous and practical approach*. CRC Press.

García-Martínez, R., Britos, P., Pesado, P., Bertone, R., Pollo-Cattaneo, F., Rodríguez, D., Pytel, P., Vanrell. J., 2011. *Towards an Information Mining Engineering. En Software Engineering, Methods, Modeling and Teaching*. Sello Editorial Universidad de Medellín. ISBN 978-958-8692-32-6. Páginas 83-99.

García-Martínez, R., Britos, P., Rodríguez, D. 2013. *Information Mining Processes Based on Intelligent Systems*. Lecture Notes on Artificial Intelligence, 7906: 402-410. ISBN 978-3-642-38576-6.

García-Martínez, R., Servente, M., Pasquini, D., 2003. *Sistemas Inteligentes*. Editorial Nueva Librería. Buenos Aires.

Marbán, O., 2003. *Modelo Matemático Paramétrico de Estimación para Proyectos de Data Mining (DMCOMO)*. Tesis Doctoral. Departamento de Lenguajes y Sistemas e Ingeniería Software. Facultad de Informática. Universidad Politécnica de Madrid (UPM). España.

Marbán, O., Menesalvas E., Fernández-Baizán, C., 2008. *A cost model to estimate the effort of datamining projects (DMCoMo)*. Elsevier. Science Direct. Information System. Volume 33, Issue 1 (March 2008). Pp. 133–150.

Martins, S., Pesado, P., García-Martínez, R. 2014. *Propuesta de Modelo de Procesos para una Ingeniería de Explotación de Información: MoProPEI*. Revista Latinoamericana de Ingeniería de Software, 2(5): 313-332, ISSN 2314-2642.

Pyle, D. 2003. *Business Modeling and Business intelligence*. Morgan Kaufmann Publishers.

Pytel, P. 2014. *Viabilidad y Estimación de Proyectos de Explotación de Información*. Tesis Doctoral en Ciencias informáticas. Facultad de Informática. Universidad Nacional de La Plata.

Pytel, P., Hossian, A., Britos, P., García-Martínez, R. 2015. *Feasibility and Effort Estimation Models for Medium and Small Size Information Mining Projects*. Information Systems Journal, 47: 01-14. Elsevier. ISSN 0306-4379.

Rosas, L., Riveros, H. 1985. *Iniciación al método científico experimental*. Trillas.

Revisión Sistemática Comparativa de Evolución de Métodos de Extracción de Conocimiento para la Web

Juan Manuel Rodríguez[1,2], Hernán D. Merlino [1,2], Ramón García-Martínez[2]

[1] Cátedra de Sistemas de Soporte para Celdas de Producción Flexible.
Departamento de Computación. Facultad de Ingeniería, Universidad de Buenos Aires.
[2] Laboratorio de Investigación y Desarrollo en Sistemas de Inteligencia Artificial
Grupo Investigación en Sistemas de Información (LIDSIA-GISI).
Depto. Desarrollo Productivo y Tecnológico. Universidad Nacional de Lanús
http://sistemas.unla.edu.ar/sistemas/gisi/LIDSIA.htm
jmrodriguez1982@gmail.com, hmerlino@gmail.com, rgm1960@yahoo.com

Resumen. En esta comunicación se revisa la evolución los métodos de extracción de conocimiento, con foco en la extracción de relaciones semánticas desde el año 2005 hasta la fecha, teniendo como hilo conductor aquellos métodos creados para trabajar en la Web, centrados en grandes volúmenes de información no estructurada en lenguaje natural y sin dominio especifico. Se resumen distintas comparaciones realizadas entre los métodos relevados.

Palabras Clave. Extracción de conocimiento, extracción de relaciones semánticas, métodos de extracción auto-supervisados, *open information extraction*, procesamiento de lenguaje natural.

1. Introducción

Desde el cambio de milenio, la Web se ha convertido en un repositorio emergente de conocimiento embebido que crece exponencial de manera continua. La necesidad de explotar estos conocimientos ha servido para recuperar la tradición de amplio campos de la Inteligencia Artificial en algo que la comunidad llama en la actualidad computación cognitiva (*cognitive computing*) [Modha et al., 2011] pero en el nuevo contexto de grandes datos. En este nuevo contexto aparecen necesidades específicas dentro de las técnicas de extracción de conocimiento como lo son las técnicas de extracción de relaciones semánticas para grandes volúmenes de datos no estructurados, en particular en lenguaje natural.

Utilizando el método de revisiones sistemáticas [Argimón, 2004] se ha realizado una investigación documental sobre métodos de extracción de conocimiento para la web y se han comparado la calidad de los resultados.

En este contexto, se introducen conceptos de extracción de conocimiento (sección 2), se revisan las distintas técnicas de extracción de relaciones semánticas (sección 3) focalizándose en los llamados métodos basados en conocimiento (sección 3.1), los métodos de tipo supervisados (sección 3.2), y los métodos auto-supervisados (sección 3.3). Se formula una revisión exhaustiva de los distintos métodos de extracción de relaciones semánticas auto-supervisados creados para la Web (sección 4), detallando

las distintas formulas utilizadas para comparar le efectividad de los métodos (sección 4.1), y revisando en dichas comparaciones (sección 4.2). Por último, se presenta una comparación global formulada por los autores sobre la calidad de respuesta de unos métodos frente a otros con detalle de la fuente de cada comparación (sección 5), se indican las diferencias observadas entre ellas y se da cuenta de las comparaciones que quedan por hacerse. Finalmente se presentaln conclusiones parciales y se señalan las futura lineas de trabajo (sección 6).

2. Introducción a la Extracción de Conocimiento

La extracción de conocimiento es cualquier técnica mediante la cual un proceso automatizable es capaz de analizar fuentes de información no estructurada, como por ejemplo textos escritos en lenguaje natural y extraer el conocimiento allí embebido para representarlo de una manera estructurada, manipulable en procesos de razonamiento automático, como por ejemplo: una regla de producción o un subgrafo en una red semántica. A la información obtenida como salida de este tipo de procesos se la llama: pieza de conocimiento [García-Martínez & Britos, 2004; Gómez et al., 1997] Si se piensa a la extracción de conocimiento como una transformación algebraica podría plantearse:

```
extracción_de_conocimiento(estructuras_de_información) = piezas_de_conocimiento.
```
(1)

El desafío de la extracción de conocimientos comienza a fines de la década de 1970 como es señalado en [Cowie & Lehnert, 1996]. Más tarde en los años 90 la investigación fue alentada y financiada por la Agencia de Proyectos Avanzados de Defensa (DARPA) [Konstantinova, 2014].

Los métodos de extracción de conocimiento comenzaron trabajando en la detección y clasificación de nombres propios, utilizando como entrada fuentes de información no estructurada, este tipo de extracción de conocimiento es llamado Reconocimiento de Nombres de Entidades (NER según sus siglas en inglés). En general estos sistemas de extracción de conocimiento buscan nombres de personas, compañías, organizaciones y lugares geográficos [Konstantinova, 2014]. El siguiente paso que dieron los métodos de extracción de conocimiento fue el de resolver correferencias y el de extraer relaciones entre nombres de entidades [Jurafsky & Martin, 2000].

Hacía fines de la década de 2000 los métodos de extracción de conocimiento se habían diversificado y especializado. En [Jurafsky & Martin, 2000] se reconocen distintos tipos de piezas de conocimiento susceptibles de ser extraídas: nombres de entidades, expresiones temporales, valores numéricos, relaciones entre entidades y expresiones previamente identificadas, eventos, entre otras.

La extracción de conocimiento tradicionalmente ha requerido de participación humana en la forma de reglas de extracción o bien de ejemplos de entrenamiento etiquetados de forma manual. En particular para los casos de extracción de relaciones entre entidades, es el usuario quien debe explícitamente especificar cada relación que le interese, tarea ardua, sobre todo cuando se trabaja con fuentes heterogéneas de

información no estructurada y con volúmenes de datos demasiado grandes, como podría ser la Web. Debido a ello en general los sistemas de extracción de conocimiento fueron utilizados sobre fuentes de información no estructurada más bien pequeñas y homogéneas [Banko et al., 2007].

3. Extracción de Relaciones Semánticas

Una subtarea comprendida dentro del conjunto de métodos de extracción de conocimiento es la de extraer de relaciones semánticas. En [Culotta et al., 2006] se define a la extracción de relaciones semánticas como: "la tarea de descubrir conexiones semánticas entre entidades". Y agrega que es de uso común realizar esta tarea utilizando como entrada textos en lenguaje natural en los cuales se suele identificar primeramente grandes cantidades de pares de entidades por documento para luego determinar si existe una relación entre estas utilizando pistas basadas en las características del lenguaje analizado.

En [Etzioni et al., 2008] se clasifican los métodos de extracción de relaciones semánticas en tres clases:

– métodos basados en conocimiento (*knowledge-based methods*)

– métodos supervisados (*supervised methods*)

– métodos auto-supervisados (*self-supervised methods*)

3.1. Métodos Basados en Conocimiento

Los primeros sistemas de extracción de relaciones eran específicos para un dominio, por ejemplo en 1991 DARPA desafió a la comunidad que estaba trabajando en procesamiento de lenguaje natural a "construir sistemas robustos capaces de llenar plantillas con piezas de conocimiento sobre el terrorismo en América Latina", los campos requeridos eran: fechas, ubicaciones, perpetradores, armas, víctimas y objetivos físicos. Más adelante los dominios fueron cambiando y se centraron en *joint ventures*, microelectrónica y planes para la sucesión de gestiones empresariales.

Este tipo de sistemas estaban basados en reglas de coincidencia de patrones (*pattern-matching)* creadas a mano para cada dominio. Estos sistemas tenían la desventaja de no ser escalables ni portables entre dominios diferentes [Etzioni et al., 2008].

3.2. Métodos Supervisados

Este tipo de métodos trabaja con un conjunto de datos de entrenamiento en donde ciertos ejemplos específicos, para un dominio de interés, son previamente etiquetados. Para luego, utilizando dichos ejemplos, entrenar un extractor de forma automática. La principal contra de este tipo de métodos radica en el tiempo y el esfuerzo que se requiere para construir el conjunto de datos de entrenamiento [Konstantinova, 2014].

3.3. Métodos Auto-supervisados

En 2005 Oren Etzioni en [Etzioni et al., 2005] presenta un método de extracción de relaciones semánticas llamadas KnowItAll, el cual es capaz de aprender a etiquetar sus propios ejemplos de entrenamiento utilizando solo un conjunto pequeño de patrones de extracción, independientes de cualquier dominio. Este fue el primer sistema publicado capaz de encarar la extracción de conocimiento de páginas Web ya que era no supervisado, independiente del dominio y escalable [Etzioni et al., 2005; Etzioni et al., 2008]. Los métodos estudiados en el presente trabajo pertenecen a esta última categoría.

4. Métodos de Extracción de Conocimiento Creados para la Web

En el año 2007 Michele Banko introduce un nuevo concepto en materia de extracción de conocimiento, al que llama en inglés: *Open Information Extraction* (OIE). Se trata de un paradigma de extracción de conocimiento en donde un sistema informático realiza una sola pasada sobre el total de las fuentes de información no estructurada en formato de lenguaje natural (llamado *corpus* de documentos), dadas como entrada y extrae un gran conjunto de tuplas relacionales sin requerir ningún tipo de participación humana. Cabe aclarar que este paradigma de extracción de conocimiento pertenece a la clase de métodos auto-supervisados. En el mismo trabajo Banko presenta un método llamado TEXT RUNNER, el cual es el primer método que trabaja dentro de este nuevo paradigma [Banko et al., 2007].

A partir de este trabajo se propusieron otros métodos de extracción de conocimiento bajo el paradigma que Banko llamó Open IE y que podríamos identificar de forma más concreta como métodos de extracción de conocimiento para la Web.

4.1. Comparaciones entre Distintos Métodos de Extracción de Conocimiento Creados para Web

En el punto 3.2 se presentan distintos métodos de extracción de conocimiento para la Web y comparaciones entre ellos. Dichas comparaciones fueron obtenidas del relevamiento de distintos artículos y es por ello que utilizan distintas formulas para evaluar de forma cuantitativa la calidad y cantidad de las piezas de conocimiento extraídas por los diferentes métodos de extracción de conocimiento.

La formula más comúnmente utilizada es la precisión, la cual se calcula como los casos de éxito sobre las extracciones totales, o más específicamente en este caso:

$$\text{Precisión} = \frac{\text{cantidad_de_piezas_de_ conocimiento_extraídas_ correctamente}}{\text{cantidad_de_piezas_de_ conocimiento_extraidas}} \quad (2)$$

La segunda fórmula más utilizada, pero prácticamente en conjunto con la precisión fue la exactitud (*recall* en inglés), la cual se calcula como la cantidad de casos de

éxito sobre la cantidad de casos relevantes totales, o más específicamente en este caso:

$$\text{Exactitud} = \frac{\text{cantidad_de_piezas_de_ conocimiento_extraídas_ correctamente}}{\text{cantidad_de_piezas_de_conocimiento_totales_en_el_documento}} \quad (3)$$

En la mayoría de los casos la cantidad de piezas totales de conocimiento fueron etiquetadas a mano, en ocasiones por más de una persona.

Otra fórmula utilizada fue la Medida-F, la cual se calcula utilizando las dos medidas anteriores más un parámetro β, que indica a cuál de las dos se le da una ponderación mayor. Su fórmula es la siguiente:

$$F_\beta = \frac{(1+\beta^2)2 * \text{Precisión} * \text{Exactitud}}{(\beta^2 * \text{Precisión}) + \text{Exactitud}} \quad (4)$$

En todos los artículos relevados, siempre se utiliza esta medida con $\beta=1$, para simplificar en este trabajo se referirá a la Medida-F con $\beta=1$, como Medida-F1 o simplemente F1.

Por último en algunos artículos se utiliza el área bajo la curva *Receiver Operating Characteristic* (ROC) como medida de la calidad de las piezas de conocimiento extraídas. Esta medida se basa en una representación grafica de la tasa de verdaderos positivos contra la tasa de falsos positivos, y el área que encierra dicha curva es una medida de la calidad. Un área de 1 representa una calidad perfecta, en este caso significaría que el método extrajo correctamente todas las piezas de conocimiento sin extraer ninguna de más ni de menos. Un área de 0,5 representa una calidad nula, en este caso significaría que el método no logró extraer ninguna pieza de conocimiento correctamente [Bradley, 1997].

4.2. Evolución de los Métodos de Extracción de Conocimiento Creados para Web

El siguiente trabajo en el que se presentó un método de extracción de conocimiento para la Web fue el de Wu y Weld en 2010 [Wu & Weld, 2010], allí se presentan dos métodos WOE-parse y WOE-pos, el primero WOE-parse utiliza un enfoque ligeramente distinto, trabaja con un árbol de dependencias, realizando un análisis sintáctico en cada oración para extraer las relaciones, y si bien logra un mayor número de extracciones que TEXT RUNNER (1.42 tuplas por oración frente a 0.75), es 30 veces más lento que su predecesor. WOE-pos por el contrario es igual de rápido que TEXT RUNNER y ligeramente mejor (1.05 tuplas extraídas por oración). Si bien WOE-parse y WOE-pos, son métodos de propósito general su base de entrenamiento fue Wikipedia.

En [Mesquita et al., 2010] se presenta un método de extracción de conocimiento llamado SONEX pensado para extraer relaciones de redes sociales y de la *blogosfera*. Si bien el trabajo realizado es de interés, no se realizan comparaciones con otros métodos que permitan discernir si aporta alguna mejora considerable.

En [Christensen et al., 2011] se presenta un nuevo enfoque bajo las mismas consignas de extracción de conocimiento planteadas por Banko, se busca utilizar la técnica de etiquetamiento secuencial, basado en la función semántica (en inglés

Semantic Role Labeling) para la extracción de relaciones entre entidades. Se crean dos métodos nuevos: SRL-Lund y SRL-UIUC, se los compara con TEXT RUNNER en dos conjuntos de piezas de información no estructurada, uno pequeño y otro grande. Ambos demuestran ser más precisos que TEXT RUNNER, SRL-Lund obtiene una precisión de 0.7 y una medida F1 de 0.59, SRL-UIUC obtuvo una precisión de 0.63 y una medida F1 de 0.68 mientras que TEXT RUNNER obtuvo una precisión de 0.55 y una medida F1 de 0.35. Sin embargo, al trabajar con el conjunto más grande de datos de entrada, TEXT RUNNER demuestra tener una ventaja adicional, es 20 veces más rápido que SRL-LUND y 500 veces más rápido que SRL-UIUC.

En [Fader et al., 2011] Fader propone un método de extracción de conocimiento, el cual logra un área bajo la curva ROC mayor que WOE-parse, WOE-pos y que TEXT RUNNER, se trata de ReVerb. ReVerb fue puesto a prueba utilizando un conjunto de información no estructurada que constaba de 500 millones de sentencias web, demostró ser más rápido incluso que TEXT RUNNER. En un subconjunto de 100 000 sentencias se obtuvieron los siguientes tiempos: WOE-parse tardó 11 horas, WOE-pos y TEXT RUNNER tardaron 21 minutos cada uno y ReVerb 16 minutos. La mejora introducida por Fader consistió en agregar restricciones a TEXT RUNNER y centrarlo en la extracción de relaciones basadas en verbos.

La restricción de ReVerb no le permite encontrar relaciones basadas en otro tipo de categorías gramaticales que no sean verbos, es por eso que algunos autores proponen diversos métodos para extraer relaciones basadas en otro tipo de categorías gramaticales, en particular sustantivos, es el caso de OLLIE [Schmitz, 2012], de ReNoun [Yahya et al., 2014] y de TRIPLEX [Mirrezaei et al., 2015]. OLLIE es planteado directamente como una mejora a ReVerb, siendo su objetivo encontrar relaciones basadas no solo en verbos, sino también en sustantivos y adjetivos. Además analiza la posibilidad de hacer un análisis del contexto para encontrar relaciones no explicitas. OLLIE logra obtener 2.7 veces más área sobre la curva ROC que ReVerb y 1.9 veces más área bajo la curva ROC que WOE-parse, además OLLIE encuentra 4.4 veces más extracciones correctas que ReVerb y 4.8 veces más que WOE-parse [Schmitz, 2012]. El enfoque de TRIPLEX es ligeramente distinto ya que funciona como un complemento a ReVerb o a OLLIE, en el estudio realizado en [Mirrezaei et al., 2015], TRIPLEX por sí solo no logra superar a OLLIE (se compara utilizando la medida F1 en este caso) y es el uso conjunto de OLLIE más TRIPLEX el que arroja un mejor resultado, aunque no muy lejano al que arroja OLLIE por sí solo.

Para concluir en [Del Corro & Gemulla, 2013] se presenta un método de extracción de conocimiento llamado ClauseIE (respetando el paradigma propuesto por Banko); en dicho trabajo ClauseIE es comparado contra ReVerb, OLLIE, TEXT RUNNER y WOE-pos utilizando distintas fuentes de información no estructurada en formato de lenguaje natural: 500 oraciones extraídas del conjunto de datos de prueba utilizado con ReVerb en [Fader et al., 2011], 200 oraciones aleatorias extraídas de Wikipedia y 200 oraciones aleatorias extraídas del New York Times. El resultado en todos los casos fue favorable a ClauseIE quien obtuvo una mejor precisión que los demás métodos.

En la figura 1 se ilustra la evolución de los distintos métodos presentados. Figura 1 se puede ver de forma cualitativa como fue evolucionando el desempeño de los distintos métodos a lo largo del tiempo. Las comparaciones fueron obtenidas al analizar los distintos trabajos y al recoger la información allí presentada. Dicha

información consiste en datos relacionados con calidad y cantidad de las piezas de conocimiento extraídas. Como se mencionó, estos datos pueden estar representado de diversas formas: precisión, exactitud, medida-F1 o área bajo la curva ROC.

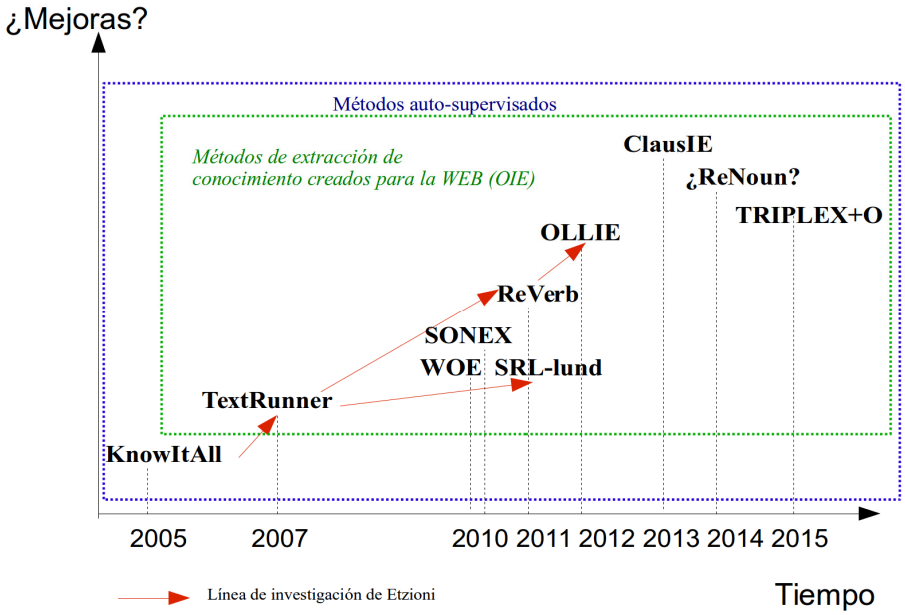


Fig. 1. En el siguiente gráfico se muestra la mejora supuesta entre los distintos métodos de extracción de conocimiento creados para la Web *versus* el tiempo, partiendo desde el trabajo de Etzioni de 2005 en donde presenta a KnowItAll.

Hay que tener en cuenta también que los métodos no siempre son comparados utilizando los mismos conjuntos de datos de entrada. Las publicaciones en donde se presentan los métodos: KnowItAll, TextRunner, ReVerb, SRL-Lund y OLLIE tienen como autor a Oren Etzioni; es decir que en el desarrollo de las comparaciones entre métodos estuvo involucrada al menos una misma persona, esta clase de continuidad sobre una línea de investigación es un fuerte indicio a favor de los datos presentados.

5. Consideraciones

En relación a la valides de las comparaciones anteriores, hay que tener en cuenta que métodos fueron comparados entre sí en un mismo ambiente controlado y cuáles no. En la Tabla 1 se muestra que método fue comparado contra que otro, indicando cual resultó mejor en dicha comparación y la publicación de referencia.

La tabla 1 es una tabla de doble entrada, en donde cada celda debe entenderse como una comparación hecha entre el método indicado en la columna contra el método indicado en la fila. En la celda se indica de manera genérica qué método logró una mayor calidad y cantidad de piezas de conocimiento extraídas, independiente-

mente de la medida utilizada en el artículo. Se indica también la referencia al artículo o los artículos de donde fue relevada la comparación.

Tabla 1. Resumen de comparaciones relevadas entre métodos

MÉTODOS	TextRunner	WOE	SRL-Lund	ReVerb	OLLIE	ClausIE	ReNoun	TRIPLEX
KnowItAll	TextRunner [1]							
TextRunner		WOE [2, 5, 9]	SRL-Lund [4]	ReVerb [5, 9]		ClausIE [9]		
WOE				ReVerb [5, 9]	OLLIE [6, 9]	ClausIE [9]		
SRL-Lund					SRL-Lund [6]			
ReVerb					OLLIE [6, 8], ReVerb [9]	ClausIE [9]		TRIPLEX, TRIPLEX + ReVerb [8]
OLLIE						ClausIE [9]		OLLIE, TRIPLEX + OLLIE [8]
ClausIE								
ReNoun								
TRIPLEX								

Referencias: 1. [Banko et al., 2007], 2. [Wu & Weld, 2010], 3. [Mesquita et al., 2010], 4. [Christensen et al., 2011], 5. [Fader et al., 2011], 6. [Schmitz et al., 2012], 7. [Yahya et al., 2014], 8. [Mirrezaei et al., 2015], 9. [Del Corro & Gemulla, 2013]

Para el caso de TRIPLEX, se indica además la comparación realizada tomando de forma conjunta TRIPLEX más el segundo método, ya que cómo se indicó TRIPLEX se crea como un método suplementario a ReVerb u OLLIE.

Se puede ver que algunos métodos fueron ampliamente comparados, es el caso de TextRunner con 8 comparaciones totales realizadas, de WOE con 8 comparaciones totales realizadas, de ReVerb con 9 comparaciones totales realizadas y de OLLIE con 8 comparaciones totales realizadas. El caso opuesto es el de ReNoun el cual no fue comparado con ningún otro método. Es interesante observar el caso de la comparación de OLLIE con ReVerb hay publicaciones [Schmitz et al., 2012; Mirrezaei et al., 2015] que concluyen que OLLIE es superior a ReVerb y otra [Del Corro & Gemulla, 2013] que concluye que ReVerb obtiene mejores resultados que OLLIE. Por último las comparaciones que se realizaron contra ClauseIE pertenecen todas a la publicación de [Del Corro & Gemulla, 2013]

6. Conclusiones

Del análisis presentado e esta comunicación sobre distintas distintos métodos de extracción de conocimiento creados para la Web (es decir para grandes *corpus* de documentos sin un dominio definido) se puede formular las siguientes conclusiones preliminares:

[i] El mejor de los métodos estudiados en términos de cantidad y calidad de piezas de conocimiento extraído es ClauseIE. No se registran comparaciones con ReNoun ni contra TRIPLEX.

[ii] Dado que TRIPLEX cuando es utilizado en combinación con OLLIE es apenas un poco mejor que OLLIE por sí solo, sería esperable que ClauseIE lo superase en precisión.

[iii] Respecto ReNoun no hay evidencia suficiente para deducir si su desempeño estará por encima o no de ClauseIE.

[iv] A continuación de ClauseIE se ubicarían los métodos: OLLIE, ReVerb y WOE, en ese orden (según su calidad), aunque según el caso de prueba utilizado alguno podría superar a otro.

Queda como trabajo futuro realizar una comparación de todos los métodos relevados para poder completar las comparaciones faltantes, unificando además los criterios para la evaluación del desempeño y la calidad de los métodos. En dicho trabajo, se tendrán en cuenta los distintos conjuntos de datos utilizados como entrada, ya que estos generarán diferentes casos de usos, capaces de aportar evidencia sobre las diferencias observadas en el desempeño de los métodos, como la mencionada entre OLLIE y ReVerb.

Financiamiento

Las investigaciones que se reportan en este artículo han sido financiadas parcialmente por los Proyectos UNLa-33B145, UNLa-33B133 y UNLa-33A205 de la Secretaria de Ciencia y Tecnología de la Universidad Nacional de Lanús.

Referencias

Argimón J. 2004. Métodos de Investigación Clínica y Epidemiológica. Elsevier España, S.A. ISBN 9788481747096.

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007, January). Open information extraction for the web. In IJCAI (Vol. 7, pp. 2670-2676).

Bradley, Andrew P. The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern recognition, 1997, vol. 30, no 7, p. 1145-1159.

Christensen, J., Soderland, S., & Etzioni, O. (2011, June). An analysis of open information extraction based on semantic role labeling. In Proceedings of the sixth international conference on Knowledge capture (pp. 113-120). ACM.

Cowie, J., & Lehnert, W. (1996). Information extraction. Communications of the ACM, 39(1), 80-91.

Culotta A, McCallum A, Betz J (2006) Integrating probabilistic extraction models and data mining to discover relations and patterns in text. In: Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Association for Computational Linguistics, New York, New York, pp 296–303

Del Corro, L., & Gemulla, R. (2013, May). ClausIE: clause-based open information extraction. In Proceedings of the 22nd international conference on World Wide Web (pp. 355-366). International World Wide Web Conferences Steering Committee.

Etzioni O, Banko M, Soderland S, Weld DS (2008) Open information extraction from the web. Communications of the ACM 51:68–74

Etzioni, O., Cafarella, M., Downey, D., Popescu, A. M., Shaked, T., Soderland, S., ... & Yates, A. (2005). Unsupervised named-entity extraction from the web: An experimental study. Artificial intelligence, 165(1), 91-134.

Fader, A., Soderland, S., & Etzioni, O. (2011, July). Identifying relations for open information extraction. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 1535-1545). Association for Computational Linguistics.

García-Martínez, R. & Britos, P. V. (2004). Ingeniería de sistemas expertos. Nueva Librería. ISBN 987-1104-15

Gómez, A., Juristo, N., Montes, C., & Pazos, J. (1997). Ingeniería del conocimiento. Editorial Centro de Estudios Ramón Areces. ISBN 84-8004-269-9.

Jurafsky, D., & Martin, J. H. (2000). Speech and language processing an introduction to natural language processing, computational linguistics, and speech recognition, 2nd edn. Prentice-Hall, Inc.

Konstantinova, N. (2014). Review of Relation Extraction Methods: What Is New Out There?. In Analysis of Images, Social Networks and Texts (pp. 15-28). Springer International Publishing.

Mesquita, F., Merhav, Y., & Barbosa, D. (2010). Extracting information networks from the blogosphere: State-of-the-art and challenges. In Proceedings of the Fourth AAAI Conference on Weblogs and Social Media (ICWSM), Data Challenge Workshop.

Mirrezaei, S. I., Martins, B., & Cruz, I. F. (2015). The Triplex Approach for Recognizing Semantic Relations from Noun Phrases, Appositions, and Adjectives. In The Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data (Know@LOD) co-located with Extended Semantic Web Conference (ESWC), Portoroz, Slovenia.

Modha, D., Ananthanarayanan, R., Esser, S., Ndirango, A.., Sherbondy, A.., Singh, R. 2011. Cognitive Computing. Communications of the ACM, 54(8): 62-71.

Schmitz, M., Bart, R., Soderland, S., & Etzioni, O. (2012, July). Open language learning for information extraction. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (pp. 523-534). Association for Computational Linguistics.

Wu, F., & Weld, D. S. (2010, July). Open information extraction using Wikipedia. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (pp. 118-127). Association for Computational Linguistics.

Yahya, M., Whang, S. E., Gupta, R., & Halevy, A. (2014, October). Renoun: Fact extraction for nominal attributes. In Proc. 2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar.

Mejora de la mantenibilidad con un modelo de medición de la calidad: resultados en una gran empresa

Emanuel Irrazábal

Departamento de Informática. FaCENA.
Universidad Nacional del Nordeste
Av. Libertad 3400, Ciudad de Corrientes, Corrientes
Facultad de Ingeniería y Tecnología
Universidad de la Cuenca del Plata
Lavalle 50, 3400, Ciudad de Corrientes, Corrientes
emanuelirrazabal@gmail.com

Abstract. Ante un mercado cada vez más competitivo y en constante desarrollo, la calidad del software está tomando mayor importancia en las organizaciones, y con ello, la calidad del producto software, medida directamente en el código fuente. En un conjunto de artículos que el autor ha publicado anteriormente se presentó una recopilación de herramientas abiertas para el análisis del código fuente y la correspondencia entre las métricas proporcionadas con la mantenibilidad. Junto con ello se ha trabajado en un modelo de medición de la mantenibilidad del código fuente en base a las mediciones de las herramientas de análisis estático. En este artículo se presentan los resultados de la utilización del modelo de medición y de las herramientas durante diez meses en una gran empresa española. Se detalla la evolución de las mediciones, la satisfacción de los desarrolladores y como han ido evolucionandolas incidencias encontradas en las pruebas de aceptación.

Keywords: mantenibilidad, mejora, gran empresa, ISO 25010, herramientas

1 Introducción

Ante un mercado cada vez más competitivo y en constante desarrollo, la calidad del software está tomando mayor importancia en las organizaciones, y con ello, la medición software [1] ha adquirido mayor relevancia como principal herramienta para medir la calidad del software. Esta no sólo tiene influencia en los costos finales, sino que es también un factor clave para mejorar la imagen frente a los clientes y como elemento diferenciador de la competencia [2].

Aunque la calidad puede describirse desde diferentes perspectivas, la calidad software tradicionalmente se divide en calidad del producto y proceso [3]. Es decir, la calidad en el proceso de desarrollo para obtener el producto (actividades, tareas, etc. del desarrollo y mantenimiento del software) y la calidad del propio producto software. Debido a ello, se ha incrementado la tendencia hacia la institucionalización de buenas prácticas de calidad en organizaciones de desarrollo software. Dicha

tendencia se ha visto reflejada en un aumento constante del número de organizaciones que tienen la intención de alinear sus procedimientos con modelos de referencia de proceso ampliamente conocidos, como CMMI-DEV [4] o ISO/IEC 12207 [5].

Pero un proceso institucionalizado no necesariamente concluye con un producto de calidad. De hecho, son muchas las críticas que argumentan que estudiar solo la calidad del proceso puede llevarnos a obtener una visión parcial de la calidad software [6]. Puede ocurrir que la estandarización de los procesos y su uso incorrecto incluso institucionalicen la creación de malos productos. En esta línea, Maibaum y Wassyng [7], comentaban que las evaluaciones de calidad deberían estar basadas en evidencias extraídas directamente de los atributos del producto, no en evidencias circunstanciales deducidas desde el proceso.

Teniendo esto en cuenta, existen diferentes modelos de referencia para la calidad del producto. Sin embargo, es difícil llevar a la práctica la implantación de estos modelos. La principal barrera viene del carácter abierto de estas normas. Esto es, no muestran detalles sobre qué métricas utilizar o cuáles son las más adecuadas; cómo agrupar los valores de las métricas para obtener métricas de más alto nivel; o cuáles son los valores umbrales recomendados para cada métrica.

Con la intención de paliar los efectos de esta situación la Organización Internacional para la Estandarización (ISO) ha publicado una serie de estándares relacionados con la calidad del producto. Uno de estos estándares o normas es la ISO/IEC 9126:1991 Software Engineering – Product Quality [8]. El principal objetivo de esta norma es proporcionar un marco de referencia para definir y evaluar la calidad del producto software. Para ello la norma especifica un modelo de calidad identificando un conjunto de características de calidad. Así mismo, en 2005 se publica la serie de estándares ISO/IEC 25000. Esta serie, también denominada Software product Quality Requirements and Evaluation (SQuaRE) [9], la conforman un conjunto de normas internacionales orientadas a la calidad del producto; teniendo como objetivo organizar y unificar los estándares que especifican los requerimientos de calidad del software y su evaluación. Más recientemente, en marzo del 2011 se publica la norma ISO/IEC 25010 [10], que reemplaza a la norma ISO/IEC 9126 y define un modelo de calidad para los productos software.

Una de las características definidas en el modelo de calidad del producto de la norma ISO/IEC 25010 es la mantenibilidad. Históricamente la mantenibilidad ha sido reconocida como una de las características más relevantes del software debido a su impacto directo sobre el costo de desarrollo y el propio mantenimiento. De hecho, estudios previos señalan la fase de mantenimiento como la fase donde más recursos se invierten en el ciclo de vida del software, implicando dos veces el coste de la fase de desarrollo [11][12][13][14][15]. Por ejemplo, según [14], el tiempo que un programador invierte en mantenimiento es alrededor del 61% frente al 39% dedicado al desarrollo. Y en ocasiones la proporción de costo que supone el mantenimiento ronda el 90% [16].

En ese sentido, el modelo de calidad del producto descrito en la norma ISO/IEC 25010 no proporciona un conjunto consensuado de métricas concretas para calcular la mantenibilidad. Aun así, establece las propiedades que deberían tener estas métricas: han de ser objetivas, reproducibles, independientes, expresadas mediante escalas válidas y suficientemente precisas para realizar comparaciones fiables. Teniendo estas premisas en mente, y el hecho de que en la actualidad los desarrollos software suelen

tener un gran tamaño, con lo que el número de medidas a obtener es muy elevado, parece que el enfoque más conveniente sería disponer de herramientas que facilitaran la obtención automática de tales medidas.

Por todo ello, en un conjunto de artículos publicados anteriormente hemos presentado: una recopilación de herramientas abiertas para el análisis del código fuente, sus métricas y la correspondencia entre estas y la mantenibilidad [17][18][19]. Y un modelo de medición de la mantenibilidad que construye una medición abstracta de la mantenibilidad del código fuente en base a las mediciones de las herramientas comentadas en el punto anterior [20][21].

Como paso final de esta serie de trabajos aquí se presentan los resultados de la utilización del modelo de medición durante diez meses en una gran empresa española. Y, especialmente, se discuten los resultados de incorporar herramientas de análisis estático del código fuente en los equipos de desarrollo software de una empresa.

En la sección 2 de este artículo se presenta la estructura del modelo de medición. En la sección 3 se indican las herramientas y métricas utilizadas. En la sección 4 se detallan los resultados obtenidos en una gran empresa española. Por último, en la sección 5 se discuten las conclusiones finales.

2 El Modelo de Medición de la Mantenbilidad

En esta sección se describe brevemente la estructura elegida para el desarrollo de la herramienta implantada en la empresa. Para poder obtener valores indicadores de la calidad del producto software, es necesario establecer unos parámetros sobre los que realizar dicha medición. Para ello, el entorno metodológico de la herramienta se basa en la norma ISO/IEC 9126 e ISO/IEC 25010, que establecen unos atributos a partir de los cuales se obtendrán valores indicadores de calidad. En artículos anteriores se describe en detalle [20][21].

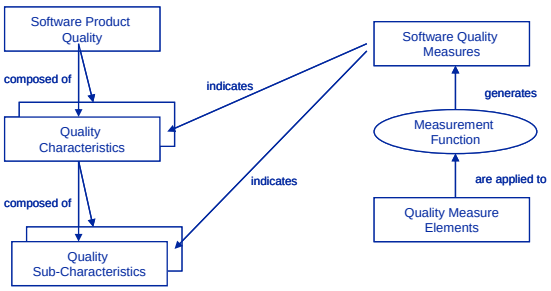


Fig. 1. Modelo de referencia de medición de la calidad del producto software (ISO/IEC 25000).

El valor de estas medidas de calidad software se obtiene por la aplicación de una función de medida (Measurement Function) a los elementos de medida de calidad (Quality Measure Elements). Aunque las normas ISO/IEC 9126 y 25010 establecen cuáles son las características de la calidad de un producto software y sus subcaracterísticas, no especifica qué medidas pueden utilizarse como indicador de una

subcaracterística (ver Fig. 1). Así, es necesario relacionar atributos del código fuente respecto de las diferentes características de calidad del producto software. Obtener una medición significa, por tanto, obtener el valor de un atributo. Si se trata de un atributo base, su valor se obtendrá leyendo el informe proporcionado por la herramienta que lo midió, y si es compuesto se calculará a través de una función de medición

3 Las Herramientas Seleccionadas y el Modelo de Medición de la Mantenibilidad Construido

A continuación se resumen las herramientas de medición y su relación con las características de calidad de la norma ISO 9126 e ISO 25010. Esto ha dado como resultado el modelo de medición de la mantenibilidad utilizado en la mejora de la empresa. La Tabla 1enumera las herramientas libres de análisis del código fuente y evaluación de pruebas unitarias seleccionadas en otros trabajos [17][18][19]. Se indica el nombre de la herramienta y las métricas básicas que pueden obtenerse.

Tabla 1. Selección final de herramientas y métricas básicas relacionadas.

Herramientas	Métricas
JavaNCSS	Sentencias de código fuente no comentadas (NCSS) Complejidad ciclomática (CC)
PMD/CPD	Evaluación del código fuente (posibles errores, código no utilizado, código duplicado)
CheckStyle	Validación de estilos
FindBugs	Evaluación del código fuente (posibles errores)
JDepend	Dependencias cíclicas (CDC) Acoplamiento aferente, acoplamiento eferente (Ca, Ce)
CCCC	Número de módulos (NOM) Líneas de código (LOC) Líneas de comentarios (COM) Complejidad Ciclomática (MVG) Peso del método por clase (WMC) Fan-In, Fan-In concreteness, Fan-Out visibility, Fan-Out (…) (FI,FIc,FIv,FO,FOc,FOv) Líneas de código (LOC) Líneas de comentarios (COM)
StyleCop	Validación de estilos
FxCop	Evaluación del código fuente (posibles errores)
Junit	Número de pruebas unitarias (UT)
CPPUnit	Número de pruebas unitarias (UT)
Nunit	Número de pruebas unitarias (UT)
EMMA	Cobertura de pruebas unitarias por paquetes (CBP) Resumen de cobertura de pruebas unitarias (OCS)

A continuación se muestra un resumen el modelo de medición de la mantenibilidad desarrollado en [20][21]. El primero de los elementos configurables del modelo de medición son las medidas de calidad software a partir de las cuales se obtienen los valores indicadores de calidad de las características y subcaracterísticas. La selección de medidas de calidad para la característica de mantenibilidad se muestra en la Tabla 2.

Tabla 1. Relación de medidas de calidad del modelo de medición respect de la mantenibilidad y sus sobcaracterísticas (ISO 9126).

Característica	Subcaracterística	Medida de calidad

Mantenibilidad	Analizabilidad	Densidad de complejidad ciclomática
		Densidad de código repetido
		Densidad de comentarios
		Densidad de defectos de la capacidad para ser analizado
	Cambiabilidad	Densidad de defectos de la capacidad para ser cambiado
		Densidad de ciclos
	Estabilidad	Densidad de defectos de estabilidad
	Capacidad de ser probado	Densidad de defectos en pruebas unitarias
		Densidad de pruebas unitarias
		Cobertura de pruebas unitarias
		Tasa de errores de pruebas unitarias

4 Análisis de los Resultados al Implantar el Modelo de Medición

En este apartado se analizarán los resultados de la implantación y uso continuado del modelo de medición de la mantenibilidad. La implementación se realizó en una gran empresa española del sector público, con desarrollos en Java. .Se analizó la mejora de la mantenibilidad y se lo comparó con la evolución de las incidencias detectadas durante el desarrollo de diferentes proyectos.

Como se puede observar en la Figura 2, durante el primer mes la analizabilidad era bastante mala (en el orden de un 30%). Se realizó un programa detallado de mejora y se planificaron formaciones para aprender la utilización de las herramientas de análisis estático de código fuente. Por otro lado se comunicaron los objetivos y la manera en la que el modelo de medición penalizaba la ausencia de comentarios y las violaciones de las reglas implantas en las herramientas. Como resultado de estos cambios y la comunicación constante de los resultados se pudo mejorar la analizabilidad (especialmente debido a que las herramientas como PMD o FindBugs contienen ejemplos, explicaciones y otras características didácticas para favorecer el aprendizaje de los desarrolladores.

De igual manera, la subcaracterística de cambiabilidad ha tenido un período prolongado de estancamiento (unos 3 meses) hasta llegar a valores aceptables (superiores incluso al 80%) y mejores que la analizabilidad. En esos tres primeros meses se realizaron refactorizaciones para romper dependencias cíclicas y disminuir la complejidad ciclomática. A partir del tercer mes los nuevos desarrollos fueron adecuándose a las nuevas características de codificación.

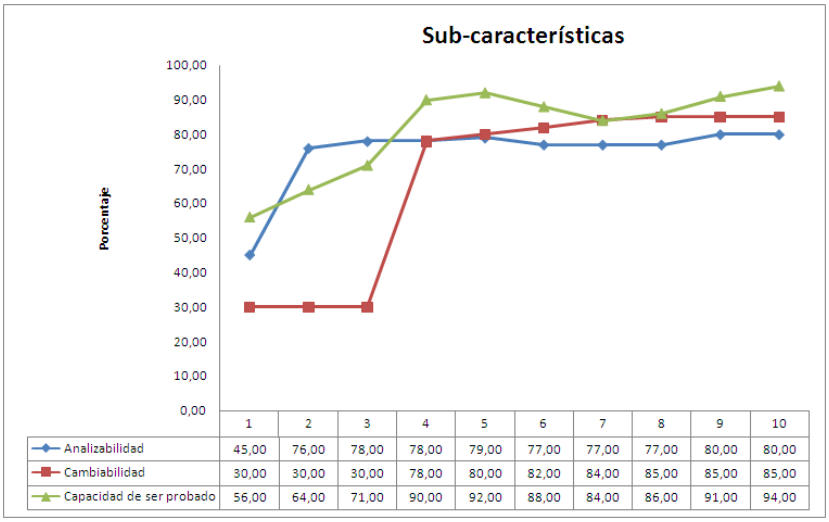


Fig. 2. Evolución de las mediciones. Subcaracterísticas de la mantenibilidad.

En el caso de la capacidad de ser probado el código fuente, existía ya una cultura de utilización de pruebas unitarias. En ese sentido, el problema era la cobertura de las pruebas unitarias y el tratamiento de los errores. Se consiguieron buenos resultados desde el inicio, incluso superando en los meses 4 y 5 el 90%. Durante los meses 7 y 8, al incorporarse nuevos desarrollos, los recursos necesarios para la realización de pruebas disminuyó.

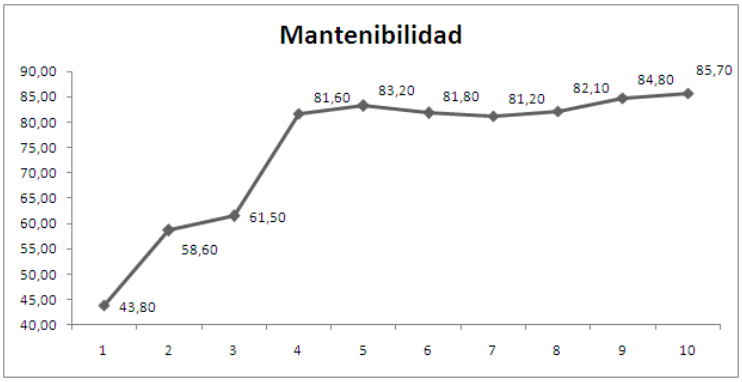


Fig. 3. Evolución de las mediciones. Característica de mantenibilidad

Como puede verse en la Figura 3, la mantenibilidad alcanzó registros aceptables (mayor de 80%) a partir del cuarto mes. Y se mantuvo en la franja de entre el 80% y el 85% en los siguientes seis meses. También en esta gráfica se advierte una pequeña disminución de la mantenibilidad, durante los meses 6, 7 y 8 debido al inicio del

desarrollo de nuevos módulos. Una de las ventajas evidentes al implementar las herramientas de análisis estático del código fuente ha sido la mejora en el nivel de conocimiento y productividad de los desarrolladores. Al final del quinto y del décimo mes se realizó una encuesta donde se preguntaba el grado de satisfacción al utilizar las herramientas de calidad.

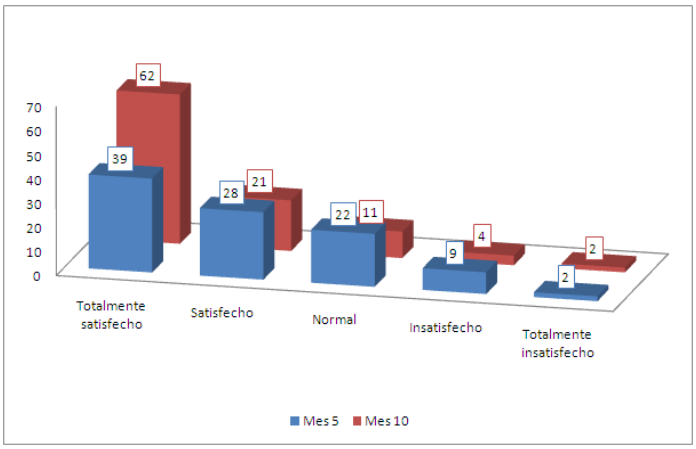
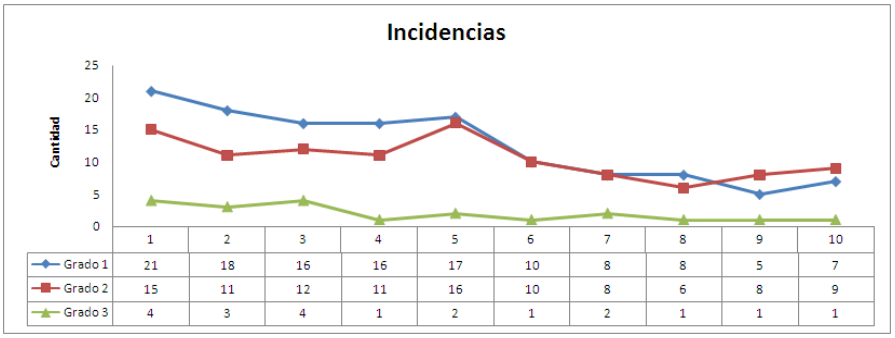


Fig. 4. Resultado de la encuesta de satisfacción realizada a los desarrolladores

Los resultados, que pueden verse en la Figura 4, muestran cómo se pasó de un 39% a un 62% de desarrolladores totalmente satisfechos. Así mismo, los desarrolladores no decididos respecto de estas buenas prácticas pasaron de un 22% a un 11%. Uno de los aspectos más importantes a destacar es que el 2% de los desarrolladores estaban completamente insatisfechos con las nuevas prácticas, permaneciendo esto hasta el décimo mes. Continuando con el análisis de los resultados se recopilaron datos de las incidencias encontradas y resueltas durante los diez meses de utilización de las herramientas de calidad. En este caso se ha considerado como incidencia cualquier error detectado en las pruebas de aceptación, ocasionado por problemas en el código fuente. Se descartaron los errores que provienen de una mala interpretación de los requisitos, y que, por otra parte, estuviesen bien codificados. Los errores se califican por grado, dependiendo el tiempo en que se puede resolver y otros aspectos técnicos que serán detallados a continuación:

- Grado 1: identificados con precisión y que pueden ser corregidos y probados en el día.
- Grado 2: errores que afectan más de una capa (lógica de negocio, comunicación, base de datos, etc.) y requiere más de 1 día de trabajo.
- Grado 3: errores de grado 2 que impiden el desarrollo y/o mantenimiento de otras partes del código fuente.

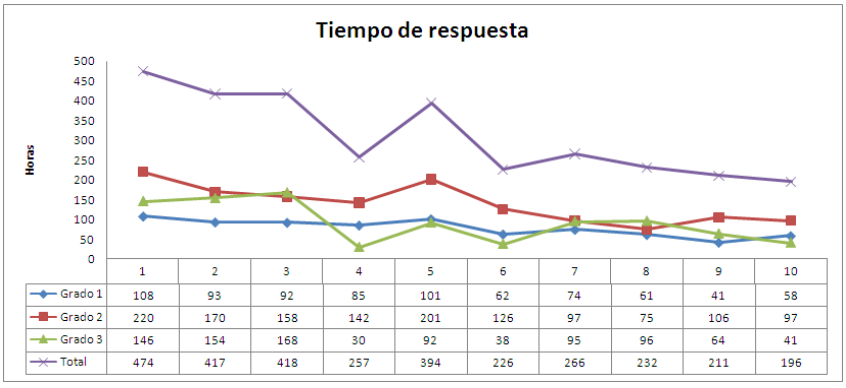
A continuación, y analizando el gestor de tareas de la empresa (utilizada desde antes de la implementación de las herramientas de calidad) se graficó la evolución del número de incidencias.



	1	2	3	4	5	6	7	8	9	10
Grado 1	21	18	16	16	17	10	8	8	5	7
Grado 2	15	11	12	11	16	10	8	6	8	9
Grado 3	4	3	4	1	2	1	2	1	1	1

Fig. 5. Evolución de las incidencias

Como se puede ver en la Figura 5, las incidencias han disminuido. Es importante recalcar aquí que no se aumentaron las incidencias en los tres primeros meses, aunque se realizaron gran cantidad de refactorizaciones para disminuir el acoplamiento entre clases. Las incidencias del tipo 1 y 2 tendieron a equipararse. Las conclusiones obtenidas con el departamento de calidad de la empresa indican que la disminución de incidencias del tipo 1 ha posibilitado la profundización en las inspecciones, y, por lo tanto, el descubrimiento de otro tipo de problemas. Esto se ha considerado ampliamente beneficioso.



	1	2	3	4	5	6	7	8	9	10
Grado 1	108	93	92	85	101	62	74	61	41	58
Grado 2	220	170	158	142	201	126	97	75	106	97
Grado 3	146	154	168	30	92	38	95	96	64	41
Total	474	417	418	257	394	226	266	232	211	196

Fig. 6. Evolución de los tiempos de respuesta de las incidencias.

En la Figura 6 se indican los tiempos de respuesta totales para cada tipo de incidencia. La tendencia general de los tiempos de respuesta ha ido en disminución. En el mes 5 se ha notado un aumento considerable (acompañando la cantidad de incidencias, vistas anteriormente, con la cantidad de horas invertidas en solucionarlo).

En este sentido, y analizándolo con el departamento de desarrollo de la empresa se ha concluido que las causas estaban relacionadas con la inclusión de nuevos desarrollos realizados por personal externo. Estos nuevos recursos no tuvieron la misma formación inicial; por el contrario se realizó una formación a distancia supervisada por los desarrolladores de la empresa cliente.

5 Conclusiones

La implementación del modelo de medición de la mantenibilidad se realizó en una gran empresa española del sector público, con desarrollos en Java. .Se analizó la mejora de la mantenibilidad y se lo comparó con la evolución de las incidencias detectadas durante el desarrollo de diferentes proyectos. Como primera conclusión y una de las ventajas evidentes al implementar las herramientas de análisis estático del código fuente ha sido la mejora en el nivel de conocimiento y productividad de los desarrolladores.

Se realizó un programa detallado de mejora y se planificaron formaciones para favorecer el aprendizaje de las herramientas. Por otro lado se comunicaron los objetivos y la manera en la que el modelo de medición penalizaba la ausencia de comentarios y las violaciones de las reglas implantas en las herramientas. Como resultado de ello, la mantenibilidad alcanzó registros aceptables (mayor de 80%) a partir del cuarto mes.

En cuanto a la percepción por parte del personal, se pasó de un 39% a un 62% de desarrolladores totalmente satisfechos. Así mismo, los desarrolladores no decididos respecto de estas buenas prácticas pasaron de un 22% a un 11%. Finalmente, la cantidad de incidencias ha disminuido, así como el tiempo de respuesta promedio.

La implantación de herramientas de análisis estático de código fuente en la empresa no sólo ha demostrado que proporciona visibilidad con sus indicadores; sino que gracias a ser soportado por un modelo de medición facilitó la personalización de las herramientas a las necesidades de la empresa. E incluso a las necesidades de distintos proyectos dentro de la propia empresa.

Como trabajo futuro se espera continuar con el estudio de la mantenibilidad desde el punto de vista de la deuda técnica, enfocando las mediciones a la manera en la cual evolucionan los intereses de la deuda.

Agradecimientos

Agradezco al proyecto F007-2009 “Modelos y métricas para la evaluación de la calidad del software” de la SECYT-UNNE. Y agradezco al grupo Kybele de la Universidad Rey Juan Carlos con quienes he trabajado en los primeros trabajos de esta línea de investigación.

Referencias

[1] Rifkin, S. (2009) Guest Editor's Introduction: Software Measurement. IEEE Software, 26, p. 70.

[2] Piattini, M., et al. (2008) Medición y estimación del software: técnicas y métodos para mejorar la calidad y productividad del software, Ra-ma.

[3] Ebert C. (2009) Guest Editor's Introduction: How Open Source Tools Can Benefit Industry. IEEE Software, 26, p: 50-51.

[4] SEI, Process Maturity Profile (2001) CMMI v1.2, SCAMPI v1.2, Class A Appraisal Results. 2011 Mid-Year Update, Septiembre 2011, Software Engineering Institute.

[5] ISO, ISO/IEC 12207:2008. Systems and software engineering - Software life cycle processes, in International Organization for Standarization.

[6] Kitchenham, B. and S.L. Pfleeger (1996) Software Quality: The Elusive Target. IEEE Software, 20(1), p: 12-21.

[7] Maibaum T, W.A. (2008) A Product-Focused Approach to Software Certification.

[8] ISO, Software Product Evaluation–Quality Characteristics and Guidelines for their Use. ISO/IEC Standard 9126, International Organization for Standarization.

[9] ISO, ISO/IEC 25000 Software and system engineering – Software product Quality Requirements and Evaluation (SQuaRE) –Guide to SQuaRE, in International Organization for Standarization. 2005: Ginebra, Suiza.

[10] ISO, ISO/IEC 25010 Software and system engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Quality model and guide. International Organization for Standarization. . 2005: Ginebra, Suiza.

[11] Frazer, A. (1992) Reverse engineering- hype, hope or here? Software Reuse and Reverse Engineering in Practice, p: 209-243.

[12] Pressman, R., (2002) Software Engineering: A Practitioner's Approach., Madrid: McGraw-Hill.

[13] Saiedian, H. and N. Carr (1997) Characterizing a software process maturity model for small organizations. , in ACM SIGICE Bulletin, p. 2-11.

[14] Janice, S. (1998) Practices of Software Maintenance, in Proceedings of the International Conference on Software Maintenance, IEEE Computer Society.

[15] Harrison, W. (2005) What Do Software Developers Need to Know about Business? IEEE Software, 22(5), p: 5-7.

[16] Erlikh L. (2000) Leveraging Legacy System Dollars for E-business. IT Professional, 2, 3, pp: 17-23.

[17] Irrazabal, E., Vara, J. M., Garzás, J., Santa Escolástica, R., & Marcos, E. (2010, June). Alignment of Open Source Tools with ISO norms for software product quality. In SOFTWARE MEASUREMENT EUROPEAN FORUM (p. 135).

[18] Irrazábal, E., Garzás, J., & Marcos, E. (2011). Alignment of Open Source Tools with the New ISO 25010 Standard-Focus on Maintainability. In ICSOFT (2) (pp. 111-116).

[19] Irrazábal, E., & Garzás, J. (2010). Análisis de métricas básicas y herramientas de código libre para medir la mantenibilidad. Innovación, Calidad e Ingeniería del Software, 6(3), 56.

[20] Quevedo A, Irrazábal E, Enríquez de Salamanca J, Vara J. M., Garzás J. Entorno de calidad KEMIS Congreso: XII QUATIC 2010, Industrial Track, . pp. 122-136; ISBN: 978-989-96867-0-0Oporto, (Portugal). 29 de Septiembre de 2010.

[21] Irrazábal, E. (2012). Construcción de un Entorno para la Medición Automatizada de la Calidad de los Productos Software.

Características funcionales avanzadas de los BPMS: Análisis comparativo de herramientas.

Anahí S. Rodríguez[1],Patricia Bazan[1], F. Javier Diaz[1]
1- LINTI – Facultad de Informática - UNLP
anahi@linti.unlp.edu.ar, pbaz@info.unlp.edu.ar, javier.diaz@linti.unlp.edu.ar.

Resumen. Los sistemas de gestión de procesos de negocio (BPMS - *Business Process Management Systems*) facilitan el modelado, despliegue, ejecución y monitorización de los procesos de negocio. En la actualidad existen diversas implementaciones de tales herramientas y su uso se encuentra medianamente difundido, pero como toda solución de software requiere revisarse a la luz de nuevos modelos computacionales, nuevas exigencias de la industria y nuevos modelos de servicio. En este trabajo se analizaron las siguientes características funcionales no clásicas: Cobertura de las etapas del ciclo de vida de los procesos, Mecanismos de actualización del modelo de procesos de negocio, Mecanismos de actualización de instancias de proceso, Capacidad de distribución de procesos en varios motores, Integración con portales de autenticación SSO (Single Sign On) y Cumplimiento con Estándar BPMN. Dichas características son definidas y analizadas sobre las herramientas *BonitaSoft* y *Bizagi*.

Palabras clave: *Procesos de Negocio - BPMS – Instancia de proceso – Procesos Distribuidos*.

1 Introducción

La gestión de los procesos de negocio se realiza a través de un sistema de gestión de procesos de negocio (BPMS). Las herramientas BPMS sirven para realizan un modelado del proceso de negocio y realizar un seguimiento en su ejecución, generando un proceso ejecutable, entre otras características. Siguiendo todos los flujos de ejecución que representan el trabajo realizado. Pueden existir varias instancias de la ejecución del proceso de negocio algunas exitosas y otras fallidas. Así algunas herramientas permiten realizar un seguimiento y monitorización del proceso de negocio y con esto las empresas pueden mejorar y optimizar sus procesos [1] [2].

Existen una variedad muy grande de BPMS open source, libres y propietarias [3], en el presente trabajo se seleccionaron dos herramientas populares para ser utilizadas con fines educativos y de investigación por las licencias que proporcionan, las cuales son *BonitaSoft* [4] y *Bizagi* [5].

En el presente trabajo se analizaron herramientas BPMS sobre las características definidas en un trabajo anterior [6]. El artículo se organiza de la siguiente manera: en

la sección 1.1 se describen definiciones, marco conceptual, y análisis de la bibliografía actual. En la sección 2 se presentan las características definidas en un trabajo previo [6]. En las secciones 3 a 5 se presentan las herramientas analizadas, el caso de uso de ejemplo y la tabla comparativa de las mismas. Para finalizar se presenta una conclusión del uso de las herramientas.

1.1 Definiciones y marco conceptual

Un proceso de negocio, según Mathias Weske, es un conjunto de actividades que se realizan en coordinación en un ambiente organizacional y técnico. Estas actividades alcanzan el objetivo del negocio. Cada proceso de negocio representa una única organización, pudiendo interactuar con otras organizaciones externas [7].

BPM (Business Process Management), según Weske "...a principios de la década de los '90 se propuso una nueva forma de organización en las empresas basándose en los procesos de negocio. La administración de procesos de negocios incluye conceptos, métodos y técnicas para apoyar el diseño, administración, configuración, promulgación y análisis de los procesos de negocios. La base de la administración de procesos de negocio es la representación explícita de los procesos de negocio con sus actividades y sus restricciones de ejecución. Una vez que se definen los procesos de negocios, pueden ser analizados, mejorados y promulgados. Tradicionalmente, los procesos de negocio son llevados a cabo manualmente, guiados por el conocimiento del personal de la empresa y asistido por las regulaciones y procedimientos establecidos por la organización. Las empresas pueden obtener beneficios adicionales al usar sistemas de software para coordinar las actividades involucradas en el proceso de negocios. Este software es el llamado sistema de administración de procesos de negocio…" [7].

El ciclo de vida de los procesos de negocios se compone de fases cíclicas que no implican necesariamente un orden temporal, pero sí una dependencia lógica, según lo define Weske. Las fases del ciclo de vida que contemplamos en el presente trabajo para su análisis en las herramientas analizadas son: Diseño y análisis - Configuración (Definición) - Promulgación (Ejecución) - Evaluación (Monitorización) [7] [8].

Los BPMS proporcionan mecanismos para mantener el ciclo de vida de los procesos de negocio y obtener la mejora continua. Los analistas de negocio pueden desarrollar sus análisis y construir modelos de procesos sin saber cómo se realiza su ejecución. Los encargados de diseñar y ejecutar los sistemas de información son el personal técnico de TI. Contar con una herramienta informática para dicho ciclo de modelado, despliegue y monitorización permite recolectar los rastros de ejecución para retroalimentar el ciclo de vida del proceso y así tener una mejora continua.

Entre los aspectos funcionales clásicos provistos por los BPMS se pueden mencionar [9] [6]:

- Obtener un gráfico que puede ser de utilidad para los dueños del negocio, como a los analistas para conocer el flujo de trabajo.
- Simular el proceso de negocio, pudiendo utilizarlo como prueba con datos actuales e históricos.
- Proporcionar la facilidad de crear automáticamente interfaces y reportes.

- Proporcionar la facilidad de crear reglas de negocio, pudiendo ser utilizadas para conducir el flujo del proceso y la toma de decisiones.

- La capacidad para la integración con otras herramientas externas, dado que muchas veces el BPMS no provee todo lo necesario para el ciclo de vida de los procesos de negocio.

- La capacidad de enviar o recibir mensajes de eventos del sistema o del negocio.

- La capacidad de reconocer indicadores de desempeño, que se pueden obtener de la ejecución de los procesos de negocio.

Considerando las componentes de un BPMS que llevan a cabo las características funcionales antes mencionadas, se puede concluir que los BPMS deben contemplar, al menos, las siguientes fases en la ejecución del proceso de negocio [10] [6]:

- Modelado del proceso - El modelado de proceso implica la definición de tareas, ordenarlas, su ramificación, definición de recursos, y otros aspectos del proceso. Con esto podemos tener un modelado del proceso. Este modelo es representado por una gráfica de un flujo y un ordenamiento de tareas involucradas.

- Instanciación del proceso - La instancia de un proceso de negocio es cada despliegue del modelo de procesos en tiempo de ejecución. Cada modelo pueden tener diferentes instancias una por cada vez que el proceso es iniciado con un caso en particular.

- Ejecución del proceso – La ejecución del proceso se lleva a cabo dentro del entorno de ejecución donde se interpreta el modelo del proceso, lo inicializa y lo ejecuta de acuerdo a la información de entrada. Generalmente el entorno de ejecución incluye un motor de procesos, y posiblemente un repositorio para los mismos.

- Monitorización del proceso –Al monitorizar el proceso de negocio permite recolectar rastros de ejecución para proponer una mejora en el rendimiento del proceso de negocio [11] [12].

2 Aspectos Funcionales No Clásicos en los BPMS

Dentro de la línea de investigación que se viene desarrollando, se han definido aspectos funcionales no clásicos que un BPMS debe incluir, tal como se presentó en [6]. En esta sección retomamos tal análisis siendo estos aspectos funcionales los siguientes:

- Cobertura de las Etapas del Ciclo de Vida de los Procesos: para el análisis de los componentes de las herramientas tendremos en cuenta las etapas de un ciclo de vida clásico cíclico, considerando las siguientes fases: Diseño y análisis -

Configuración (Definición) - Promulgación (Ejecución) - Evaluación (Monitorización) [7] [8].

- Mecanismos de Actualización del Modelo de Procesos de Negocio: sería útil que la herramienta tenga un mecanismo colaborativo durante el modelado por parte de varios usuarios para el cambio de algún proceso o flujo de trabajo. Si la herramienta utiliza una tecnología WEB 2.0 podría mejorar el trabajo colaborativo independientemente del lugar donde se encuentre el usuario [13], en esto hay que tener en cuenta algunos riesgos que debemos afrontar con respecto a la seguridad de los archivos compartidos y el acceso a los mismos [14].
- Mecanismos de Actualización de Instancias de Proceso: ante las distintas instancias que se pueden crear de la ejecución del proceso de negocio, del mismo pueden surgir diferentes subconjuntos de tareas y pueden tener diferentes caminos de ejecución, por lo que los cambios en el mismo pueden ocasionar inconsistencias en el mismo. En los cambios del esquema de flujos de trabajo, hay dos cuestiones a tener en cuenta: se deben analizar la nueva situación excepcional y ver si los cambios se deben propagar hacia abajo en la implementación del proceso de negocio. Los procesos pueden respetar una estructura determinada o variar con el tiempo, ya sea porque la definición de los mismos presenta cambios por naturaleza, o los requerimientos aún no están tan definidos completamente o simplemente porque modelan una realidad que presenta cambios continuos. Los cambios a realizarse en las instancias del proceso, por cambios en el modelo por ejemplo, se pueden realizar una vez terminada la ejecución del mismo, o bien durante la ejecución de las mismas. Hacer los cambios una vez que termine la ejecución de los casos puede ser un inconveniente cuando el tiempo de ejecución de las instancias es muy largo. En ese caso, es importante considerar cuál sería el mecanismo de actualización más adecuado. Los cambios en la ejecución del proceso, en las instancias del mismo, pueden ser causados por eventos no planificados, excepciones, o información errónea en la entrada a las tareas [10]. Ante los cambios en las instancias, la herramienta BPMS debe tener en cuanta: ¿Cómo se resuelven los cambios? ¿Qué estrategias de cambios siguen? [15]. Existen patrones de diseño que tienen en cuanta cambios en las instancias como se muestran en [16].
- Capacidad de Distribución de Procesos en Varios Motores: hoy en día es muy común la colaboración entre procesos de negocio como servicios, lo que se lleva a pensar en flujos de trabajo más dinámicos y flexibles, así como la distribución de los procesos de negocio. La necesidad de escalar y la naturaleza netamente colaborativa que subyace a la ejecución de procesos de negocio, ponen en juego la necesidad de contar con entornos descentralizados que permitan optimizar el uso de la tecnología y dar respuesta a la necesidad de crecimiento de las organizaciones. En la actualidad hay varios métodos para descomponer un proceso de negocio en múltiples procesos que se despliegan en las instalaciones propias y en la nube en función del rendimiento y requisitos de sensibilidad, expresándose los mismos como anotaciones sobre las actividades y los datos. Este modelo de distribución da cuenta de la factibilidad de contar con un modelo de distribución de procesos [17] [18] [19].

- Integración con Portales de Autenticación SSO (Single Sign On): Los sistemas SSO permiten a los usuarios utilizar los servicios de varios sitios web en diversas aplicaciones o servicios web sin identificación y contraseña si los usuarios se autentican en un sitio web. Por lo que SSO ofrece la reducción de costos y comodidad a los administradores de sitios web y a los usuarios [20].
- Cumplimiento del Estándar BPMN: BPMN ha sido desarrollado para proveer a los usuarios de una notación estándar, fue definido por Object Management Group (OMG). BPMN está dirigido a usuarios y proveedores de servicios que requieren comunicar los procesos de negocio de una forma estándar [8]. El estándar BPMN v.1.x se definió en el año 2010 luego tuvo mejoras con la versión v.2.0., algunas mejoras son [21]: 1- formaliza la semántica de la ejecución para todos los elementos de BPMN, 2- agrega nuevos eventos que no interrumpen (Non-interrupting Events) y eventos subprocesos entre procesos, 3- genera un meta modelo formal mostrado mediante diagramas de clases, 4- define el modelo de colaboración entre procesos y 5- define el modelo de coreografía.

3 Herramientas Analizadas

En este trabajo se evaluaron dos BPMS, muy populares y con muy buena documentación online disponible. La selección de las mismas se llevó a cabo siguiendo los siguientes puntos:

- *BonitaSoft*: es una herramienta muy popular, las actualizaciones de las mismas son constantes. Tiene una comunidad muy activa la misma se puede ver en sus foros, documentación muy completa y videos de tutoriales disponibles, tiene también un curso Online gratis.

- *Bizagi*: tiene una comunidad muy activa donde hay un foro por cada módulo de la misma, proporciona una documentación muy completa (proporciona una guía de usuario muy completa), tutoriales en videos, la actualización de versiones es muy activa agregando nuevas funcionalidades y realizando mejoras al producto.

En las cuales se evaluaron y analizaron las características funcionales avanzadas descriptas anteriormente. Existen muchas herramientas BPMS con distintos tipos de licencias Free, Open Source o Propietarias.
A continuación se describen las herramientas analizadas:

3.1 *BonitaSoft*

Es una herramienta BPMS la cual consta de cuatro ediciones [22]:

- Community: Es la edición Open Source, la cual permite entre las funcionalidades más importantes: modelado, conectores, generación de aplicaciones, manejos de los procesos, etc.
- Teamwork: esta edición está basado en los ambientes colaborativos, y entre las funcionalidades más importantes: un repositorio compartido, productividad avanzada para desarrolladores y personas de negocio, etc.

- Efficiency: esta edición presenta un ambiente avanzado, además de contar con las características de la edición anterior, también presenta plantillas de procesos y perfiles personalizados, entre otras características.
- Performance: esta edición es para un ambiente más crítico, monitorización, tareas de gestión, entre otras características.

La edición Community tiene licencia GNU GPL v2. Disponible para todas las plataformas de sistemas operativos. El lenguaje en el cual está desarrollada en Java. Esta herramienta consta de dos partes principales [23]:

Bonita BPM Studio
Es una ambiente gráfico para la creación de procesos, presenta las siguientes herramientas:

- El whiteboard en el cual se dibuja el diagrama del proceso de negocio,
- El constructor de formularios, el cual permite de manera WYSIWYG

Una vez listo el diseño del proceso, se construye e implementa en Bonita Platform.

Bonita BPM Platform
Dentro de este módulo encontramos lo siguiente; JEE Application Server, en el cual se encuentran la aplicación web del portal de bonita y el Bonita Engine; además de una base de datos.

En el portal el usuario puede ver su proceso en ejecución.

3.2 *Bizagi*

Es una herramienta BPMS, la cual consta de tres módulos [24]:

- *Bizagi Modeler*: permite modelar y documentar procesos de negocio.
- *Bizagi Studio*: permite convertir diagramas de Proceso en aplicaciones ejecutables (workflow).
- *Bizagi Engine*: Permite ejecutar y controlar los procesos de negocio ya desplegados en los módulos Modeler y Studio.

Bizagi Modeler y *Bizagi Studio* pueden utilizarse de manera gratuita. *Bizagi Studio* soporta un máximo de veinte Usuarios en ambientes de desarrollo y pruebas.

4 Tabla comparativa de las herramientas

Se analizaron las herramientas propuestas en la sección 3 en las cuales se realizó una cuadro comparativo (Tabla – 1) de las siguientes características: Cobertura de las Etapas del Ciclo de Vida de los Procesos, Mecanismos de Actualización del Modelo de Procesos de Negocio, Mecanismos de Actualización de Instancias de Proceso, Capacidad de Distribución de Procesos en Varios Motores, Integración con Portales

de Autenticación SSO (Single Sign On) y Cumplimiento del Estándar BPMN; descriptas en la sección 2.

Del análisis propuesto podemos decir que las dos herramientas nos permiten el seguimiento del proceso de negocio en todo su ciclo de vida, la sincronización con un sistema SSO y el modelado con utilizando el estandar BPMN v2.0. El mecanismo de actualización de modelos de procesos es resuelto por ambas herramientas mediante la gestión de un repositorio común con manejo de versiones. En el caso de *BonitaSoft* utiliza el SVN clásico mientras que *Bizagi* opta por una opción de repositorios en la nube. En la actualización de instancias en los procesos *BonitaSoft* tiene en cuenta la posibilidad de cambios en la instanciación de los procesos, mientras que *Bizagi* propone patrones de diseño para estos casos.

En la tabla siguiente se detalla cada una de las características antes mencionadas.

Tabla 1: Comparación de herramientas

Características funcionales avanzadas	BonitaSoft	Bizagi
Versión	v.7.0.1	*Módulo Modeler:* Versión 2.9*, Módulo Studio:* Versión 10.6.1, *Módulo Engine:* Versión 10.6
Licencia	GNU GPL v2	*Módulo Modeler:* FreeSoftware, *Módulo Studio:* FreeSoftware, solo permite un máximo de veinte (20) Usuarios en ambientes de desarrollo y pruebas, *Módulo Engine:* Propietaria
Etapas del ciclo de vida cubiertas	*Edición comunidad*: Análisis, Definición, Ejecución. *Ediciones de subcripción Teamwork, Efficiency y Performance*: Monitorización.	*Módulo Modeler*: Análisis y Definición, *Módulo Studio:* Ejecución, *Módulo Engine:* Monitorización
Mecanismos de actualización del modelo de procesos de negocio	*Ediciones de subcripción Teamwork, Efficiency y Performance:* permite la colaboración de un equipo de trabajo a través de la herramienta Apache Subversion (es un sistema de control de versiones centralizado Open Source) [25] [26]	*Módulo Modeler:* colaboración del modelo en repositorio de archivos o en la nube utilizando un servicio de almacenamiento online, también se puede compartir un modelo en un Servidor de Archivos local, o en un almacenamiento online como ser Dropbox, Sugarsync o Sky drive. También tienen la posibilidad de una conversación/discusión on-line en el panel "Comentarios", esta funcionalidad es similar a Twiter [32].
Mecanismos de actualización de instancias de proceso	*Edición comunidad:* En la instanciación de un proceso si existe una tarea la cual falla y no se quiere ejecutar, frenar su ejecución y seguir con la ejecución del proceso [27]. *Ediciones de subcripción de Efficiency y Performance*: realizar una actualización durante la ejecución del proceso (durante su instanciación), estos cambios solo se pueden realizar en la nueva versión BonitaSoft 7.0.1[28][29]	*Módulo Modeler:* esta herramienta presenta en su documentación patrones de diseño [16] ante cambios en las instancias de procesos, pero no tienen contemplado el tema de los cambios en la ejecución y el impacto que puede ocasionar estos cambios.
Capacidad de distribución de procesos en varios motores	*En todas las ediciones:* a través de las extensiones API REST, BonitaSoft permite que sus recursos puedan ser exportados, importados, modificados y eliminados en el Bonita BPM Portal [30].	*Módulo Modeler:* Exposición de Servicios Web Hasta este momento se ha considerado la integración de *Bizagi* hacia otros sistemas, sin embargo es posible que otros sistemas desearan integrarse con el BPM. Para esto, Bizagi cuenta con su Capa de Integración Orientada a Servicios. En forma automática, cada proceso modelado en Bizagi está disponible para ser invocado mediante Servicios Web cuyo input y output son documentos XML. Las aplicaciones que invocan la capa de servicios Bizagi pueden llevar a cabo tareas como la creación de nuevas instancias de procesos, ejecutar actividades del proceso, ejecutar eventos, consultar o adicionar información sobre el proceso. Desde el punto de vista de arquitectura de sistemas, Bizagi es compatible con arquitecturas SOA, en la que las aplicaciones exponen sus principales funcionalidades como servicio para lograr integraciones entre sistemas que corren sobre plataformas heterogéneas. [33][34]
Integración con porta les de autenticación SSO (Single Sign On)	*Edición Comunidad:* solo está disponible el conector LDAP. *Ediciones de subscripción Teamwork, Efficiency y Performance:* está disponible la posibilidad de la sincronización de los datos de usuarios. [31]	*Módulo Studio:* si ya existe un servidor de LDAP configurado en la entidad donde se ha instalado Bizagi, este se sincroniza con este servicio. [35]
Cumple con Estándar BPMN	*En todas las ediciones:* A partir de la versión 6.x de BonitaSoft cumple con el estándar BPMN v.2.0.	*Módulo Modeler*: esta herramienta respeta BPMN v2.0 [36]

.

6 Conclusión

La utilización de un BPMS para el modelado, despliegue y monitorización de un proceso presenta muchas ventajas dado que dependiendo de la herramienta se puede utilizar como un trabajo colaborativo, facilita la creación del modelo y su despliegue y la correcta utilización de los elementos que representan cada una de las partes del modelo, como ser actividades, flujos, conectores, siguiendo el estándar preestablecido en la herramienta utilizada.

El objetivo de este trabajo es analizar y evaluar características funcionales avanzadas que hemos analizado en trabajos anteriores [6], en dos herramientas seleccionadas: *BonitaSoft* y *Bizagi*.

Muchas de las características analizadas en *BonitaSoft* se encuentran en las ediciones de subscripción, pero la versión comunidad brinda la oportunidad de extender los componentes, conectores o funcionalidad de la misma, teniendo un fuerte apoyo de la comunidad. Algunas de estas características son: la utilización de una herramienta de trabajo colaborativo como Apache Subversion, poder actualizar las instancias de los procesos en ejecución y sincronización con un servidor LDAP. En el caso de *Bizagi* algunas de estas características están disponibles en el módulo "Modeler" y otras en el módulo "Studio" al ser libre no permite agregar funcionalidad que se crea necesario como lo permite *BonitaSoft*. *Bizagi* presenta una ventaja muy importante en la actualización de modelo del proceso, dado que permite compartirlo en la nube, lo cual es una característica que hoy en día se está utilizando cada vez más en las empresas, brindando reducción de costos y mejoras en el procesamiento.

Este trabajo permite brindar un análisis y descripción de las herramientas propuestas, a partir del cual poder tener una visión distinta de las mismas lo cual puede ayudar para la selección para su uso de alguna de ellas, aportando un cuadro comparativo.

Referencias

1. Evert Duipmans, "Business Process Management inthe cloud: Business Process as aService (BPaaS)", University of Twente (Abril 2012)
2. Fuhua GH, Shaowen Yao. "Architecture Combining SOA and BPM". Information Management, Innovation Management and Industrial Engineering, 2008. ICIII '08. International Conference on (Volume: 1) (Diciembre 2008)
3. https://bpmnmatrix.github.io/
4. BonitaSoft.http://www.bonitasoft.com/
5. Bizagi.http://www.bizagi.com/
6. Anahí Soledad Rodríguez, Patricia Bazán y Javier Díaz, "BPMS - Análisis de aspectos funcionales avanzados: una clasificación propuesta", LINTI – Facultad de Informática - UNLP (Abril 2015)
7. Mathias Weske. "Business Process Management. Concepts, Languages, Architectures" ISBN 978-3-540-73521-2 Springer Berlin Heidelberg New York
8. P.Bazán. "Un modelo de integrabilidad con SOA y BPM" Tesis de maestría. (Diciembre 2009)
9. TIAN Guo-shuang, QUAN Liang. "An Improved Framework of Business Process Management System. Which Integrating the Strategy Management". Management Science and Engineering, 2008. ICMSE 2008. 15th Annual Conference Proceedings, International Conference on. (Septiembre 2008)
10. Ru Fang, Zhi Le Zou, Corina Stratan, Liana Fong, David Marston, Linh Lam, David Frank. "Dynamic Support for BPEL Process Instance Adaptation". Services Computing, 2008. SCC '08. IEEE International Conference on (Volume: 1). (Julio 2008)
11. Jan-Philipp Friedenstab, Christian Janiesch, Martin Matzner, Oliver Muller. "Extending BPMN for Business Activity Monitoring"Proceedings of the 2012 45th Hawaii International Conference on System Sciences. (2012)
12. David W. McCoy. "Business Activity Monitoring: Calm Before the Storm", (Abril 2012).
13. August-Wilhelm Scheer, Joerg Klueckmann. "BPM 3.0". 7th International Conference, BPM 2009, Ulm, Germany. (Septiembre 2009)
14. Documentación Bizagi(Julio 2015). "Bizagi BPM Suite Descripción Funcional" - https://www.bizagi.com/docs/BizAgi%20Descripcion%20Funcional.pdf.
15. "Process-oriented System Automation - Executable Process Modeling & Process Automation" http://fundamentals-of-bpm.org/wp-content/uploads/2013/02/ProSA-7-Automation.pptx (Febrero 2013)
16. Documentación Bizagi."Modeling Workflow Patterns Bizagi Suite" (2014)
17. J. Martinez Garro, P. Bazán, "Monitoreo de procesos distribuidos en el cloud - Una propuesta arquitectónica". (2013)
18. T. Anstett, F. Leymann, R. Mietzner, and S. Strauch, "Towards bpel in the cloud: Exploiting different delivery models for the execution of business processes" in Proceedings of the 2009 Congress on Services - I. Washington, DC, USA: IEEE Computer Society, 2009, pp. 670–677
19. J.Martinez Garro, P.Bazán "Constructing hybrid architectures and dynamic services in Cloud BPM" Science and Information Conference 2013 October 7-9, 2013 | London, UK

20. Dae-Hee Seo, Im-Yeong Lee, Soo-Young Chae, Choon-Soo Kim. “Single sign-on authentication model using MAS (multiagent system)”. Communications, Computers and signal Processing, 2003. PACRIM. 2003 IEEE Pacific Rim Conference on (Volume: 2) (Agosto 2013)
21. “Business Process Model and Notation (BPMN) Version 2.0”. (Enero 2011).
22. Bonitasoft Team, Francisco Kem Iti Saito,“BonitaSoft Corporate and Product Overview”.(Abril 2012).
23. Documentación BonitaSoft(Julio 2015). http://documentation.bonitasoft.com/bonita-bpm-overview
24. Documentación Bizagi(Julio 2015). http://help.bizagi.com/bpmsuite/es/
25. Documentación BonitaSoft (Julio2015)
http://www.bonitasoft.com/products/feature/team-collaboration-bpm-repository
26. Documentación BonitaSoft (Julio2015)
http://documentation.bonitasoft.com/5x/create-shared-repository-1
27. Documentación BonitaSoft (Julio 2015). http://documentation.bonitasoft.com/tasks-2#skip
28. Documentación BonitaSoft (Julio 2015). http://documentation.bonitasoft.com/new-features
29. Documentación BonitaSoft (Julio 2015). http://documentation.bonitasoft.com/live-update
30. Documentación BonitaSoft (Julio 2015). http://documentation.bonitasoft.com/rest-api-extensions
31. Documentación BonitaSoft (Julio 2015). http://documentation.bonitasoft.com/user-authentication-3
32. Documentación Bizagi (Julio 2015). http://help.bizagi.com/processmodeler/es/index.html?where_to_share.htm
33. Documentación Bizagi (Julio 2015).https://www.bizagi.com/docs/Standard%20Descripci%C3%B3n%20Funcional.pdf
34. Documentación Bizagi (Julio 2015).http://help.bizagi.com/bpmsuite/es/index.html?servicios_web_-_capa_soa.htm
35. Documentación Bizagi (Julio 2015).http://help.bizagi.com/bpmsuite/es/index.html?ldap.htm
36. Documentación Bizagi (Julio 2015). https://www.bizagi.com/docs/BPMNbyExampleSPA.pdf

Mejoras a un Modelo Léxico mediante Mapas Conceptuales

Alberto Sebastián[1], Graciela D.S. Hadad[2,3]

[1]Escuela de Posgrado, Universidad Católica Argentina
[2]Escuela de Informática, Universidad Nacional del Oeste
[3]Facultad de Ingeniería y Tecnología Informática, Universidad de Belgrano
asebastian@outlook.com.ar, ghadad@uno.edu.ar

Abstract. Para aumentar la probabilidad de éxito de un proyecto de software es fundamental partir de buenas especificaciones de requisitos, pues defectos en ellas se trasladan a subsiguientes modelos y componentes de software. Es por ello que deben realizarse verificaciones tempranas de los modelos que se construyen en el proceso de requisitos. La mayoría de las técnicas de verificación apuntan a detectar defectos del tipo hechos incorrectos, inconsistencias y en menor medida omisiones. Frecuentemente, los modelos en la Ingeniería de Requisitos suelen escribirse en lenguaje natural, lo que propende también a la aparición de ambigüedades. Se presenta una propuesta preliminar de verificación de un modelo léxico utilizando mapas conceptuales, centrada en detectar principalmente omisiones y ambigüedades, y en formular correcciones a las mismas. Esta propuesta fue aplicada a un caso real, donde se comprobó la detección de un número razonable de defectos.

Keywords: Ingeniería de Requisitos, Verificación de Requisitos, Omisiones, Mapas Conceptuales, Léxico Extendido del Lenguaje.

1 Introducción

La falta de completitud es uno de los problemas de mayor dificultad de detección, que atenta contra la calidad de los modelos que se construyen en la Ingeniería de Requisitos [1, 2]. Es sabido que la completitud es una propiedad imposible de lograr en problemas complejos, por ende, se establece como meta lograr un nivel aceptable de completitud [3]. En el caso de modelos escritos en lenguaje natural, aparece otro tipo de defecto: ambigüedad, que conlleva a más de una interpretación [4].

Dado que las especificaciones de requisitos son la base para la construcción del software, entonces es mandatorio realizar actividades que comprueben la calidad de las mismas en procesos de software bien establecidos. La verificación es una de ellas, la que dispone de diversas técnicas dependiendo del tipo de modelo a verificar. Se suelen utilizar revisiones para verificar modelos escritos en lenguaje natural [1, 5, 6].

El objetivo de este artículo es presentar una propuesta preliminar de verificación de un modelo léxico construido en un proceso de Ingeniería de Requisitos [7], que identifica mayoritariamente omisiones y ambigüedades. Este proceso de verificación incluye corrección de defectos y utiliza un artefacto intermedio: mapas conceptuales

[8], que se construyen a partir del modelo léxico. Estos mapas conceptuales permiten estudiar conceptos y su semántica, estableciendo relaciones entre conceptos, por lo que facilitan identificar omisiones y ambigüedades, dado que esta falta de completitud impide la comprensión del mapa. Esta propuesta preliminar fue aplicada a un caso real, en el cual se detectaron distintos tipos de defectos, indicándose pasos para su resolución, lo que mejoró la comprensión del modelo por parte de los clientes.

En la sección siguiente se exponen técnicas de verificación que pueden aplicarse a modelos en lenguaje natural y se presenta el modelo léxico, objeto de verificación de esta propuesta. En la sección 3 se presentan las características de los mapas conceptuales, mientras que en la sección 4 se detalla la propuesta de verificación con corrección de defectos en el modelo léxico. Finalmente, se exponen las conclusiones sobre el trabajo presentado y futuros pasos para extender este proceso de verificación.

2 Verificación de Requisitos en Modelos en Lenguaje Natural

En la Ingeniería de Requisitos, está muy difundido el uso de modelos en lenguaje natural, como casos de uso y escenarios [7], dado que son de fácil comprensión por los clientes. Cuando se acompañan de glosarios, se mejora aún más dicha comprensión. Un glosario específico, denominado Léxico Extendido del Lenguaje (LEL) [9], representa el vocabulario utilizado en un contexto particular. Este modelo se utiliza en un proceso de requisitos orientado al cliente [7], el que incentiva el uso de términos definidos en el LEL en las descripciones de otros modelos. Cada término o símbolo se describe con un nombre (o más de uno para sinónimos), una noción (denotación del símbolo) y un impacto (connotación del símbolo). La noción y el impacto se describen mediante una o más oraciones, las que deben cumplir con dos principios [9]: i) principio de circularidad (maximizar el uso de símbolos en la definición de otros símbolos), y ii) principio de vocabulario mínimo (minimizar el uso de términos externos al LEL en la definición de símbolos).

El primer principio implica que cada símbolo mencione al menos a otro símbolo. Esta mención implica la incorporación de un hipervínculo a la definición del símbolo. El principio de vocabulario mínimo requiere la existencia de una lista de términos que permitan describir ideas generales en cualquier dominio. Los símbolos del LEL se clasifican en cuatro tipos: sujeto, objeto, verbo o estado, para permitir homogeneizar sus descripciones según el tipo. La Fig. 1 muestra dos símbolos del LEL construidos en el contexto de una institución de salud para la gestión de auditoría, análisis y consulta de la facturación emitida por la misma a los respectivos prestadores de salud. Este modelo fue construido para uso interno de la institución, con el fin de formalizar la terminología común utilizada por médicos, prestadores y el equipo de desarrollo.

Estudios de completitud, basados en métodos estadísticos y correcciones semánticas, realizados sobre 9 muestras de un mismo modelo léxico [10] establecieron que el nivel de completitud en el mejor casos era del 59%, valor extremadamente bajo. Es por ello que se considera importante poder aportar un proceso de verificación que permita mejorar la identificación de omisiones.

La técnica de revisiones está ampliamente difundida para verificar modelos escritos en lenguaje natural [1, 5]. Las revisiones presentan variantes para detectar

defectos: lectura ad-hoc del modelo, lectura usando checklist, lectura usando procedimientos que guían la detección, o la construcción de artefactos intermedios a partir de los cuales se realiza la detección de defectos [6]. Esta última variante no es tan frecuentemente utilizada, debido principalmente a la necesidad de construir una nueva representación a partir del modelo a verificar [11]. Sin embargo, puede promover la detección de más defectos por la tarea misma de construcción. La propuesta de verificación que se presenta en este artículo se basa en esta variante.

Símbolo del LEL **(a)** – Tipo Verbo	Símbolo del LEL **(b)** – Tipo Objeto
GESTIONAR LOS FILTROS APLICADOS **Noción:** • Proceso por el cual el Prestador ICBA puede Exportar Los Filtros Aplicados o Importar Los Filtros Aplicados • Lo realiza cuando analiza o consulta la facturación mensual emitida **Impacto:** • El Prestador ICBA almacena los Filtros Aplicados para su posterior uso o transmisión	**NAVEGADOR OS** **Noción:** • Aplicativo del ICBA que facilita a los Prestadores ICBA, la gestión de la Facturación Mensual Emitida ICBA **Impacto:** • Permite Gestionar los Filtros Aplicados

Fig. 1. Ejemplos de símbolos de LEL. Los términos subrayados son hipervínculos a la definición de dicho símbolo. Estos símbolos son las versiones originales sin corrección.

Existe en la literatura una propuesta de verificación del modelo LEL basada en la técnica de inspección mediante procedimientos que involucra completar formularios [1]. Las omisiones que detecta son por simple aparición de noción o impacto vacío, o por repetición de una misma frase en varios símbolos sin identificarse como símbolo, requiriendo una comparación de todas las oraciones de la noción e impacto por pares de símbolos, todos contra todos. Las ambigüedades son tratadas como omisiones.

3 Uso de Mapas Conceptuales

En esencia, las diferencias que surgen en los requisitos, se basan en lo que los ingenieros piensan o entienden que se tiene que construir, y lo que los interesados piensan o entienden que van a obtener. Existen diversas alternativas para afrontar el problema de la ambigüedad, pero la más significativa consiste en aprender a detectar la misma y sus imprecisiones en los documentos que generan los ingenieros de requisitos.

Desde la lingüística castellana, como disciplina que estudia el significado de las palabras, se define a la ambigüedad como la capacidad de entender de dos formas diferentes una palabra o frase, así como también, no ser entendida en su totalidad o ser incierta [12]. En el marco de la Ingeniería de Requisitos, desde la lingüística se entiende la ambigüedad como léxica, sintáctica, semántica y gramatical [4]. No obstante, en todo texto no se presenta en forma aislada, sino combinada. Para desambiguar un texto, se han propuesto métodos, como cambio de las construcciones, acentuación y puntuación, reescritura e incorporación de palabras [13].

La construcción de significado e interpretación de una frase no es exclusiva del

dominio de la lingüística, sino también de varias disciplinas. Entre ellas se encuentra la Gestión de Conocimiento, cuyo fin es la transmisión de conocimiento desde el lugar donde se genera hasta el lugar donde se va a aplicar en el marco de una organización. A las organizaciones les interesa lo que saben los individuos de las mismas, pues dichos conocimientos son transmisibles [14]. Existen diferentes estrategias de Gestión de Conocimiento, siendo una de las más utilizadas la construcción de mapas conceptuales, los que son un recurso esquemático para representar un conjunto de significados conceptuales incluidos en una estructura de proposiciones [8]. La técnica de construcción de mapas conceptuales [15] es un proceso simple que responde a: i) establecer un dominio de conocimiento restringido para el mapa conceptual; ii) escribir los conceptos identificados, considerando que dentro del mapa no se pueden repetir conceptos; iii) ordenar dichos conceptos de los más generales a los más específicos, siendo que existe un concepto raíz que se quiere estudiar con un fin determinado; iv) establecer las relaciones entre conceptos, mediante palabras o frases de enlace, las que pueden ser verbos, artículos, preposiciones y/o conjunciones; v) pueden existir relaciones entre conceptos de diferentes segmentos o dominios del mapa conceptual, denominados enlaces cruzados; y vi) revisar el mapa conceptual para respetar la claridad y simplicidad del mismo. En cuanto a la representación del mapa conceptual, los conceptos se colocan dentro de óvalos; las relaciones entre conceptos se representan mediante flechas que los unen, junto con frases de enlace; y se pueden utilizar diferentes colores para conceptos y relaciones para facilitar el impacto visual.

4 Verificación del LEL con mapas conceptuales

El proceso propuesto de mejora del modelo LEL consta de dos etapas: i) construcción de mapas conceptuales de símbolos del LEL, y ii) detección y corrección de defectos. La primera etapa básicamente consiste en generar mapas conceptuales de cada símbolo del LEL a partir de la información contenida en el símbolo y su relación con otros símbolos. La segunda etapa se focaliza en detectar y corregir omisiones y ambigüedades, a partir del análisis de los mapas conceptuales construidos.

4.1 Etapa de Construcción de Mapas Conceptuales de Símbolos del LEL

Se presenta una adaptación de la técnica de construcción de mapas conceptuales para utilizarlos como medio de verificación. Considerando la recomendación de generar mapas conceptuales en un dominio acotado [15], es que se construye un mapa conceptual por cada símbolo del LEL. El concepto raíz de cada mapa conceptual representa el símbolo del LEL bajo estudio (CRL: Concepto Raíz símbolo del LEL). Cada oración en la noción y en el impacto de este símbolo en el LEL se considera una proposición y se la transforma en conceptos y frases de enlace para armar el mapa conceptual, donde los conceptos pueden ser de dos categorías: los que representan otros símbolos del LEL (CSL: Concepto Secundario símbolo del LEL) y los que no representan símbolos del LEL (CSnoL: Concepto Secundario no símbolo del LEL).

El mapa se divide en dos partes: espacio de Mapa, donde se representa el propio mapa conceptual del CRL, y un espacio de Mención, donde se representan los conceptos secundarios símbolos del LEL (CSL) que mencionan al símbolo raíz CRL en sus respectivos mapas conceptuales. Cada concepto símbolo del LEL, sea CRL o CSL, se dibuja con un óvalo con su nombre y su tipo (s: sujeto, o: objeto, v: verbo, e: estado). Cada concepto se dibuja en un óvalo, pintado con un color distinto dependiendo del tipo: CRL, CSL o CSnoL. Las relaciones entre conceptos provenientes de la noción del CRL se representan con flechas continuas y las provenientes del impacto con flechas punteadas, indicando las frases de enlace extraídas de la descripción del símbolo. La Fig. 2 representa el mapa conceptual de un símbolo del LEL del caso de la institución de salud, donde se distinguen los espacios de Mención y de Mapa. La definición del símbolo en el LEL se presenta en Fig.1(a).

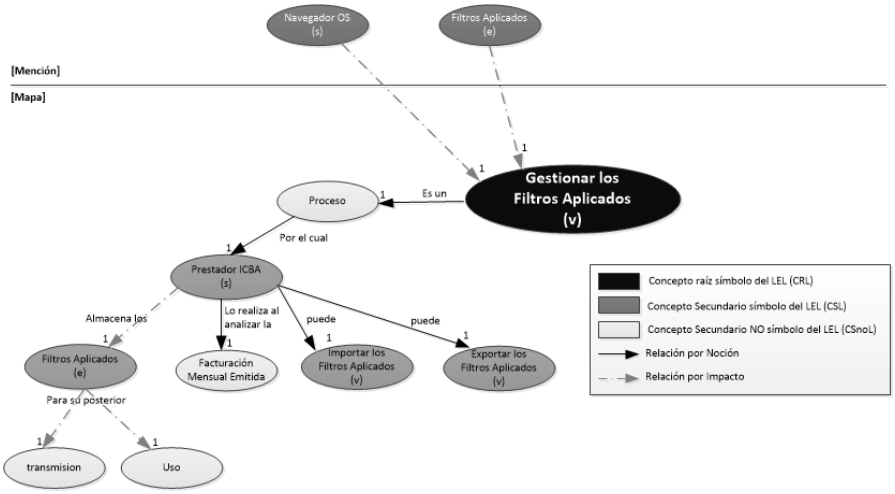


Fig. 2. Ejemplo de mapa conceptual de un símbolo del LEL

4.2 Etapa de Detección y Corrección de Defectos

El propósito de esta etapa consiste en detectar y corregir defectos del modelo LEL, a través del análisis de los mapas conceptuales de cada símbolo. Se enfoca en defectos del tipo: omisiones, ambigüedades y errores. Consiste en un análisis de conceptos que no son símbolos del LEL y un análisis de relaciones de cada mapa y entre mapas.

Análisis de Conceptos no Símbolos del LEL. Se analizan CSnoL de todos los mapas conceptuales construidos. Cada concepto CSnoL pertenece al vocabulario mínimo. Se realizan los siguientes pasos:

1) Construir una lista ordenada alfabéticamente con todos los conceptos CSnoL de cada mapa conceptual, indicando el símbolo CRL del mapa donde aparece. Los CSnoL pueden estar repetidos por aparecer en más de un mapa.
2) Establecer si hay sinónimos entre los CSnoL de la lista (ver Tabla 1). Se mejora

la aplicación del principio de vocabulario mínimo, reduciendo la ambigüedad en el LEL. Tipo de defecto: Ambigüedad. *Corrección*:

a) Unificar el nombre del concepto CSnoL y reemplazar este nombre en todos los mapas conceptuales donde se menciona el CSnoL.
b) Reemplazar este nombre en toda aparición en el modelo LEL.
c) Reducir el vocabulario mínimo dejando solo el nombre unificado.

Tabla 1. Ejemplo de lista parcial de símbolos CSnoL, donde el concepto *Acción* es reemplazado por su sinónimo *Proceso*, que es otro CSnoL.

CSnoL	CRL donde aparece el CSnoL	Sinónimo de CSnoL detectado
Acción	Exportar los Filtros Aplicados	Proceso
Acción	Importar los Filtros Aplicados	Proceso
Proceso	Visualizar la Facturación Mensual Emitida	
Proceso	Filtrar la Facturación Mensual Emitida	

3) Establecer conceptos CSnoL que son sinónimos de conceptos símbolos del LEL (CRL). Esto permite reducir la ambigüedad en las definiciones del LEL, al mejorar la aplicación de los principios de circularidad y de vocabulario mínimo. Puede también detectarse una omisión de sinónimo de símbolo del LEL.
Tipo de defecto: Ambigüedad y posible Omisión. *Corrección:*
   a) Reemplazar cada CSnoL por el nombre del correspondiente símbolo del LEL en todos los mapas conceptuales, identificándolo como CSL.
   b) Reemplazar en el LEL cada aparición del término CSnoL por su correspondiente nombre del símbolo del LEL, incluyendo el hipervínculo a la definición del símbolo.
   c) Eliminar el CSnoL del vocabulario mínimo.
   d) Validar si el nombre del CSnoL es también usado en el contexto. En tal caso, agregar el nombre como sinónimo en la definición del símbolo del LEL.

4) Calcular para cada CSnoL, que ha permanecido como tal en la lista, la frecuencia total de aparición en todos los mapas conceptuales.

5) Realizar un análisis de Pareto sobre los CSnoL, donde se considera que el 20% de los CSnoL que tengan la mayor frecuencia formarán una lista de conceptos candidatos a transformarse en símbolos del LEL (ver Tabla 2).

Tabla 2. Ejemplo de análisis de Pareto sobre CSnoL

CSnoL	Frecuencia	% Frecuencia	Frecuencia Acumulada	
Proceso	8	14%	14%	} Candidatos a símbolo
Medio	4	7%	21%	
Archivo	3	5%	26%	
Paciente	3	5%	32%	

6) Analizar el significado de cada CSnoL candidato a símbolo para establecer si aporta nueva semántica propia del contexto, para incluirlo como símbolo del LEL. Esto permite la detección de omisiones de símbolos. Tipo de defecto:

Omisión. *Corrección;*

a) Validar si el CSnoL se usa en el contexto. En tal caso, incluir el concepto como un nuevo símbolo con su definición en el LEL.
b) Indicar cada aparición del concepto en otros símbolos del LEL con un hipervínculo a su definición.
c) Construir su mapa conceptual para que luego se analicen sus relaciones.
d) Marcar el concepto como CSL en todos los mapas conceptuales restantes.

Análisis de Relaciones en Mapas Conceptuales. El análisis de las relaciones de cada mapa permite establecer problemas en la semántica proposicional de la noción y del impacto de cada símbolo del LEL. Se realizan los siguientes pasos:

1) Detectar sub-grafos inconexos en un mapa conceptual. Se identifican relaciones omitidas en la noción o impacto del símbolo. Su corrección permite una mejor comprensión del símbolo (ver Fig. 3). Tipo de defecto: Omisión. *Corrección:*

a) Establecer una relación entre el símbolo CRL y el concepto desconectado, ya sea CSL o CSnoL, y reescribir las relaciones derivadas del CSL o CSnoL.
b) Reescribir la oración en la noción o impacto del símbolo CRL en el LEL.

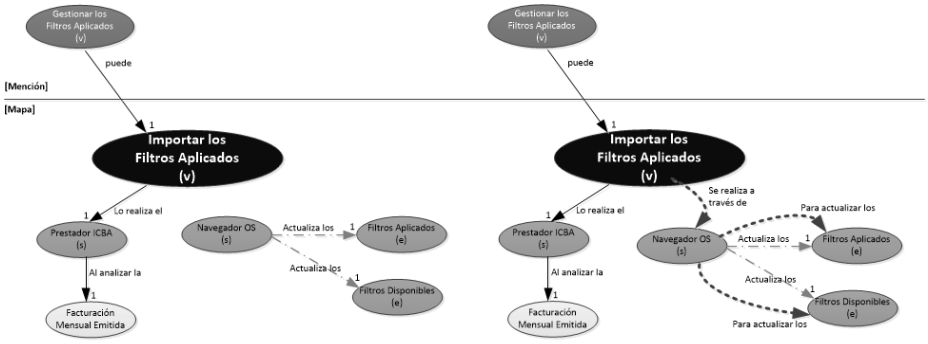


Fig. 3. Ejemplo de sub-grafo inconexo. A la izquierda, se observa la desconexión en el mapa conceptual. A la derecha, se presenta la corrección con flechas curvas punteadas descendientes.

2) Detectar ausencia de relaciones de noción y de impacto del CRL con ningún CSL en el espacio de Mapa. Esto permite detectar símbolos del LEL sin referencias a otros símbolos en su definición, vulnerando el principio de circularidad. Incorporando una relación omitida se logra mejorar la comprensión de la definición del símbolo del LEL. Eventualmente puede establecerse que el símbolo no es relevante y se eliminaría.

a) Buscar conceptos CSnoL del mapa conceptual del CRL que estén usados en otros mapas conceptuales, y analizar si dichos mapas conceptuales tienen relación semántica con el CRL bajo estudio.

2.1) Caso en que se puede establecer una relación semántica entre ambos mapas conceptuales a través del CSnoL. Tipo de defecto: Omisión. *Corrección*:

b) Incorporar el símbolo CSL o CRL detectado en otro mapa conceptual como un CSL en el mapa conceptual bajo estudio en el espacio Mapa. Dos casos:
c) Generar una relación indirecta en el mapa conceptual del CRL bajo estudio,

desde el CSnoL al nuevo CSL incorporado; o

d) Copiar la relación existente del CSnoL con el CSL del otro mapa conceptual, al mapa conceptual del CRL bajo estudio.
e) Actualizar la oración en la noción o impacto del símbolo CRL en el LEL, incorporando el hipervínculo al símbolo CSL identificado.

2.2) Caso en que no se puede establecer una relación semántica con ningún mapa conceptual, analizar la posibilidad de que falte información para establecer la relación o que el CRL no sea relevante en ese contexto. Tipo de defecto: Omisión o Error. *Corrección*:

f) Elicitar en el contexto información faltante respecto al símbolo CRL.
g) Si se obtiene una relación con otro símbolo, incluir en el mapa conceptual del CRL un CSL existente y su relación con el CRL, e incorporar la relación en la descripción del símbolo CRL del LEL como una nueva oración o parte de una oración en la noción o impacto (ver Fig. 4); o
h) Si se detecta la no relevancia del símbolo, eliminar el mapa conceptual del CRL, en los otros mapas conceptuales donde figura como CSL marcarlo como CSnoL, eliminar el símbolo del modelo LEL, eliminar las referencias desde otros símbolos al CRL e incorporarlo al vocabulario mínimo.

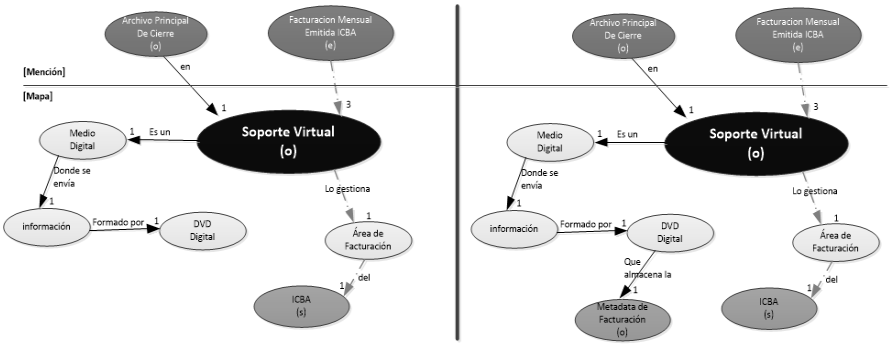


Fig. 4. Ejemplo de ausencia de relación entre CRL y algún CSL. En el mapa de la izquierda, el símbolo CRL *Soporte Virtual* no tiene un concepto CSL asociado en la noción. El mapa de la derecha representa la incorporación de una relación al concepto CSL *Metadata de Facturación*.

3) Detectar símbolos CRL que tienen el espacio de Mención vacío. Esto significa que el símbolo no es referenciado por ningún otro símbolo en el modelo LEL, vulnerando el principio de circularidad.

a) Buscar si existen CSnoL del mapa conceptual del CRL usados en otros mapas conceptuales y si dichos mapas tienen relación semántica con el CRL.

3.1) Caso en que se puede establecer una relación semántica. Tipo de defecto: Omisión. *Corrección:*

b) Generar la relación en el otro mapa conceptual con el CSnoL encontrado y el CRL bajo estudio marcándolo como CSL. Incorporar la relación en el espacio Mención del mapa del CRL.
c) Incluir la relación como una oración o parte de oración en la noción o impacto del símbolo del otro mapa conceptual en el LEL, colocando el hipervínculo a la definición del símbolo CRL.

3.2) Caso en que no se puede establecer relación semántica desde ningún otro mapa conceptual, analizar la posibilidad de que falte información para establecer la relación o eventualmente que falte un símbolo que lo vincule con el resto de los símbolos. Tipo de defecto: Omisión. *Corrección:*

d) Elicitar en el contexto información faltante respecto al símbolo CRL.
e) Si se detecta una nueva relación semántica con el CRL en algún mapa existente, incluir la relación en dicho mapa. Incorporar el símbolo CRL de dicho mapa en el espacio Mención del mapa del CRL. Incluir la relación en una oración o parte de oración de noción o impacto del símbolo correspondiente al mapa donde se incorporó la relación; o
f) Si se detecta la existencia de un nuevo símbolo del LEL, construir su mapa incluyendo una relación al símbolo CRL bajo estudio. Incorporar el nuevo símbolo como concepto CSL en el espacio de Mención del CRL bajo estudio. Incorporar en el LEL la descripción del nuevo símbolo con un hipervínculo al CRL bajo estudio.

4.3 Resultados de la Verificación

El modelo léxico construido en el contexto de un instituto de salud contiene 22 símbolos, 28 oraciones en las nociones y 34 oraciones en los impactos entre todos los símbolos. En el conjunto de mapas conceptuales iniciales se presentaron 49 conceptos no símbolos del LEL que se redujeron a 37, pues se detectaron que 14 conceptos eran sinónimos entre sí, quedando solo 8 de ellos. Además, 4 conceptos eran sinónimos de 3 símbolos del LEL. La reducción de conceptos que eran sinónimos entre sí provocó una mejora en el uso del vocabulario mínimo, reduciendo el nivel de ambigüedad propio de modelos escritos en lenguaje natural. Los conceptos detectados como sinónimos de símbolos del LEL fueron reemplazados por dichos símbolos, mejorando la comprensibilidad del texto por parte de los clientes. En resumen, el proceso de verificación presentado y aplicado sobre este modelo léxico detectó un total de 34 defectos, compuesto por: 15 omisiones, 18 ambigüedades y 1 error (ver Tabla 3).

Tabla 3. Defectos detectados al verificar el modelo léxico utilizando mapas conceptuales

Análisis de Conceptos No Símbolo del LEL			**Análisis de Relaciones en Mapas Conceptuales**		
Paso	**Tipo de Defecto**	**Cantidad de Defectos**	**Paso**	**Tipo de Defecto**	**Cantidad de Defectos**
2	Ambigüedad	14	1	Omisión	3
3	Ambigüedad	3	2.1	Omisión	2
3	Omisión	1	2.2	Omisión	1
6	Ambigüedad	1	2.2	Error	1
6	Omisión	7	3.1	Omisión	0
			3.2	Omisión	1

5 Conclusiones

Se ha presentado un proceso de verificación del modelo Léxico Extendido del Lenguaje, dado que es de vital importancia asegurar su calidad, por ser el primer modelo que se construye en un proceso de Ingeniería de Requisitos, desde el cual

puede extraerse información para derivar otros modelos [7]. Por otro lado, estudios de completitud sobre este modelo han detectado un alto grado de omisiones [3, 10].

A diferencia de la técnica de inspección del modelo LEL basado en formularios presentado en [1], el proceso de verificación basado en mapas conceptuales se concentra en la detección de omisiones y ambigüedades. A través de este proceso se han podido detectar principalmente: omisiones de símbolos, de nociones y de impactos, y ambigüedades por no cumplimiento de principios de circularidad y de vocabulario mínimo. También se detectan errores por símbolos no relevantes.

En próximos pasos, se realizará un estudio semántico más profundo, incluyendo nuevos pasos y la generación de árboles semánticos complementarios a los mapas conceptuales, para la detección de otros tipos de defectos.

Referencias

1. Kaplan, G., Hadad, G., Doorn, J., Leite, J.: Inspección del Léxico Extendido del Lenguaje. En: 3rd Workshop on Requirements Engineering, pp.70--91. Río de Janeiro (2000)
2. Firesmith, D.: Are Your Requirements Complete? Journal of Object Technology, vol.4, nº1, pp.27--43 (2005)
3. Hadad, G., Litvak, C., Doorn, J., Ridao M.: Dealing with Completeness in Requirements Engineering. En: Mehdi Khosrow-Pour (ed), Encyclopedia of Information Science and Technology, 3rd ed., pp.2854--2863. IGI Global, Information Science Reference (2014)
4. Berry, D.M., Kamsties, E.: Ambiguity in Requirements Specification. En: J. Leite, J. Doorn (eds.) Perspectives on Software Requirements, pp.7--44. Kluwer Academic Publishers. Springer US (2004)
5. Porter, A.A., Votta Jr., L.G.: Comparing Detection Methods for Software Requirements Inspections: A Replication Using Professional Subjects. Empirical Software Engineering, vol.3, nº4, pp.355--380 (1998)
6. Regnell, B., Runesom, P., Thelin, T.: Are the perspectives really different? Further experimentation on scenario-based reading of requirements. Requirements engineering with use cases – a basis for software development. Technical Report 132, Lund University, pp.141--180 (1999)
7. Leite, J.C.S.P, Doorn, J.H., Kaplan, G.N., Hadad, G.D.S., Ridao, M.N.: Defining System Context using Scenarios. En: J.C.S.P. Leite, J.H. Doorn (eds.) Perspectives on Software Requirements, pp.169--199. Kluwer Academic Publishers. Springer US (2004)
8. Novak, J., Gowin, D.B.: Aprendiendo a aprender. Edic,Martínez Roca, Barcelona (1988)
9. Hadad, G.D.S., Doorn, J.H., Kaplan, G.N.: Creating Software System Context Glossaries. En: Mehdi Khosrow-Pour (ed) Encyclopedia of Information Science and Technology, 2ºed., pp.789--794. IGI Global, Information Science Reference, EEUU (2008)
10. Litvak, C.S., Hadad, G.D.S., Doorn, J.H.: Correcciones semánticas en métodos de estimación de completitud de modelos en lenguaje natural. En: 16th Workshop on Requirements Engineering, pp.105--117. Montevideo (2013)
11. Dyer, M.: Verification-based inspection. En: 26th annual Hawaii international conference on system sciences, pp 418--427 (1992)
12. Diccionario de la Lengua Española, Real Academia Española, 23º edición (2014)
13. García Negroni, M.M.: Escribir en español: claves para una corrección de estilo, 2da ed. actualizada. Santiago Arcos, Buenos Aires (2011)
14. Gallego, D., Ongallo, C.: Conocimiento y Gestión. Pearson Prentice-Hall, Madrid (2004)
15. Novak, J., Cañas, A.: The Theory Underlying Concept Maps and How to Construct and Use Them. Technical Report, Florida Institute for Human and Machine Cognition (2008)

Adopción de BPMN para Modelado CIM y su Transformación hacia Vistas de un modelo PIM

Leopoldo Nahuel[1,2], Leandro Rocca[1], Cecilia Ariste[1], Matías Caputti[1], Iván Zugnoni[1], Julieta Ponisio[1], Lautaro Mendez[1], Roxana Giandini[1,2]

[1] Laboratorio de Innovaciones en Sistemas de Información, LINSI.
Universidad Tecnológica Nacional, Facultad Regional La Plata.
Calle 60 esq. 124, La Plata, Buenos Aires, Argentina.

[2] Laboratorio de Investigación y Formación en Informática Avanzada, LIFIA.
Universidad Nacional de La Plata. Facultad de Informática.
Calle 50 y 115, La Plata, Buenos Aires, Argentina.

{lnahuel, lrocca, cariste, mcaputti, izugnoni, jponisio, lmendez, rgiandini}@linsi.edu.ar

Resumen. Bajo un desarrollo tradicional, cualquier cambio producido sobre un requisito, impactaba sobre el código fuente, produciendo inconsistencias y dificultades de mantenimiento. Actualmente, es el desarrollo dirigido por modelos (Model Driven Development, MDD), el paradigma más creativo que pretende resolver los problemas típicos del desarrollo de aplicaciones. En este trabajo se destaca la importancia de involucrar a los especialistas del dominio y usuarios del sistema, desde las primeras etapas a través de la obtención de modelos CIM creados entre informáticos y los stakeholders. Esta participación entre analistas y usuarios, tiene lugar a través de lenguajes de modelado desprovistos de conceptos computacionales que los usuarios puedan manejar rápidamente, como BPMN. Los modelos creados, servirán para la confección de un modelo CIM dentro de un desarrollo MDD, sobre el cual se aplicaran transformaciones para la obtención de ciertas vistas de modelo PIM, como diagramas de actividades y diagramas de clases conceptuales.

Palabras claves: Modelado de Software, BPMN, MDD, Transformaciones de modelos, ATL.

1 Introducción

El desarrollo de la informática en las últimas cuatro décadas ha estado signado por la elevación del nivel de abstracción de las tecnologías de desarrollo. Pasamos de lenguajes de bajo nivel escritos sobre editores de texto básicos, ha lenguajes de alto nivel, cada vez más cercanos al lenguaje humano, y a ambientes de desarrollo integrados muy sofisticados con multitud de asistencias al programador. Este proceso ha sido guiado por la necesidad de lograr aplicaciones cada vez más carentes de errores; fáciles de mantener y actualizar; procesos de desarrollo flexibles a los cambios en los requerimientos; y sobre todo, por la necesidad de que el software se ajuste bien a los requerimientos y cumpla los objetivos por los cuales fue

desarrollado. Esto último, aunque es obvio, es el motivo principal de todos los avances en desarrollo de aplicaciones desde el surgimiento del concepto "crisis del software" en la década de los 60.

En los últimos años hemos sido testigos del surgimiento de nuevas tecnologías que avanzaban en esta misma dirección: herramientas CASE [1] complejas, Frameworks, lenguajes de cuarta generación, etcétera. El paso siguiente fue evidente, y se comenzó a concentrar esfuerzos en la generación automática de código. Actualmente, el desarrollo dirigido por modelos (MDD por sus siglas en inglés) [2] se alza como el paradigma más novedoso para la creación de software. El Object Management Group [3] ha llevado a este paradigma a un estándar llamado Model Driven Architecture (MDA, pos sus siglas en inglés) [4]. La idea central subyacente a esta tecnología es que los modelos guíen todo el proceso de producción de software, y sea a partir de refinaciones sucesivas de estos, que el código se genere de forma automática. Bajo este paradigma, el eje se centra sobre el modelado, y con un motor de transformaciones, estos modelos van transformándose en otros más precisos y robustos, hasta llegar a la instancia en la que el próximo paso sea el código compilable (o interpretable). Los modelos en MDD tienen nombres que identifican el nivel de abstracción que tienen, desde el de más alto nivel llamado CIM (Computer Independent Model), pasando por el PIM (Platform Independent Model), hasta el más concreto de todos antes del código: el PSM (Platform Specific Model). De manera que partiendo del modelo de un CIM, mediante transformaciones automáticas se pueden obtener modelos más refinados y robustos, que contemplen las tecnologías de desarrollo y las plataformas subyacentes, hasta llegar a un modelo PSM que será transformado a código que puede ser ejecutado.

Alrededor de estas nuevas formas de desarrollo de software, se han generado múltiples técnicas y tecnologías que ayudan a completar el paradigma MDD y las transformaciones. Entre estas herramientas se encuentran los lenguajes de especificación de transformaciones, como ATL [5]. El Atlas Transformation Language es un lenguaje para escribir transformaciones. Funciona integrado con el ambiente de desarrollo Eclipse a través de un plugin, y es capaz de realizar transformaciones modelo a modelo. Para que este motor de transformaciones funcione, el modelo fuente y el destino deben tener una definición precisa a través de un lenguaje gráfico de más bajo nivel que los especifique (de manera análoga a como BNF describe lenguajes formales). El metamodelo estándar que describe los modelos gráficos utilizados en MDD es MOF (Meta Object Facillity) [6] Esto concluye una plataforma tecnológica integral capaz de lograr transformaciones modelo a modelo escritas en ATL, dentro de un proceso de desarrollo dirigido por modelos. El proceso señalado puede darse en la etapa de CIM a PIM, como de PIM a PSM. El nivel de abstracción de los modelos indicaran en qué paso del desarrollo nos encontramos, pero las técnicas explicadas en este trabajo pueden extenderse a modelos de todos los niveles de abstracción.

A través de modelos con altos niveles de abstracción, es posible involucrar a personas ajenas a la informática, pero relacionadas en el dominio de implantación del sistema en desarrollo. Esta interacción puede conseguirse a través de modelos BPMN [7], los cuales pueden fácilmente ser entendidos, e incluso, desarrollados por los propios usuarios. Esto es altamente beneficioso, ya que incorpora una herramienta más para la recopilación de requisitos del sistema, al clásico abanico de posibilidades

que tenían los analistas tradicionalmente. El modelado de procesos de negocio, realizado por los propios involucrados e interesados del sistema, provee una fuente confiable y sólida de información que los analistas pueden utilizar para la confección de un CIM, o parte de él. Este modelo CIM, bajo un desarrollo dirigido por modelos, se puede transformar en diversos diagramas UML (con las limitaciones de semántica que BPMN tiene, claro está) que completaran el CIM, y ayudarán a la creación de un modelo PIM, más preciso y útil para las etapas de desarrollo siguiente.

2 Marco Teórico: antecedentes y trabajos previos

Dentro del marco teórico y los objetivos que marcan el contexto de este trabajo, se deben mencionar algunos artículos previos. Los mismos forman la base de conceptos y procedimientos que aquí se detallan. En primer lugar, en [8] se menciona la idea del uso del lenguaje BPMN en un entorno de desarrollo ágil. Continuando el desarrollo de este aporte, en [9] se destaca la construcción de modelos de procesos de negocio (Business Procces Diagram) como parte del desarrollo de un modelo CIM que refleje los requerimientos del dominio del sistema. Por último, en [10] se presenta una alternativa sobre software libre para la construcción de una plataforma de transformación de modelos, presentando algunos ejemplos de transformación tomando como entrada modelos BPMN. A partir de estos desarrollos, el presente artículo avanzará sobre éstas ideas, aunando los esfuerzos conseguidos, y formalizando una propuesta de transformación con inclusión de usuarios (que por lo general no tienen conocimientos de computación) que ayuden al modelado de un CIM que sirva como entrada para lograr parte de un PIM a través de transformaciones automáticas.

3 Un motor para transformaciones Model-To-Model en Eclipse

El entorno para desarrollo de software Eclipse [11], en conjunción con el lenguaje ATL, nos proveen un ambiente adecuado para la confección de un motor de transformaciones capaz de cumplir con las primeras etapas de un proceso MDD.

Un proceso de transformación completo haciendo uso de ATL, se compone de los siguientes elementos: metamodelos fuente y destino; modelo fuente; código ATL. En la figura 1 se puede observar la interacción de estos componentes. La definición de todos los metamodelos está basada sobre MOF. Si bien no se mencionó, aunque ATL es un lenguaje textual, también puede definirse con MOF.

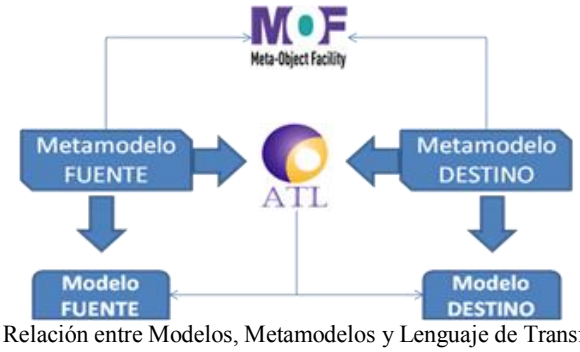


Figura 1: Relación entre Modelos, Metamodelos y Lenguaje de Transformación.

3.1 El Metamodelo Fuente y el Metamodelo Destino

La creación de MOF como estándar regulado por la OMG, trajo consigo una formalidad que se echaba en falta en el modelado de sistemas. Los lenguajes gráficos aparecieron, pero al revés de los lenguajes textuales, éstos no tenían una base formal desde dónde ser definidos. A partir de MOF esto cambio, y los lenguajes de modelado de software, pasaron a contar con un metalenguaje que los definía de manera precisa. Con MOF se han definido una gran diversidad de lenguajes gráficos, desde UML hasta BPMN, pasando por otros no tan relacionados al software, como las redes de Petri.

Un metamodelo es un modelo a partir del cual puede ser instanciado todo un lenguaje de modelado. Todos los lenguajes gráficos aquí mencionados tienen su metamodelo escrito con MOF. A través del estándar XMI [12], los metamodelos MOF pueden ser transcriptos en forma textual. Esto es aprovechado por ATL y el motor de transformaciones. De manera general, cada elemento del metamodelo de un lenguaje, va a tener una etiqueta XMI que lo define, conjuntamente con algunos atributos específicos.

Ejemplo 1: etiquetado para el elemento "class" de UML:

```
<eStructuralFeatures xsi:type="ecore:EReference"
name="class" ordered="false"
eType="#//Class" changeable="false" volatile="true"
transient="true">

 <eAnnotations
source="http://www.eclipse.org/emf/2002/GenModel">

<details key="documentation" value="References the Class
that owns the Property."/>

 </eAnnotations>

      <eAnnotations source="subsets"
references="#//NamedElement/namespace
#//Feature/featuringClassifier"/

 </eStructuralFeatures>
```

Para los fines de este artículo, se seguirá un ejemplo sencillo de transformación desde un diagrama BPMN, hacia tres tipos de diagramas UML: diagrama de actividades, de Casos de Uso, y de clases. La elección de BPMN se debe a que es un lenguaje desprovisto de conceptos computacionales y técnicos, lo cual lo hace especialmente útil para la confección de modelos CIM en cuya elaboración se encuentren involucrados personas que formen parte del dominio del sistema, y que no sean informáticos.

Los metamodelos, tanto fuente como destino, deben estar escritos de manera textual con XMI. A partir de los mismos, la transformación ATL es ejecutada.

3.2 El lenguaje ATL y una transformación Model-To-Model

El motor de transformaciones puede interpretar el código ATL y pasarlo a un bytecode específico que una máquina virtual es capaz de ejecutar. De esta manera, el

toolkit que se integra a Eclipse como un plugin, cuenta con tres capas: el lenguaje ATL, el compilador (que compila a bytecode), y la máquina virtual para ejecución del bytecode. El proceso de interpretación y ejecución es completamente transparente para el desarrollador. Solo basta con escribir las transformaciones y contar con los metamodelos involucrados.

El lenguaje ATL cuenta con dos modos de programación: imperativo y declarativo. El elemento principal para describir una transformación es llamado regla. Las reglas sirven para indicar cuál elemento del modelo fuente, se convertirá a otro elemento del modelo destino. Esta relación es unívoca, de manera que para cada elemento (o conjunto de ellos) del modelo origen, existe una, y solo una relación con otro elemento del modelo destino. Existen tres formas para declarar una regla: las "matched rules" que corresponden a la forma declarativa de programación; y las "lazy rules" y las "called rules" que corresponden a la forma imperativa. En este trabajo usaremos la forma declarativa, escribiendo la transformación con reglas matched.

Ejemplo 2: forma típica de declaración de reglas en forma declarativa:

```
-- @path bpmn=/test2/bpmn1.ecore
-- @path UML=/test2/MAPS_UML.ecore
module trans;
create OUT: UML from IN: bpmn;
-- Reglas de transformación

rule BPMNDiagram2UMLDiagram {
from
s: bpmn!ProcessModel
to
t: UML!UMLDiagram (
name <- s.name,
activity <- s.pools)}

rule Task2Activity {
from
s: bpmn!Task
to
t: UML!Activity (
name <- s.name)}
```

Las líneas que comienzan con "**--@path"**, muestran cómo indicarle al motor de ATL dónde encontrar los metamodelos (archivos con extensión .ecore) y darles un alias para usar en el código.

Ejemplo 3: una etiqueta XMI para el elemento "Activity" del metamodelo de UML se escribe de la siguiente forma:

```
//Metamodelo BPMN

<eClassifiers   xsi:type="ecore:EClass"   name="Activity"
abstract="true"            eSuperTypes="#//FlowObject"/>
<eClassifiers     xsi:type="ecore:EClass"     name="Task"
eSuperTypes="#//Activity"/>
……

//Metamodelo UML
```

```
<eClassifiers xsi:type="ecore:EClass" name="Activity"
eSuperTypes="#//Behavior"/>
```

4 Uso de BPMN para la construcción de un modelo CIM

En MDD, el primer modelo que se confecciona es el modelo independiente de la computación, o CIM. Este modelo está integrado por diversos elementos y diagramas, que están (o deben estarlo) alejados de cualquier tecnología, idea, o noción de informática. En la literatura sobre MDD, se comienza a hablar sobre transformaciones automáticas a partir del modelo PIM. Este modelo, incorpora mayor precisión y está compuesto por elementos relacionados con la computación, aunque en conceptos generales, y todavía se mantiene aislado de la plataforma específica de implementación. Es a partir de este modelo, que MDD comienza a transformar para lograr obtener el código en una última etapa. Resulta conveniente pensar en los beneficios de una transformación automática, desde antes de un modelo PIM, es decir, desde el primer modelo: el CIM. Para este fin, se dispone a crear diagramas de procesos de negocio con un lenguaje gráfico de alto nivel de abstracción: el BPMN. A partir de estos diagramas, seremos capaces de lograr tres tipos de transformaciones para llegar a tres diferentes modelos destinos: un diagrama de casos de uso; un diagrama de actividades; y un diagrama conceptual de clases.

4.1 Modelado con BPMN como técnica adicional para los analistas de negocio

Se han mencionado ya los beneficios que se obtendrían haciendo participes de la etapa de modelado a los usuarios y especialistas del dominio. Esto incluso puede verse como una herramienta más en la adquisición de requerimientos en la primera etapa de desarrollo. Las técnicas tradicionales, como entrevistas, lectura de documentación, etc., suelen estar acompañadas de ciertas incertezas, y ambigüedades. La incorporación de esta técnica podría aportar más valor al proceso inicial del desarrollo.

Las personas que están relacionadas al dominio del sistema, o aquellos interesados en su implantación, generalmente serán ajenas a la disciplina informática. Sería muy difícil por lo tanto, pedirles que hagan diagramas haciendo uso de un lenguaje como UML. Se necesita entonces, un lenguaje que sea fácil de entender, y de un algo lenguaje de abstracción que transparente todos aquellos conceptos con los cuales las personas no estén involucrados. El lenguaje de Notación para Modelado de Procesos de Negocio (BPMN por sus siglas en inglés), adquiere relevancia en este apartado. Su intuitiva sintaxis y su simplicidad, hacen que sea relativamente sencilla su utilización. De esta manera, se puede pedir a los stakeholders que modelen procesos con BPMN, los cuales serán la entrada de nuestras transformaciones. En el proceso de modelado de estos procesos de negocio no hay involucradas personas intermedias. El volcado de información sobre los mismos es directo, y por ende, se minimiza la posibilidad de malos entendidas o interpretaciones erróneas.

4.2 Limitaciones semánticas de BPMN para transformaciones automáticas

Una transformación entre modelos, solo tiene lugar gracias a las características semánticas de los lenguajes involucrados. Se debe mapear entre elementos semánticos similares entre un modelo fuente, y otro modelo destino. Mientras más significados, conceptos y relaciones compongan un modelo fuente, las posibilidades de

transformación se incrementaran. El BPMN es un lenguaje acotado en comparación con UML. Esto es debido a la diversidad y complejidad de un lenguaje de modelado como UML, el cual es de propósito general. Los diversos diagramas que componen su notación, hacen que éste sea capaz de modelar componentes del sistema, estructurales y de comportamiento. Para el caso de BPMN esto no es así, dado que fue creado con la finalidad de modelar procesos de negocio. Los elementos que componen éste lenguaje, y los diagramas creados a partir de los mismos, están relacionados con procesos y actividades dentro del sistema, lo cual estaría emparejado con la parte de modelado de comportamiento dentro de los modelos de UML. Es por esto que no es factible obtener la información necesaria, a partir de un diagrama BPMN, para lograr obtener una gran diversidad de otros diagramas a partir de una transformación. Menos aún si la intención es transformar a diagramas que modelen componentes de la arquitectura del sistema.

La idea de generar parte de un CIM con diagramas BPMN presenta limitaciones obvias por lo ya comentado. Pero sí es posible obtener un conjunto de diagramas que pueden ayudar al analista a tener un panorama más completo del sistema, con un mínimo esfuerzo a partir de la aplicación de transformaciones automáticas dentro de un desarrollo MDD. Más adelante se tratará una posibilidad de transformación hacia un modelo destino que contengo información estructural del sistema con un diagrama de clases conceptuales de muy alto nivel de abstracción, pero no por esto deja de ser de gran utilidad.

5 Transformaciones M2M entre BPMN y UML con ATL

Las transformaciones modelo a modelo, o model-to-model, son una parte fundamental dentro del desarrollo dirigido por modelos. Si bien se ha naturalizado que en un desarrollo MDD, el motor de transformaciones tiene participación a partir de un modelo PIM (generado manualmente) en este trabajo se plantea la posibilidad de comenzar con transformaciones entre modelos en una etapa anterior: desde el modelo CIM. El primer modelo, y por tanto más abstracto e independiente de la computación, es visto desde un desarrollo clásico dirigido por modelos, como un conjunto de elementos, de confección manual y cuyos constructores son solamente los analistas e involucrados con el desarrollo. El CIM, tradicionalmente, es la culminación de un proceso de relevamiento. Aquí se intenta ampliar esta concepción, adjudicando dos nuevas alternativas y conceptos en su construcción: por un lado se plantea la posibilidad de lograr transformaciones a partir del modelo CIM que culminen en un modelo CIM más completo, y que con algún esfuerzo pueda también resultar en parte de un modelo PIM; por otro lado, el uso de BPMN por parte de los stakeholders puede incluir beneficios en el proceso de construcción del software en las primeras etapas, al incluir en su desarrollo a aquellas personas interesadas en la implantación del sistema. Concretamente, las transformaciones descritas en este trabajo apuntan a lograr la automatización del modelado partiendo desde un modelo independiente de la computación. Los nuevos modelos obtenidos, serán parte de una nueva versión del CIM, más completa y diversa. Además, se prevé que con algo de trabajo adicional por parte de analistas, algunos de estos modelos sean parte de un PIM. Tal es el caso de los diagramas de clase obtenidos de una transformación planteada en una sección posterior, que pueden completarse con poco esfuerzo, y que podrían formar parte de un modelo menos abstracto que un CIM.

En las siguientes secciones se plantean tres transformaciones posibles partiendo desde un diagrama BPMN: diagrama de actividades, diagrama de casos de uso y un diagrama de clases conceptuales.

5.1 Transformación de diagrama de procesos de negocio (BPMN) a diagrama de actividades (UML)

La obtención de diagramas de actividades como conclusión de un proceso de transformación a partir de diagramas BPMN, resulta un proceso bastante simple. El mapeo es bastante directo y por ello es muy simple ver las relaciones entre los metamodelos y la transformación escrita en ATL. En el ejemplo de código 2 se puede ver la forma típica de declaración de reglas en forma declarativa. En el ejemplo 3, se ejemplificó la forma de etiquetado de XMI para el elemento "Activity" de metamodelo de UML. También, en el ejemplo de código 3, se pueden observar las partes del código XMI de los metamodelos BPMN y UML (fuente y destino de la transformación) que hacen referencia al elemento Actividad del diagrama BPMN y al elemento Actividad del diagrama UML. En el ejemplo 2 está el código de la regla que relaciona ambos conceptos y deriva uno a través del otro ejecutando la transformación. Observe que, en realidad, el elemento que empareja la regla declarada es "Task". En el metamodelo, "Task" es una instancia concreta que hereda de "Activity", la cuál es una clase abstracta. Esto es posible dado que en MOF son válidos los conceptos de objetos como clases y herencia.

5.2 Transformación de diagrama de Procesos de Negocio (BPMN) a diagrama de Casos de Uso (UML)

Las características semánticas de los modelos de procesos de negocio, hacen que sea posible obtener otro tipo de diagrama de comportamiento UML: diagramas de caso de uso. Se seguirá la metodología propuesta en [13] en la cual se plantea el siguiente mapeo entre elementos de los metamodelos:

1. **BpmnDiagram → Model**: tanto el diagrama fuente, como el destino, se encuentran circunscriptos en contenedores conceptuales que encuadran a los mismos bajo un nombre que los identifican. Esta relación mapea el nombre de los diagramas.

2. **SubProcess → UseCase**: cada subproceso del diagrama fuente, representa una ejecución particular del sistema, sobre el contexto que indica su carril. Por lo tanto, cada subproceso puede tomarse como una unidad funcional, y por lo tanto es posible mapear hacia un caso de uso UML.

3. **Activity (Task) → UseCase**: las actividades tienen un significado semántico similar a los subprocesos en BPMN, y es por esto que el mapeo es igual. Cada actividad, se convierte en un caso de uso.

4. **Activity (Gateway) → UseCase**: el elemento "Gateway" de la notación de BPMN, es en definitiva, una actividad en la cual se realiza un proceso de decisión para realizar una bifurcación en el proceso, para definir continuar por alguno de los diferentes caminos posibles. Este proceso, al igual que los subprocesos y las actividades comunes, se empareja a un caso de uso.

5. **Lane → Actor**: un "Lane" o carril, pone en contexto la ejecución de las actividades que se dan sobre él, y por ende otorga una categorización de las tareas. Esta organización puede expresarse en un rol el cuál es representado en un actor del diagrama de casos de uso a través de una transformación.

6. **SecuenceFlow → UseCase**: las flechas que indican un flujo de secuencia en BPMN denotan un orden de ejecución entre actividades. Este avance significa la finalización de una actividad, y el comienzo de otra. Entre ambas actividades, habrá un procedimiento que permita este cambio entre una y otra, dentro del curso de ejecución modelado. Por tanto, puede verse a esta ejecución de finalización y comienzo de actividad, como una unidad funcional del sistema, modelado en un caso de uso.

5.3 Transformación de diagrama de Procesos de Negocio (BPMN) a diagrama de Clases Conceptuales (UML)

Un diagrama de clases es parte de los componentes de modelado estructural de UML. Sus componentes modelan la arquitectura del sistema y está relacionado a elementos estáticos del sistema, alejado de los procesos de negocio modelados a partir de BPMN. Sin embargo, es posible obtener un primer acercamiento a un diagrama de clases, de un alto nivel de abstracción, a partir de modelos BPMN. Como ya se ha mencionado, todo modelo fuente tiene sus limitaciones en una transformación, y por tanto no es posible lograr cualquier tipo de modelo destino. Aquí se plantea un mapeo simple entre elementos de ambos lenguajes. El modelo de clases que se obtendrá constará de clases sueltes con algunos métodos. Los atributos y las relaciones entre clases podrán ser completados posteriormente mediando un proceso manual de los analistas con un esfuerzo considerablemente menor al que resultaría de realizar todo el proceso de forma manual.

Para conseguir un modelo estructural compuesto de clases sueltas con algunos métodos, a partir de un diagrama de un proceso escrito en BPMN, se plantea el siguiente emparejamiento entre elementos de los metamodelos que los componen:

1. **Carriles y Clases**: los carriles o "lane", son elementos que separan los procesos dentro de un diagrama BPMN según su rol o categoría. Un carril es el contexto donde se ejecutan ciertas actividades de un proceso, y por ende puede ser tomado como una Clase de UML que tendrá ciertos métodos relacionados a las actividades que se ejecutan dentro del carril.

2. **Task y Métodos de clase**: siguiendo la idea mencionada en un mapeo lane-clase, las actividades que se dan dentro de un carril, serán los métodos de la clase conseguida a partir de ese carril.

6 Conclusiones y Trabajo Futuro

La posibilidad de comenzar un desarrollo dirigido por modelos, mediante la aplicación de transformaciones automáticas a partir del primer modelo, se halla como una idea atractiva dentro de MDD. La bibliografía tradicional sobre el tema, plantea al modelo CIM como un producto del trabajo artesanal del analista, y compuesto por una heterogeneidad de componentes que pueden llegar a ser muy disímiles entre sí. La idea propuesta aporta homogeneidad y criterio al desarrollo de un modelo

independiente de la computación, con el agregado del diseño de tres tipos de transformaciones posibles a partir de diagramas BPMN. El producto obtenido, compone un CIM mucho más rico en contenido y semántica, lo cual contribuye a un mejor modelo del sistema sobre el cual se trabaja. A su vez, se presentó la idea subyacente al desarrollo de estas transformaciones, de lograr hacer uso de la experiencia de los stakeholders. Estas personas, que por lo general no tienen contenidos computacionales ni formación en informática, sí conservan una alta experiencia y un elevado conocimiento sobre el dominio del sistema sobre el cual se trabaja. La participación activa de estos usuarios, a través del modelado con BPMN, enriquece el proceso de obtención de requerimientos, y ayuda a los analistas a obtener un modelo CIM más claro.

Como líneas de trabajo futuro, se pretende lograr una unificación de las transformaciones propuestas, en una metodología de diseño de modelo CIM, capaz de orientar las actividades y aportar solidez al desarrollo. A su vez, se puede proponer otro tipo de transformación a partir de diagramas de procesos de negocio: obtención de diagramas de máquina de estados.

Referencias Bibliográficas

1. Sommerville, I. "Ingeniería del Software". 7° Edición. Pearson Education S.A., pp. 11, 79-83 (2005). ISBN-10: 8478290745.
2. Pons, C., Giandini, R. y Pérez, G. "Desarrollo de Software Dirigido por Modelos: conceptos teóricos y su aplicación práctica". 1°Edición. EDULP & McGraw-Hill Education, Argentina, pp. 28-38 (2010). ISBN-13: 9789503406304.
3. Object Management Group, OMG (2015). www.omg.org
4. OMG Contact: Watson, A. "Model Driven Architecture Guide". Version 2.0, (2014). http://www.omg.org/cgi-bin/doc?ormsc/14-06-01.
5. ATL/User Guide Introduction (2015) http://wiki.eclipse.org/ATL/User_Guide
6. Meta Object Facility. Core Specification. Version 2.4.2, (2014). http://www.omg.org/spec/MOF/2.4.2/
7. BPMN Specification. Versión 2.0.2, (2013). http://www.omg.org/spec/BPMN/2.0.2/
8. Giandini, R., Nahuel, L., Mendez, L., Mangano, M. "Los procesos ágiles en la producción de productos software en ambiente MDD". COMTEL (2012).
9. Mendez, L., Di Girolamo, R., Vargas, L., Pérsico, M., Santos, N. "Lenguajes notacionales y técnicas de ingeniería de software basada en modelos aplicado a entornos bpm". Jornadas de Estudiantes Investigadores, JEI (2012).
10. Perelli, J., Mendez, L., Pérsico, M., Martínez Astudillo, I., Blasi, N. "Desarrollo de prototipo case para transformación de modelos en contexto mdd aplicado a modelos BPMN". Jornadas de Estudiantes Investigadores, JEI (2012).
11. Eclipse IDE, https://eclipse.org/
12. XML Metadata Interchange, especification. Versión 2.4.2 (2014). http://www.omg.org/spec/XMI/2.4.2/
13. Giandini, R., Nahuel, L., Roca, L., Caputi, M., Zugnoni, I. "Implementando Transformación de Modelos utilizando MOSKitt Tool en adhesión al Paradigma MDD". Congreso Nacional de Ingeniería en Informática/Sistemas de Información, CoNaIISI (2014).

Técnicas de elicitación de requerimientos

Alejandro Oliveros[1], Leandro Antonelli[2]

[1]INTEC – UADE / UNTREF, Buenos Aires, Argentina
aoliveros@gmail.com
[2]LIFIA, Fac de Informática, UNLP, La Plata, Argentina
lanto2004@gmail.com

Abstract. El desarrollo de los requerimientos se alimenta del conocimiento obtenido a través del proceso de elicitación. Para obtener este conocimiento se utilizan ciertas *técnicas* específicas del proceso de elicitación que se alimentan de una variedad de técnicas. Actualmente son escasos los estudios empíricos de Ingeniería de Requerimientos que se concentren en el proceso de elicitación, paradojalmente parte relativamente importante de estos se produjeron en nuestra región. Esta investigación relevó en un conjunto particular de desarrolladores (estudiantes de Maestría en Ingeniería de Software) las técnicas de elicitación y las fuentes utilizadas en el desarrollo de los requerimientos. Tratándose de un estudio exploratorio no se pueden establecer formulaciones conclusivas, pero claramente surgen elementos consistentes con otras investigaciones y fuertes indicios para formular hipótesis de trabajo.

Keywords: Elicitación de requerimientos, investigación experimental, evolución de la práctica

1 Introducción

La producción de los requerimientos del software es sustancialmente interactiva e iterativa [1], el punto de arranque es el conocimiento que resulta del proceso de elicitación para el que se dispone de una serie de técnicas A menudo se hacen afirmaciones sobre enfoques de desarrollo sin datos que los soporten. Asimismo las actividades académicas necesitan conocer la real utilización de los enfoques que proponen. También es de interés de las organizaciones de desarrollo de las prácticas predominantes en sus mercados. No es menos importante poder establecer las diferencias que existentes con otros ámbitos de desarrollo de software.

La pregunta que se propone contestar este artículo es: ¿Cuáles son las prácticas de elicitación que utilizan las organizaciones de desarrollo de software?

Lo que sigue de este artículo está organizado de la siguiente forma. En el punto 2 se despliega el marco conceptual de la investigación y se revisa el estado del arte. En el punto 3 se detallan la metodología, procedimientos e instrumentos de la investigación. En el punto 4 se informan los resultados de la investigación y en 5 se formulan las conclusiones de la investigación y se enuncian trabajos futuros.

2 Prácticas de Elicitación de Requerimientos

2.1 Técnicas de Elicitación de requerimientos

El resultado del proceso de elicitación es "…todo el conocimiento relevante necesario para producir un modelo de los requerimientos de un dominio de problema" [1], para el que se dispone de una serie de técnicas.

Los documentos del CMMI incluyen una extensa variedad [2], que se ha ido enriqueciendo a los largo del tiempo (reflejo de la evolución positiva de la Ingeniería de Requerimientos). Por su parte el cuerpo de conocimiento de la ingeniería de software [3] enuncia como principales técnicas de elicitación a: entrevistas, escenarios, prototipos, reuniones con facilitadores (que incluyen los *brainstorming*), observación e historias de usuarios. En términos generales las taxonomías ayudan a comprender y comunicar el conocimiento [4]. La Ingeniería de Requerimientos dispone de varias taxonomías (o clasificaciones) de técnicas de elicitación.

La clasificación de Dieste y Juristo [5], originada en [6], identifica 43 Técnicas organizadas en 8 categorías: entrevistas, cuestionarios, introspecciones y observaciones, técnicas cognitivas (*contrived techniques*), selección de una lista de atributos, prototipos, análisis de escenarios y diagramación. En esta investigación se escogió una taxonomía más sencilla para no introducir categorías poco o nada utilizadas en el medio en estudio y para reducir la granularidad de respuestas.

En [1] se clasifican las técnicas en las siguientes categorías (no necesariamente disjuntas): partiendo del usuario, análisis de objetivo y meta, escenarios, análisis de formularios, lenguaje natural, reutilización de requerimientos y análisis de tareas. Esta taxonomía ubica a toda técnica en al menos una de las categorías y estas responden a diferentes criterios de definición.

Categoría principal	Componentes
Técnicas tradicionales	Cuestionarios; Surveys; Entrevistas (de comienzo y final abierto, estructuradas); Análisis de documentos (formularios, organigramas, modelos, estándares, manuales, normas, etc.).
Técnicas grupales	Brainstorming; Focus groups; RAD/JAD
Prototipos	Solo o combinado con otras técnicas
Técnicas orientadas por modelos	Métodos basados en objetivos; métodos basados en escenarios
Técnicas cognitivas	Laddering; Card Sorting; Repertory Grids
Técnicas contextuales	Métodos etnográficos (observación del participante); Etnometodología; Análisis de conversación (estudio de conversación e interacción)

Tabla 1. Taxonomía de técnicas de elicitación de Nuseibeh y Easterbrook

La clasificación de Nuseibeh y Easterbrook [7] incorpora los conceptos y técnicas más recientes tales como las técnicas cognitivas y contextuales. Las técnicas basadas en objetivos o en escenarios, orientan el conocimiento obtenido en términos de un

modelo previamente establecido. Ha obtenido un amplio reconocimiento en la comunidad. Estas características impulsaron a utilizarla en este proyecto (Tabla 1).

2.2 Fuentes de requerimientos

Identificar las fuentes del conocimiento necesario para la formulación de los requerimientos es un objetivo de la Ingeniería de Requerimientos [8] y una actividad dificultosa e importante para la *completitud* de la Especificación de Requerimientos.

Para esta investigación se elaboró un esquema de fuentes de requerimientos de tres niveles sin pretender ortogonalidad, la *taxonomía propuesta* (Tabla 2). Organizada en términos de *clases* de fuentes, *tipos* y *fuentes primarias*.

Clase	Tipo	Primaria
Entrevistas	Cantidad de personas	Individuales; Más de uno; Grupales
	Productos	Anotaciones propias; Anotaciones de terceros; Minutas propias; Minutas de terceros; Respuestas a cuestionarios
Análisis de formularios		Formularios
Desarrollo previo	Requerimientos	Documento de especificación de requerimientos; Diagramas de Casos de Uso
	Modelo de datos	Diagramas ER
	Diseño	DFD; Carta de estructura; Diagrama de clases; Diagrama de interacción; Diagrama de estados; Diagrama de actividades
	Software	Prototipos; Aplicaciones
	Manuales	Manuales del usuario; Manuales del operador
Productos del mundo "real"		Leyes; Reglamentos; Tratados; Normas internas; Estándares generales; Información institucional; Publicidad

Tabla 2. Taxonomía de fuentes de elicitación utilizada

Para evaluar la cobertura de la *taxonomía propuesta* se comparó con otras taxonomías ([9], [10], [11], [1]), asignando las categorías de estas a las clases de la *taxonomía propuesta* (ver Tabla 3). La conclusión es que las técnicas de elicitación de las cuatro clases de la *taxonomía propuesta* incluyen las técnicas de las cuatro taxonomías consideradas. En el cuestionario se agregó la opción de incorporar *otras* fuentes primarias y los entrevistados dispusieron de la opción de incorporar *otras* clases. El propósito fue construir una taxonomía para el proyecto y no una taxonomía de fuentes de requerimientos.

2.3 Estado del arte

El primer estudio de las prácticas de la Ingeniería de Requerimientos en el mundo real [12], analizó 23 proyectos en 10 organizaciones de desarrollo y los resultados se con-

centraron en la especificación de requerimientos sin referencias al proceso de elicitación. Varios estudios de las prácticas de Ingeniería de Requerimientos y su evolución siguieron a este, pero también sin atender a la elicitación [13], [14], [15], [16], [17].

Clase	**Zowghi y Coulin** [10]	**Alexander y Stevens** [9]	**Wiegers** [11]	**Loucopoulos y Karakostas [1]**
Entrevistas	Stakeholders y expertos del área	Entrevistas; workshops; mesa de ayuda y equipo de soporte; entrenadores y consultores	Entrevistas a potenciales usuarios; encuestas de marketing y cuestionarios de usuarios	Expertos del dominio; stakeholders del sistema que aloja el sistema software
Análisis de formularios				Formularios
Desarrollo previo	Procesos y sistemas existentes, documentación de sistemas y procesos de negocio	Prototipos; usos no intencionales de productos; diseños y especificaciones existentes; informes de problemas; sugerencias y quejas de consumidores; observar al usuario; representar lo que debe suceder; mejoras hechas por usuarios; productos rivales;	Informes de problemas y pedidos de mejora al sistema actual; especificaciones de requerimientos; observación del usuario; análisis de escenarios de tareas del usuario; productos competitivos actuales	Software disponible en el dominio; software similar en otros dominios
Productos del mundo "real"		Contratos mal escritos		Estándares; literatura acerca del dominio

Tabla 3. Comparación de taxonomías de fuentes de elicitación

En marzo-abril de 2002 se realizó una importante investigación en Estados Unidos de América [18], [19]. Se formularon 22 preguntas y respondieron 194 organizaciones, solamente una pregunta consultaba sobre las prácticas de elicitación (*What techniques do/did you use for requirements elicitation?*).En 2012 se informó una muy importante investigación [20] con el objetivo de establecer el impacto del conocimiento del dominio en la forma que los analistas conducen las entrevistas de elicitación. Estudiaron dos poblaciones independientes de estudiantes de posgrado obteniendo 27 y 31 respuestas. Se estableció que el proceso de elicitación de requerimientos es afectado en la comunicación con los consumidores y en la comprensión de sus necesidades. Lo que viene a confirmar presunciones preexistentes, pero detectaron que esos efectos pueden tener signo negativo o positivo. En 2013 se realizó un extenso survey [21] con 119 respuestas enfocado en las prácticas de la ingeniería de requerimientos, una sola de las preguntas estaba destinada a elicitación e indagaba sobre las técnicas utilizadas en elicitación.

En la región latinoamericana ha habido cierta actividad en este campo. En [22] se comunicó una investigación realizada en la Argentina en la que se entrevistaron 38 y

se enfocó en el proceso de elicitación, analizando las técnicas y las fuentes involucradas en la elicitación de requerimientos[1]. En 2013 se reportó un *survey* realizado en el seno de la comunidad de desarrollo del Brasil [23] en el que se analizaron las actividades de RE practicadas por las empresas de Porto Digital de Pernambuco, Brasil. Los resultados se compararon con los obtenidos en una investigación publicada en *Jama* Software. Se consideró el proceso de elicitación con resultados sobre las fuentes de requerimientos y se identificó la comprensión de las necesidades del cliente como el segundo gran desafío de los requerimientos. En 2014 se informó [24] un estudio de las prácticas de ingeniería de requerimientos para el desarrollo de aplicaciones Web en Argentina. Se consideró el proceso de elicitación y se estableció una frecuencia de uso de las principales técnicas de elicitación de requerimientos.

En esta revisión hemos encontrado muy pocos trabajos que analizan las prácticas de elicitación de requerimientos y de requerimientos en general.

3 Investigación de campo

3.1 Método de investigación

La investigación se realizó con estudiantes de un curso de posgrado[2]. Se utilizó un cuestionario con preguntas cerradas y con una abierta. Se ejecutó en un punto del curso que aseguró la homogeneidad del entendimiento de los conceptos utilizados.

El cuestionario lo soportaba una planilla de cálculo y se invitó a participar en la investigación por correo electrónico y por igual medio devolvieron el formulario completo. Se utilizaron criterios de participación informada en investigaciones que involucran seres humanos [25]. Participaron los 13 estudiantes del curso, los que provienen de diferentes regiones. Esta muestra la forman profesionales con iniciativa para cursar un posgrado, por lo que podrían considerarse activos respecto a la mejora de las prácticas de ingeniería de software. El tamaño resultante se encuentra en el rango de este tipo de estudios con llas restricciones propias de un estudio hecho a estudiantes. La constitución no probabilística y el tamaño de la muestra no permite sacar conclusiones generalizables, aunque brinda indicaciones de valor sobre el estado de la práctica. Cabe recordar el carácter exploratorio del estudio.

3.2 Cuestionario

El cuestionario está compuesto por 3 secciones de preguntas cerradas. La primera sección incluía datos personales, estudios académicos y experiencia en el desarrollo de software. La segunda sección incluía la experiencia en: *Técnicas tradicionales, Técnicas de elicitación grupales, Prototipación, Técnicas model-driven, Técnicas cognitivas, Técnicas contextuales*. La tercera sección encaraba los productos (fuentes)

---

[1] La investigación que comunica en este artículo forma parte de un proyecto de replicación de dicha investigación 14 años más tarde

[2] En el curso de Ingeniería de Requerimientos, de la Maestría de Ingeniería de Software de la Facultad de Informática de la Universidad Nacional de La Plata

de las que se elicitaban requerimientos. Un total de 29 productos se agruparon en 4 *clases*: *Entrevistas, Análisis de formularios, Desarrollos previos, Productos del mundo real*. Se solicitaba indicar la frase que mejor describía la relación con cada producto: *utilizo frecuentemente, utilizo ocasionalmente, alguna vez lo usé, conozco o conozco alguien que lo utiliza*.

4 Resultados

4.1 Perfil de los participantes

La *mediana de años de experiencia* en el desarrollo de software de los participantes era de 6 años (77% tenían menos de 10 años de experiencia) y la *mediana del año de graduación* era 2010. Un proyecto típico en el que se involucraban los participantes participaban 4 personas y duraba 9 meses

4.2 Resultados sobre las técnicas

La distribución de la utilización de las técnicas de elicitación (Tabla 4), muestra que las técnicas *tradicionales* son utilizadas por casi la totalidad de los entrevistados, lo que se encuentra en línea con otros estudios que identificaron a las *entrevistas* como una técnica predominante en la elicitación [24].

Técnicas	%	Técnicas	%
Tradicionales	92%	Orientadas por modelos	31%
Grupales	62%	Cognitivas	23%
Prototipación	23%	Contextuales	46%

Tabla 4. Técnicas de elicitación

Las técnicas *grupales* ocupan un segundo lugar muy por encima de las restantes. Este resultado es consistente con datos de utilización del *brainstorming* de otras investigaciones [21], [24]. Las restantes técnicas se encuentran por debajo del 50% en cuanto a su utilización, aunque las técnicas *orientadas por modelos* y las *cognitivas* han crecido en las últimas dos décadas [26].

Este comportamiento puede explicarse en términos de años de experiencia de los entrevistados. En los casos de *cuestionarios y encuestas* y *entrevistas* los que la utilizan se dividen en partes iguales entre los que tienen menos de seis años de experiencia o más (mediana de los años de experiencia). En el caso del *análisis de documentación existente*, de los 6 que la utilizan 4 tienen 6 o menos años de experiencia. En cuanto al uso de *brainstorming* de los 8 que las usan, 5 tienen 6 o menos años de experiencia. Estas dos últimas técnicas mencionadas sugieren que el uso de ellas se da principalmente entre aquellos que tienen menos años de experiencia.

Considerando a los entrevistados que tuvieron un curso de requerimientos en la carrera de grado (5 en total) utilizan *cuestionarios y encuestas* todo ellos. En cuanto al uso de las otras técnicas tradicionales tienen cuatro respuestas positivas cada una. La

técnica de *brainstorming* es utilizada por 3 de los 5. El resto no presenta tasas de utilización diferenciales del resto. Esto parece indicar que haber cursado una asignatura de Ingeniería de Requerimientos, no es un predictor de seguir un patrón de uso de técnicas más actualizado que el resto. En cuanto al manejo individual de varias técnicas, 9 entrevistados utilizan 3 o 4 técnicas de elicitación

Estos resultados contrastan con las recomendaciones de la literatura en cuanto a utilizar en el proceso de elicitación de requerimientos una variedad de técnicas. Por otra parte la difusión en las carreras universitarias que ha adquirido la Ingenierìa de Requerimientos (en parte derivada de exigencias explícitas de los organismos de acreditación de las carreras del área), permite prever un crecimiento de la utilización de técnicas actualmente subutilizadas.

4.3 Resultados sobre las fuentes

Se indagó acerca de la utilización de cuatro categorías de fuentes: *entrevistas, análisis de formularios, desarrollos previos y productos del mundo real* (Tabla 2) mediante la siguiente escala: *Usa regularmente, Usa ocasionalmente, Usó alguna vez, Conoce el producto, Conoce gente que lo usa*. Las respuestas se consolidaron en tres grupos de categorías: *Uso regular* (1era. Categoría), *Uso irregular* (2da y 3era); *Tiene conocimiento* (4ta y 5ta).

Entrevistas. En la utilización de las entrevistas como fuente de requerimientos (Tabla 5), predominan de las *entrevistas individuales* sobre las otras opciones. La *entrevista individual*, es muy profunda en términos de contenido pero lo restringe a un individuo y es muy difícil destinar gran cantidad de tiempo a entrevistarlos. De allí que sea una limitante al alcance: la duración del proyecto y la cantidad de desarrolladores involucrados tienen medianas menores que las del resto para los que son usuarios regulares de reuniones individuales.

Tipo	Primaria	Regular	Irregular	Conoce
Cantidad de personas	Individuales	62%	15%	8%
	Más de uno	31%	54%	8%
	Grupales	23%	31%	31%
Productos de las entrevistas	Anotaciones propias	85%	8%	8%
	Anotaciones de terceros	8%	62%	0%
	Minutas propias	38%	31%	15%
	Minutas de terceros	8%	46%	8%
	Respuestas a cuestionarios	15%	46%	8%
	Otros	15%	8%	0%

Tabla 5. Utilización de entrevistas como fuentes

Las entrevistas generan una serie de productos que se utilizan en diferente medida. Predominan los productos propios (*minutas* o *anotaciones*) sobre los de terceros. Ello introduce una seria limitación de alcance a la labor del ingeniero de requerimientos en la medida que solamente procesa sus propios productos.

Análisis de formularios. El 31% de los entrevistados hacían *uso regular* de los formularios como fuente de requerimientos. Entre ellos predominaban los que no había tenido curso de requerimientos en sus estudios.

Desarrollo previo. En la Tabla 6 se reproducen los datos de los productos (*work products*) de desarrollos previos que tienen un nivel de *Uso regular* significativo. En términos de la experiencia de los entrevistados el uso de los *Diagramas de Clase* tiene una fuerte asociación con la menor experiencia de los entrevistados, esto podría denotar que estos diagramas se han difundido en menor medida entre aquellos desarrolladores que ya estaban activos cuando surgió esta herramienta. Pero, en contraste con esta idea, la utilización de los Casos de Uso se asocia fuertemente con la mayor experiencia de los desarrolladores. Este contraste sugiere la necesidad de afinar un modelo de penetración de herramientas en la comunidad de desarrolladores.

Tipo	Primaria	R	I	C	Predominio
Requerimientos	Especificación	61%	31%	0%	sin tendencia
Diseño	Casos de Uso	46%	39%	0%	mayor experiencia
Diseño	Diagramas ER	77%	15%	0%	menor experiencia
Diseño	Diagrama de clases	69%	15%	0%	menor experiencia
Diseño	Diagrama de interacción	31%	46%	0%	sin tendencia
Diseño	Diagrama de estados	31%	31%	15%	sin tendencia
Diseño	Diagrama de actividades	39%	23%	15%	sin tendencia
Software	Prototipos	46%	15%	15%	menor experiencia
Software	Aplicaciones	54%	23%	0%	menor experiencia
Manuales	Manuales del usuario	38%	31%	15%	menor experiencia

Table 6. Uso de productos seleccionados

Productos del mundo "real". Las fuentes primarias de mayor nivel de utilización son los *Reglamentos* y *Normas internas* y la siguen en importancia *información institucional* y *Estándares generales*.

5 Conclusiones y trabajos futuros

Como se ha señalado se dispone de escasos estudios de las prácticas de elicitación. En esta investigación se detectó que las técnicas tradicionales de elicitación son las de mayor utilización en línea con otros estudios [24]. En segundo lugar aparecen las técnicas grupales en especial el *brainstorming*, también alineado con esos estudios [24]. Se observa una tendencia creciente de uso de técnicas más avanzadas. La mayoría de los entrevistados utiliza un conjunto reducido de técnicas (3 ó 4). En cuanto a las fuentes de requerimientos, aquellos que usan la información obtenida en entrevistas individuales se vinculan con proyectos de menor envergadura, lo que es previsible por alcance que puede lograrse con la entrevista individual.

En cuanto a la reutilización de documentos de desarrollos previos como fuente en el proceso de elicitación predominan documentos clave como la Especificación de

Requerimientos de Software, los Casos de Uso y los Diagramas de Clase. Aunque con diferentes composiciones en sus usuarios.

Los trabajos futuros se orientarán a formular una investigación con una muestra que trascienda el curso de posgrado, considerando desarrolladores en general y atendiendo a una muestra considerablemente mayor.

Referencias

[1] P. Loucopoulos and V. Karakostas, *Systems Requirements Engineering*. McGraw-Hill, 1995.

[2] CMMI Team Product, "CMMI® for Development, Version 1.3," Software Engineering Institute, Pittsburgh, Technical Report CMU/SEI-2010-TR-033, 2010.

[3] P. Bourque and R. E. Fairley, *SWEBOK. Guide to the Software Engineering Body of Knowledge. Version 3.0*, 3rd. ed. IEEE Computer Society, 2014.

[4] M. Unterkalmsteiner, R. Feldt, and T. Gorschek, "A Taxonomy for Requirements Engineering and Software Test Alignment," *ACM Trans. Softw. Eng. Methodol.*, vol. 23, no. 2, Article 16, p. 38, Mar. 2014.

[5] O. Dieste and N. Juristo, "Systematic Review and Aggregation of Empirical Studies on Elicitation Techniques," *IEEE Trans. Softw. Eng.*, vol. 37, no. 2, pp. 283–304, Apr. 2011.

[6] R. R. Hoffman, N. R. Shadbolt, A. M. Burton, and g. Klein, "Eliciting Knowledge from Experts: A Methodological Analysis," *Organ. Behav. Hum. Decis. Process.*, vol. 62, no. 2, pp. 129–158, 1995.

[7] B. Nuseibeh and S. Easterbrook, "Requirements Engineering: A Roadmap," in *ICSE '00 Proceedings of the Conference on The Future of Software Engineering*, Limerick, Ireland, 2000, pp. 35 – 46.

[8] J. A. Goguen and C. Linde, "Techniques for requirements elicitation," in *Requirements Engineering, 1993., Proceedings of IEEE International Symposium on*, San Diego, CA , USA, 1993, pp. 152 – 164.

[9] I. Alexander and R. Stevens, *Writing better Requirements*. Pearson, 2002.

[10] D. Zowghi and C. Coulin, "Requirements Elicitation: A Survey of Techniques, Approaches, and Tools," in *Engineering and Managing Software Requirements*, Berlin Heidelberg: Springer-Verlag, 2005.

[11] K. Wiegers, *Software Requirements*, 1st ed. Microsoft Press, 1999.

[12] Lubars, Mitch, C. Potts, and C. Richter, "A Review of the State of the Practice in Requirements Modeling," in *Proceedings of IEEE International Symposium on Requirements Engineering*, San Diego, CA , USA, 1993, pp. 2 – 14.

[13] L. Groves, R. Nickson, G. Reeve, S. Reeves, and M. Utting, "A survey of software development practices in the New Zealand software industry," in *Proceedings. 2000 Australian Software Engineering Conference*, Canberra, ACT, 2000, pp. 189 – 201.

[14] U. Nikula, J. Sajaniemi, and H. Kälviäinen, "A State-of-the-Practice Survey on Requirements Engineering in Small- and Medium-Sized Enterprises," Telecom

Business Research Center. Lappeenranta University of Technology, Lappeenranta, Finland, Research Report 1, 2000.
[15] T. Hall, Beecham, S., and Rainer, A., "Requirements problems in twelve software companies: an empirical analysis," *IEE Proc.*, vol. 149, no. 5, pp. 153 – 160, Oct. 2002.
[16] A. Talbot and A. Connor, "Requirements Engineering Current Practice and Capability in Small and Medium Software Development Enterprises in New Zealand," in *Proceedings 2011 Ninth International Conference on Software Engineering Research, Management and Applications*, Baltimore, MD, 2011, pp. 17 – 25.
[17] A. Tahir and R. Ahmad, "Requirement Engineering Practices – an Empirical Study," presented at the 2010 International Conference on Computational Intelligence and Software Engineering (CiSE), Wuhan, 2010, pp. 1 – 5.
[18] C. J. Neill and P. A. Laplante, "Requirements Engineering: The State of the Practice," *IEEE Softw.*, vol. 20, no. 6, pp. 40–45, Dec. 2003.
[19] P. A. Laplante, C. J. Neill, and C. Jacobs, "Software Requirements Practices: Some Real Data," in *Proceedings of the 27 th Annual NASA Goddard/IEEE Software Engineering Workshop (SEW-27'02)*, Goddard, 2002.
[20] I. Hadar, P. Soffer, and K. Kenzi, "The role of domain knowledge in requirements elicitation via interviews: an exploratory study," *Requir. Eng. J.*, vol. 19, no. 2, pp. 143 – 159, Jun. 2014.
[21] M. Kassab, C. J. Neill, and P. A. Laplante, "State of practice in requirements engineering: contemporary data," *Innov. Syst. Softw. Eng.*, vol. 10, no. 4, pp. 235–241, Dec. 2014.
[22] L. Antonelli and A. Oliveros, "Fuentes utilizadas por desarrolladores de software en Argentina para elicitar requerimientos," in *Proceedings del 5to. Workshop de Ingenieria de Requerimientos (WER'02)*, Valencia, España, 2002, pp. 106 – 116.
[23] D. Arruda, R. Soares, D. Vieira, R. Ferreira, T. Cabral, and M. Lencastre, "Engenharia de Requisitos: Um Survey realizado no Porto Digital, Recife/Brasil.," in *Proceedings of the XVII Ibero---American Conference on Software Engineering*, Pucón, Chile, 2014, pp. 477 – 490.
[24] A. Oliveros, F. J. Danyans, and M. L. Mastropietro, "Prácticas de Ingeniería de Requerimientos en el desarrollo de aplicaciones Web," in *Proceedings of the XVII Ibero---American Conference on Software Engineering*, Pucón, Chile, 2014, pp. 491 – 505.
[25] A. Oliveros and S. Martinez, "Aspectos éticos de la investigación en Ingeniería Software que involucra seres humanos," in *A nales del XVIII Congreso Argentino de Ciencias de la Computación*, Bahía Blanca, 2012, pp. 814 – 827.
[26] A. Oliveros and L. Antonelli, "Fuentes y técnicas para elicitación de requerimientos. Un estudio de evolución de la práctica," in *Proceedings XVIII Ibero-American Conference on Software Engineering*, Lima, Perú, 2015, pp. 563 – 577.

Estrategia para la Calificación y Priorización de Nuevos Requerimientos de Software Propietario Utilizando un Registro de Historial de Clientes

Sara De Federico[1], Mariela Avogradini[1], Noelia Sincosky[1], Javier Gago[1], Diana Moschetti[1], Lucía Melfi[1], Silvio Gonnet[2]

[1] Departamento Ingeniería en Sistemas de Información, Universidad Tecnológica Nacional Facultad Regional Rosario, Zeballos 1341. 2000 Rosario, Santa Fe, Argentina
{sdefederico, mavogradini, nsincosky, jgago, dmoschetti, lmelfi}@frro.utn.edu.ar
http://www.frro.utn.edu.ar/contenido.php?cont=604&subc=26

[2] INGAR Instituto de Desarrollo y Diseño, Conicet, UTN, Avellaneda 3657 S3002GJC Santa Fe, Argentina
sgonnet@santafe-conicet.gov.ar,
http://www.ingar.santafe-conicet.gov.ar/institucional/ personal/investigadores/silvio-gonnet/

Resumen. La priorización y selección de un nuevo requerimiento para su incorporación a un producto de software establecido en el mercado, es un punto decisivo en la etapa de mantenimiento del software. De un estudio de las empresas de la ciudad de Rosario y alrededores, se obtuvo información de la utilización de gestores de procesos e incidencias como apoyo a la priorización y selección de las funcionalidades solicitadas. Sin embargo, estas herramientas no ofrecen un análisis de todos los aspectos para una toma de decisión óptima, que no solo se basa en la aplicación de métricas sino en un estudio profundo de los resultados de la implementación del nuevo requerimiento dentro del sistema. Este estudio analiza el impacto en la empresa, en los usuarios propietarios del producto y los potenciales que se pueden sumar tras una nueva versión, como en el historial del usuario solicitante. Este trabajo propone una estrategia de priorización y selección de nuevos requerimientos en software propietario en etapa de mantenimiento y versionado, y la implementación de la etapa inicial de esta estrategia, la creación de una Base de Datos Integral de Clientes.
Palabras clave: Priorización, Requerimientos, Mantenimiento, Priorización de Requerimientos según Clientes, Historial de Clientes.

1 Introducción

La Ingeniería de Requerimientos ha sido una de las ramas de la Ingeniería de Software que en los últimos tiempos ha adquirido particular importancia. La negociación de las prioridades en la construcción de las listas de requerimientos influye directamente en el éxito posterior del desarrollo [1]. Los fracasos en desarrollos de software y la corta vida de algunos productos evidencian la importancia de la buena interpretación de un requerimiento, y de una lista de requerimientos que priorice los de mayor valor para la

construcción del sistema [2]. Al incorporar estos factores en el ciclo de vida de software, las empresas desarrolladoras tienen la seguridad de una implementación exitosa y la aceptación del producto por parte de los usuarios [3]. Cuando estas circunstancias se trasladan a la incorporación de nuevas funcionalidades a un software con una posición establecida en el mercado, se instala un desafío de equilibrio entre satisfacer las necesidades crecientes de los usuarios y mantener un producto estable y de calidad comprobada [4]. Para estos casos, la Ingeniería de Requerimientos debe contemplar situaciones paralelas a las generalmente observadas, como por ejemplo el impacto de la incorporación del nuevo requerimiento al sistema, y el efecto de la nueva versión en los otros usuarios [5][6].

Si bien existen herramientas informáticas de apoyo a la toma decisiones que incorporan técnicas de priorización, la calificación inicial de cada requerimiento de software en etapa de mantenimiento involucra un conjunto de perspectivas que se interrelacionan en forma compleja. Si se contemplan todas las circunstancias que implican la selección de un nuevo requerimiento, su incorporación mantendrá los estándares de calidad obtenidos en el desarrollo [7] y permitirá afianzar el modelo de madurez del software alcanzado por el producto [8] [9]. En la literatura se destacan como criterios comúnmente empleados para la priorización la importancia, el costo, el daño, la duración, el riesgo, y la volatilidad de los requerimientos [6]. Sin embargo, a partir de un estudio realizado a las empresas desarrolladoras de software de la ciudad de Rosario y alrededores, se identificó que los criterios de priorización se centran en la *criticidad* del requerimiento. Pero las distintas perspectivas abordadas en la priorización de requerimientos están estrechamente relacionadas al cliente que solicita la nueva funcionalidad. Por tal motivo, a la hora de la elección de un nuevo requerimiento, se debe analizar profundamente qué usuario solicitó esta funcionalidad y su historial. La historia de los anteriores casos solicitados por un mismo cliente puede determinar la no elección de un nuevo requerimiento, teniendo en cuenta el comportamiento tanto financiero como personal del cliente, los resultados del desarrollo y su conclusión exitosa o no. Asimismo se debe tener en cuenta cómo se manejó el Cliente durante la implementación con el personal del área soporte o línea de ayuda, y finalmente, la opinión de los demás clientes con respecto a la nueva versión del producto.

En la sección siguiente se define el problema abordado y el contexto donde fue identificado, junto con técnicas y herramientas de priorización más usadas en dicho contexto. Luego, en la Sección 3 se presenta una estrategia de calificación y priorización de nuevos requerimientos a partir de información de los Clientes de una empresa desarrolladora de Software. En esta misma sección se delinean los siguientes pasos del trabajo. Finalmente se presentan las conclusiones del mismo.

2 Definición del problema

Este trabajo surge de un estudio realizado a las empresas desarrolladoras de software de la ciudad de Rosario y alrededores, de las cuales se obtuvo información a través de encuestas, en las que participaron el 38% de las empresas más influyentes y de peso del mercado. Se realizó un análisis descriptivo de preguntas booleanas y cerradas, análisis cualitativo de preguntas abiertas [10], y clustering de datos para la detección de grupos

[11] [12]. Los resultados de estas etapas muestran que los procedimientos utilizados se apoyan en: la experticia de las líneas gerenciales; un uso esporádico de métricas; manejo con datos dispersos a lo largo de toda la empresa y con la intervención de diferentes departamentos y secciones; la actuación de personal de distintos tipos y categorías; sin un seguimiento consistente a lo largo de todo el ciclo de desarrollo del nuevo requerimiento; no muchas mediciones posteriores del impacto de la implementación sobre la empresa, el sistema y la comunidad de usuarios. Los criterios de priorización se centran en la *criticidad* del requerimiento, siendo ésta definida por la rama gerencial en función de las reglas de negocio de la empresa [13]. La Tabla 1 expone estos resultados y la medición de los puntos indicados.

Tabla 1. Conclusiones del análisis de la información obtenida de las encuestas a empresas de Rosario y alrededores

Punto de importancia en el proceso	Grado de uso	Observaciones
Tiene historial de requerimientos	Muy alto	Historial de pedidos en narrativas
Clasificación de los requerimientos	Muy alto	La escala de valoración es cualitativa, uso relativo de herramientas
Tiene historial de los clientes	Muy alto	Historial financiero no directamente relacionado con cada pedido
Dispersión de la información	Muy alto	A lo largo de todas la áreas
Se genera una nueva versión tras la implementación	Muy alto	Distribución compulsiva a todos los usuarios
Problemas con el desarrollo	Muy alto	Muy variados, íntimamente relacionados con el usuario solicitante
Diferenciación entre bugs y requerimientos	Muy alto	Se tratan por diferentes vías
Interés en poseer una herramienta integral de priorización	Alto	Predisposición a la implementación
Aplica Criterio de *criticidad* para priorizar	Alto	Definición diferente para cada empresa
Otras formas de clasificación	Alto	Amplia variedad y características
Interés en la opinión del usuario solicitante	Alto	Encuestas de satisfacción y mesas de ayuda al cliente
Seguimiento del desarrollo	Leve	Sólo algunas áreas de la empresa
Uso de métricas y métodos	Leve	Algunas empresas, no se detallan cuáles
Impacto financiero	Leve	Diferentes técnicas para evaluar
Impacto sobre el producto	Leve	Test de producto
Documentación integral	Bajo	Datos dispersos o incompletos
Uso de herramientas informáticas para la priorización	Bajo	Herramientas de monitoreo del ciclo de vida del software
Captación de opinión de los otros usuarios	Bajo	Concentración en el usuario solicitante

2.1 Técnicas de priorización

Las empresas encuestadas expresaron usar técnicas de priorización conocidas, aunque no destacaron una en especial. Existe amplia bibliografía acerca de técnicas y métodos

de priorización de requerimientos, y algunas técnicas como la Analytical Hierarchy Process (AHP), se encuentran entre las más elegidas por los líderes de proyectos según el consenso de autores y presentaciones. Otras técnicas elegidas son: Cumulative Voting, Kano Model, Quality Functional Deployment (QFD) [14]. Generalmente los autores explican las técnicas en la especificación de requerimientos al inicio del ciclo de vida del producto, pero también son consideradas como parte del proceso de administración de cambios de requerimientos [6]. Existe una gran variedad de software de toma de decisiones que implementan varias técnicas de priorización.

AHP, creado por Saaty en 1980 [15], realiza comparaciones por pares de alternativas asignando prioridades, este proceso involucra diferentes criterios de calificación. Se utiliza una escala fundamental para la comparación de a pares y se crean matrices de comparación de las diferentes alternativas. El número de comparaciones aumenta dramáticamente al aumentar el número de requerimientos.

Cumulative Voting o la prueba de “los 100 dólares” es muy sencilla y consiste en valorar hipotéticamente a cada requerimiento con un monto imaginario de dinero teniendo en total solo 100 unidades disponibles. Si bien por su sencillez es la más usada, depende de la subjetividad de los tasadores [2].

Kano Model, creado por Noriaki Kano clasifica los requerimientos en solo cuatro categorías, todas orientadas a la satisfacción del cliente y a la excelencia del producto [16].

QFD utiliza matrices para representar cronológicamente las expectativas del cliente y cómo deben cumplir los desarrolladores para cubrirlas. Hace un alto énfasis en las necesidades del cliente. La matriz más usada es la llamada Quality House creada por Mitsubishi en 1972 [17].

Estas técnicas son poderosos instrumentos para la priorización, pero todas requieren una calificación previa de cada requerimiento. Esta calificación debe estar basada en todos los intereses y variables que intervienen y afectan al proceso de implementación de un nuevo requerimiento a un software en uso. En el caso del AHP esto se evidencia en la selección de los criterios y la puntuación de cada requerimiento con respecto a cada uno de ellos. De las encuestas a las empresas se obtuvo a partir del análisis léxico de preguntas abiertas un conjunto de criterios para la priorización, que son: problemas (urgentes, graves o críticos), control temporal o antigüedad del requerimiento, tipo de cliente que lo solicita, calificación del mismo por un experto perteneciente a la empresa, una evaluación del impacto sobre el código y costo, y por último el valor del requerimiento para el negocio o el sistema. Asimismo las empresas no indicaron claramente el procedimiento, pero aclararon no usar métricas, cálculos matemáticos o estructuras formales.

2.1 Uso de Gestores de Desarrollo para la priorización

Si bien en las encuestas no se especificaron herramientas, es común el uso de gestores de desarrollo que tienen opciones de priorización de requerimientos. Según información obtenida de los empleados de las empresas encuestadas, los gestores más comúnmente utilizados son Target Process, Jira y Team Foundation Server, todos con un alto nivel de reconocimiento en el mercado. Sin embargo, en cuanto a la priorización de los requerimientos (tickets) todos presentan algún tipo de restricción. A continuación se

expone un detalle resumido provisto por personal que usa directamente los gestores de desarrollo acerca de las deficiencias y problemas que poseen:
Target Process: Su gestor de incidentes es deficiente y poco flexible. No maneja el concepto de cliente, por lo tanto no brinda estadísticas acerca de él. Tampoco maneja el concepto de presupuesto ni el concepto del valor monetario de las horas. No existe el concepto de horas compradas de antemano por el cliente. La única forma de priorización es el orden en la lista de tickets. No se puede asociar una historia al cliente.
Jira: Los tickets se organizan por proyecto por lo tanto no permite la organización de tareas cross proyecto, es decir no se puede priorizar un requerimiento por sobre otro de otro proyecto diferente. No brinda ninguna forma de calificación de los tickets, solo se le agrega una prioridad obtenida en un análisis por fuera del gestor. No brinda estadísticas que sirvan para maximizar recursos y ganancias. Defectos de usabilidad.
Team Foundation Server: Entorno muy poco amigable, necesidad de crear cuentas de Microsoft. No posee repositorio para guardar documentación asociada. Dificultades para heredar tareas inconclusas de un sprint al otro. Necesidad de actualización permanente.

Viendo los problemas y deficiencias que declaran los usuarios de los gestores, y la forma en que usan las técnicas, se destaca permanentemente la falta de métodos, herramientas y datos estadísticos de apoyo, para obtener un fundamento que ayude a la calificación de un requerimiento para su priorización. Aunque los gestores brindan listados de incidentes, tickets, bugs, etc., la calificación y la priorización depende de la opinión de los Team Leaders, que previamente deben hacer reuniones y tratar de armar las tareas basándose la mayoría de las veces en su capacidad y preguntando a otros sectores en forma informal y subjetiva. Para obtener una estrategia efectiva de apoyo a la calificación y priorización de nuevos requerimientos se deben considerar los siguientes puntos:

1. Definición de criterios para calificación de los clientes: Visión de las características del cliente desde el punto de vista de las áreas Finanzas, Gerencia, área de Soporte, Desarrollo. Apreciaciones basadas en la historia de las solicitudes anteriores.
2. Historial del cliente (para los requerimientos anteriores): Situaciones en el desarrollo. Conclusión exitosa o no del desarrollo. Problemas y cambios presupuestarios. Resultados del impacto sobre el sistema (código). Resultados del uso de los recursos de la empresa. Resultados del impacto sobre otros clientes. Mediciones de la satisfacción sobre la implementación. Otros datos relevantes
3. Calificación de los clientes en función de los puntos 1 y 2
4. Calificación de requerimientos en función de la historia del cliente
   Priorización de los requerimientos: Uso de diferentes técnicas. Adecuación a las reglas de negocio (matrices de selección)

3 Estrategia de calificación y priorización de nuevos requerimientos: Creación del Historial de Clientes

A partir de los problemas detectados se perfila la construcción de una Estrategia de calificación y priorización de nuevos requerimientos teniendo en cuenta todos sus aspectos técnicos y cualitativos [5]. No deben dejarse de lado las técnicas más usadas,

sino incorporarlas como opciones de análisis o toma de decisiones [14]. Para la construcción se tomaron las premisas obtenidas en las encuestas, y los problemas detectados en las herramientas, técnicas y métodos. De ellas se desprendieron un conjunto de tareas, que fueron validadas directamente con representantes de las empresas, quienes fueron identificando las diferentes tareas que se hallaban *embebidas* dentro de diferentes procedimientos propios y guiaron en la selección de nombres para una rápida comprensión. De esta interacción nació un primer diagrama de procesos que fue revelado usando la herramienta Bizagi [19]. Dado que el proceso completo es complejo se lo separó en subprocesos. Se tomó especial detalle a generar un proceso de negocio que luego sea base de una futura herramienta informática teniendo en cuenta los lineamientos de uso en las empresas [20]. Esta primera versión, particionada en subprocesos se volvió a exponer a los colaboradores en varias reuniones con representantes de todas las áreas. Junto con el diagrama se acompañaron gráficos explicativos de las tareas involucradas. En la Figura 1 se muestra la primera versión del proceso, en la cual se observan tres subprocesos *Sección Clientes* (usuarios), *Sección Requerimiento*, y *Sección Priorización.*

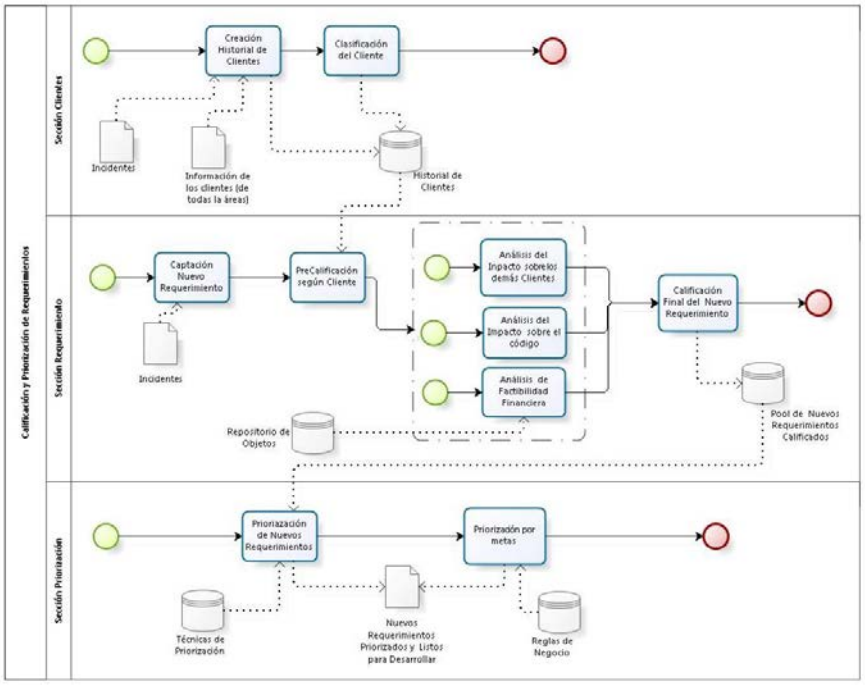


Fig. 1. Diagrama de procesos de la estrategia de priorización y selección de requerimientos.

3.1 Subproceso Sección Clientes

En los primeros intercambios de opinión se pudo constatar el alto nivel de dispersión de la información existente acerca de los usuarios y los requerimientos, tanto en la amplia variedad de soportes como en la distribución dentro de las empresas. Las tareas definidas previamente en el subproceso *Sección Clientes* son explotadas en múltiples

subtareas de recopilación, agrupación y normalización de datos relacionados con los *Clientes y los Requerimientos*, para luego buscar las técnicas más adecuadas para su unificación en el repositorio llamado *Historial de Clientes* [21] [22]. La Figura 2 muestra un ejemplo de modelo conceptual del *Historial de Clientes*, en él se observan las entidades relacionadas con la calificación de los Clientes desde el punto de vista de las diferentes áreas de la empresa y el seguimiento de cada requerimiento solicitado por el Cliente y los incidentes relacionados con el mismo. El seguimiento de los requerimientos continúa a nivel más detallado, involucrando los eventos ocurridos en su desarrollo, los cambios en los presupuestos y el impacto sobre los otros clientes. Las tareas del subproceso *Sección Clientes* toman toda la información dispersa en la empresa y mediante procesos de unificación de datos, se vuelca en *Historial de Clientes*, incluyendo calificaciones del Cliente según todas las áreas de la empresa que interactúan con él. Esta calificación se recoge con cuestionarios de valoración que pueden ser escritos, comunicaciones orales, o vía web, según necesidades de los responsables de la calificación.

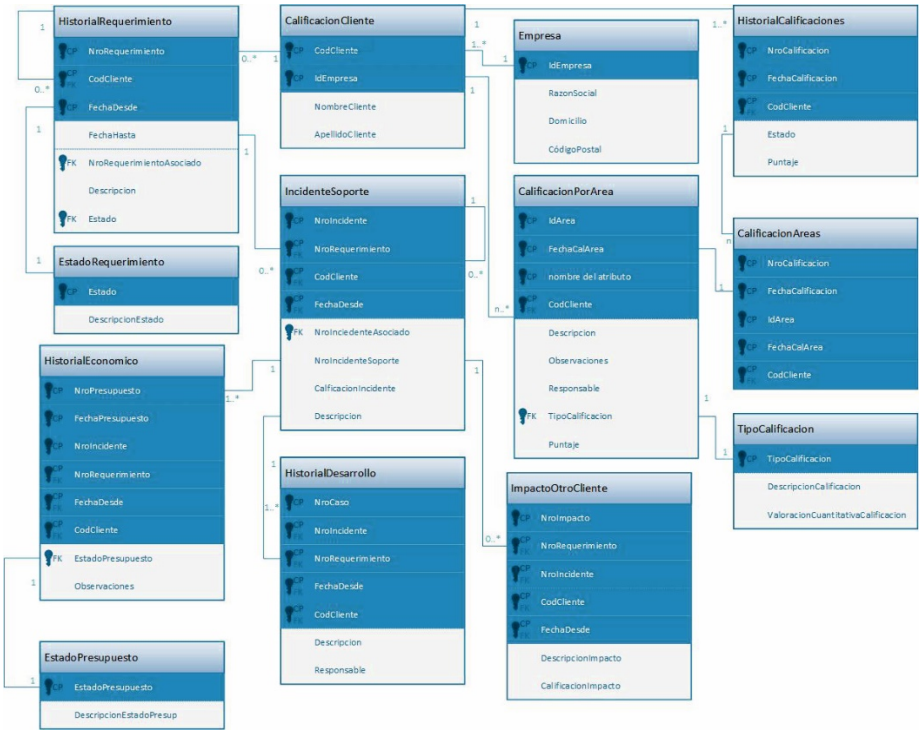


Fig. 2. Modelo conceptual del *Historial de Clientes*

3.2 Caso de Prueba en una Empresa Colaboradora

Se realiza un caso de prueba de la primera fase en una de las empresas colaboradoras. Esta empresa desarrolla un producto bien posicionado desde hace más de quince años

y cuenta con una enorme cartera de clientes. La cantidad de pedidos de nuevos requerimientos es muy alta, y las características son variadas, a lo que se le añade nuevas funcionalidades impositivas en forma constante. Los repositorios de datos de los nuevos requerimientos están dispersos en las áreas *Soporte, Desarrollo, Ayuda al Cliente, Contabilidad, Gerencia, Diseño.* Los repositorios están administrados por un gestor de desarrollo y un gestor de mesas de ayuda (aplicaciones diferentes), sistema de gestión contable, chat interno entre áreas, documentación anexa, registros financieros y de antiguos requerimientos en papel, etc. Los gestores brindan informes puntuales y generan vuelcos de datos en Excel y archivos de texto.

Los gerentes, jefes de áreas, supervisores y personal de alta jerarquía en cada área tienen experiencia de trato con los clientes y valoraciones personales. En este caso hay un enlace entre los pedidos de requerimientos y los proyectos, ya que se utilizan herramientas diferentes, que se realiza con planillas *Excel* desde el área soporte. Además ésta posee información importante sobre los clientes, por lo que se hace una extracción de los archivos de incidentes. Se procede a la interpretación y tipo de guardado en *Historial Clientes*. Los informes y vuelcos de datos se normalizan y se pasan por diferentes métodos, a repositorios intermedios para su migración a *Historial de Clientes*. Mucha información se encuentra en textos extensos que poseen un alto componente de valoración. Se toma la decisión de copiar los textos y agregar registros de resumen por cada incidencia. Las valoraciones gerenciales se toman directamente en cuestionarios con formato para migración al historial. Finalmente se generan registros para la calificación de los clientes y se llenan con consultas a la misma base. La Figura 3 muestra un esquema de ejecución del subproceso *Sección Clientes* con todos los pasos desde la recopilación de datos, las herramientas utilizadas y el personal involucrado.

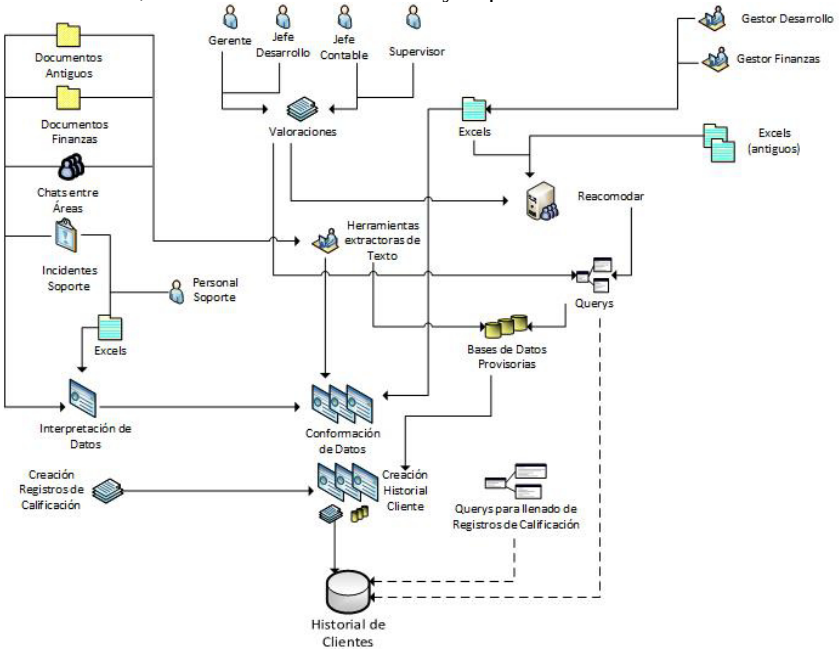


Fig. 3. Esquema de ejecución del subproceso *Sección Clientes*

Las tareas involucradas en el subproceso Sección Clientes como la creación del *Historial de Clientes* son *Customizados* a las características y distribución de la información que posee cada empresa, como así también a los objetivos de calificación a nivel gerencial.

3.3 Pasos Siguientes

Teniendo en cuenta las primeras experiencias recopiladas de las reuniones con los representantes de las empresas se seguirá avanzando sobre el proceso haciendo un intercambio permanente con las empresas y su aporte a la construcción de la estrategia. Los pasos siguientes irán en una línea similar a la de la primera explosión del subproceso Sección Clientes, con la generación de los subprocesos *Sección Requerimiento* y *Sección Priorización.* Luego se procederá a probar los mismos en empresas colaboradoras para su revisión, corrección y optimización.

4 Conclusiones e interrogantes

Las tareas realizadas en el marco de la investigación indican que la calificación, priorización y selección de nuevos requerimientos es en sí un sistema de alta complejidad que cruza transversalmente la mayoría de las áreas de una empresa desarrolladora de software. La construcción de una estrategia que sirva de herramienta para este proceso debe ser un trabajo conjunto entre el equipo de investigación y las empresas intervinientes. El entorno de trabajo debe incluir la presencia de representantes de las áreas, en similitud a los entornos ágiles de desarrollo.

Se observa que el primer paso de esta estrategia es un *proceso en sí mismo* que debe ejecutarse si se quisiera implementar una futura herramienta. La creación del repositorio *Historial de Clientes* da pie a la exploración de una aplicación de minería de datos en dicha herramienta. La amplia variedad de factores que intervienen en la toma de decisiones a la hora de hacer una priorización de requerimientos conlleva a una revisión permanente de las tareas, variables y métricas utilizadas.

La interacción permanente con las empresas trae aparejados ciertos riesgos que deberán analizarse y controlarse para no incluir desvíos al objetivo principal que es la creación de la estrategia. Además deberán tomarse medidas para que ninguna empresa en particular influya demasiado en las decisiones, generando una *especialización* del producto resultante de la investigación hacia sus necesidades concretas.

Referencias

1. Sommerville I.: Software engineering 9th ed. Addison Wesley, USA (2011)
2. Berander, P., Andrews A.: Requirements Prioritization. In: Aurum, A., Wohlin, C. (eds.) Engineering and Managing Software Requirements.1st ed. Springer Verlag (2005)
3. Bourque P., Farley R.: SWebok v3.0 Guide to the Software Engineering Book of Knowledge IEEE Computer Society Project (2014)

4. Canfora G., Cimitile A.: Software Maintenance. University of Sannio, Faculty of Engineering at Benevento Palazzo Bosco Lucarelli, Piazza Roma 82100, Benevento, Italy (2000)
5. Sher F., Jawawi D., Mohamad R., Babar M.: Multi-Aspects Based Requirements Priortization Technique for Value-Based Software Developments International Conference on Emerging Technologies ICET (2014)
6. Pohl K.: Requirements Engineering. Fundamentals, Principles, and Techniques. Springer (2010)
7. IEEE 12207-2008, Systems and software engineering -- Software life cycle processes, http://standards.ieee.org/findstds/standard/12207-2008.html
8. Erdil, K., Finn, E., Keating, K., Meattle, J., Park, S., Yoon, D.: Software Maintenance As Part of the Software Life Cycle, Department of Computer Science Tufts University (2003)
9. April A., Desharnais J: Software Maintenance Maturity Model (SMmm): A Software Maintenance Process Model, Department of Software Engineering École de technologie supérieure 1100 Notre-Dame West, Montréal, Québec, Canada (2006)
10. Fernández Núñez, L.**:** Cómo analizar datos cualitativos Butlletí La Recerca Ficha para investigadores**,** Ficha 7 Universitat de Barcelona (2006)
11. Garre M., Cuadrado J., Sicilia M.,Rodriguez D., Rejas, R.: Comparación de diferentes algoritmos de clustering en la estimación de coste en el desarrollo de software. Revista Española de Innovación, Calidad e Ingeniería del Software 3:1 (2007)
12. De Federico S., Sincosky N., Lascano A., Avogradini M., Moschetti D,: Análisis para la identificación de clusters en información recopilada de empresas de desarrollo de software sobre técnicas de priorización de requerimientos. Actas del 2° Congreso Nacional de Ingeniería Informática/Sistemas de Información 2ª CoNaIISI, pp. 1171-1176 (2014)
13. De Federico, S., Sincosky N., Avogradini M., Moschetti D.: Construcción de una Metodología para la Priorización y Selección de Nuevos Requerimientos a Implementar en Software en Etapa de Mantenimiento. Workshop de Investigadores en Ciencias de la Computación WICC, Salta (2015)
14. Achimugu P., Selamat A., Ibrahim R., Mahrin M.: A systematic literature review of software requirements prioritization research. Information and Software Technology 56:6, 568-585 (2014)
15. Saaty T. Relative Measurement and Its Generalization in Decision Making Why Pairwise Comparisons are Central in Mathematics for the Measurement of Intangible Factors. The Analytic Hierarchy/Network Process. RACSAM Rev. R. Acad. Cien. Serie A. Mat. 102:2, 251–318 (2008)
16. Matzler K, Hinterhuber H. How to make product development projects more successful by integrating Kano's model of customer satisfaction into quality function deployment. Technovation 18:1, 25–38 (1998)
17. Hauser, J., Clausing, D.: The House of Quality Matrix, Harvard Business Review (1988)
18. Aksyonov K., Spitsina I., Aksyonova O., Schaible P.: The state analysis of intelligent tools of Computer Aided Software Engineering. 24th International Crimean Conference of Microwave & Telecommunication Technology CriMiCo (2014)
19. Bizagi, Business Process Management Software, http://www.bizagi.com/es/
20. Okawa T., Kaminishi T., Hirabayashi S., Koizumi H., Sawamoto J.: An Information System Development Method Based on the Link of Business Process Modeling with Executable UML Modeling and its Evaluation by Prototyping. 22nd International Conference on Advanced Information Networking and Applications (2008)
21. Visalakshi N., Thangavel K., Alagambigai P. Distributed Clustering for Data Sources with Diverse Schema. Third International Conference on Convergence and Hybrid Information Technology (2008)
22. Asghar S., Rukhsana, Subhan S., Fong S.: Assimilation of Heterogeneous Resources by Utilizing a Unified Format of XML for Constructing OLAP. Fourth International Conference on Cubes Computer Sciences and Convergence Information Technology (2009)

SIGC - Herramienta para la administracion de SGC certificado bajo ISO 9001

Ariel Pasini[1], Silvia Esponda[1], Delfina Diaz[1], Julieta Calabrese[1], Maximiliano Swaels[1,] Marcos Boracchia[1,] Patricia Pesado[1,2]

[(1)]Instituto de Investigación en Informática LIDI (III-LIDI)
Facultad de Informática – Universidad Nacional de La Plata
50 y 120 La Plata Buenos Aires
[(2)] Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CIC)
526 e/ 10 y 11 La Plata Buenos Aires
{apasini,sesponda,ddiaz,jcalabrese,mswaels,
marcosb,ppesado}@lidi.info.unlp.edu.ar

Abstract. En la actualidad la gestión de la calidad es una parte fundamental dentro de cualquier organización al momento de prestar un mejor servicio o tener un producto que cumpla las exigencias del cliente.
La Norma ISO 9001 "Sistemas de gestión de la calidad. Requisitos" se centra en la eficacia del sistema de gestión para dar cumplimiento a los requisitos del cliente.
La implantación y constante mantenimiento de un Sistema de Gestión de la Calidad (SGC) en una organización, es una tarea que requiere de mucho esfuerzo, atención y uso de diferentes herramientas para agilizar el trabajo.
Este artículo propone SIGC (Sistema Integral de Gestión de la Calidad), una herramienta integrada para el análisis, generación y mantenimiento de un SGC, haciendo foco en: *el Relevamiento inicial, la Generación de documentos, la Mejora continua, los Permisos y roles, el Envío automático de notificaciones* y *un Calendario integrado.*

Keywords: ISO 9001- Sistema de gestión de la Calidad- Gestión de procesos – PDCA.

1 Introducción

En la actualidad la gestión de la calidad es una parte fundamental dentro de cualquier organización ya que ésta puede llegar a ser una ventaja competitiva que fortalezca a la organización en el momento de prestar un mejor servicio o tener un producto que cumpla las exigencias del cliente [1].

La Norma ISO 9001:2008 "Sistemas de gestión de la calidad. Requisitos" se centra en la eficacia del sistema de gestión para dar cumplimiento a los requisitos del cliente [2]

Se encuentra en preparación la versión 2015 que se espera sea aprobada por ISO en octubre - noviembre del corriente año, luego, IRAM deberá aprobar la traducción

oficial, por lo que seguramente recién esté disponible para principios del próximo año en nuestro país. Según informó IRAM, la transición entre la versión 2008 y 2015 abarcará un periodo de 3 años de convivencia de ambas versiones. El presente trabajo se basa en la versión 2008. Hasta el momento, por el conocimiento público de la nueva versión, se prevé que no acarreará grandes cambios.

La norma promueve la adopción de un enfoque basado en procesos.

- Una organización funciona de manera eficaz, gestionando de manera eficiente un gran número de actividades relacionadas entre sí.
- Una actividad o un conjunto de actividades que utiliza recursos, y que se gestiona con el fin de permitir que los elementos de entrada se transformen en resultados, se considera un proceso.
- La aplicación de un sistema de procesos dentro de la organización, se denomina "enfoque basado en procesos".
- La principal ventaja del enfoque basado en procesos es el control continuo que proporciona sobre los vínculos entre los procesos individuales dentro del sistema como sobre su combinación e interacción.

En la siguiente sección se describe el proceso desde la concepción hasta la mejora continua de un SGC en una organización, basada en el punto de vista de ciclo de Deming (Planificar, Hacer, Verificar y Actuar). En la tercera sección se presentan las herramientas que existen hoy en día en el mercado y sus falencias. A continuación se describe la herramienta propuesta para la concepción de un SGC desde el punto de vista planteado. Por último se discutirán las conclusiones y trabajos futuros.

2 Sistema de Gestion de la Calidad

2.1 Iniciar el camino hacia la Gestión de Calidad

Una organización que se propone aplicar e implementar un SGC bajo la norma ISO 9001 se enfrenta con diferentes barreras. La visualización del punto de partida es crítico para un alto porcentaje de las mismas, sobre todo si partimos de la base que no cuentan con personal capacitado para su ejecución y los costos relacionados a consultorías.

Como base inicial se tiene que lograr el compromiso de la máxima autoridad de la organización - *alta dirección*- . Ella conforma un rol fundamental en el momento de establecer el alcance, determinar la política y los objetivos, y proveer los recursos necesarios. También es la que comunica y hace que toda la organización esté comprometida con la calidad [2].

La definición de un proceso para realizar un Relevamiento de Situación Inicial (RSI) ayudaría a los involucrados a obtener un diagnóstico inicial del estado de la organización respecto a los requisitos de la norma, y además establecería los distintos

pasos para alcanzar, de manera factible, cada uno de los elementos impuestos por la norma.

Para lograr un efectivo RSI el proceso que desarrolle la organización debería contar inicialmente con los siguientes pasos:

- **Paso 0:** Establecer el equipo de trabajo
- **Paso 1:** Determinar y/o revisar el alcance del SGC

Este deberá contener una definición clara de los procesos y unidades que implantarán el SGC. El alcance descripto de manera precisa permite definir qué actividades se realizarán, los tiempos, gastos, los requisitos de la norma que se incluirán y cuáles pueden ser excluidos.

Para facilitar este paso se sugiere: a) Identificar productos y/o servicios controlados por el SGC. b) Establecer los procesos dentro del SGC. c) Definir exclusiones. d) Centrarse en la naturaleza de los productos y/o servicios y sus procesos de realización, el resultado esperado, los riesgos y consideraciones específicas de cada caso.

- **Paso 2:** Identificar y/o revisar los procesos que son necesarios para elaborar y gestionar el producto o servicio.

Los procesos se clasifican en tres familias: a) Los **procesos de realización:** contribuyen directamente a la realización del producto, desde la detección de la necesidad del cliente a su satisfacción. b) Los **procesos de soporte:** contribuyen al buen desarrollo de los *procesos de realización* aportándoles los recursos necesarios. c) Los **procesos de gestión:** contribuyen a la determinación de la política y al desarrollo de los objetivos en la organización. Bajo la responsabilidad total del equipo dirigente, ellos permiten orientar y asegurar la coherencia de los procesos de realización y de soporte.

- **Paso 3:** Determinar la presencia y/o revisar el mapa de procesos del sistema: Los mapas de procesos son una técnica de gran utilidad para la evaluación de los procesos, mostrando cómo sus actividades están relacionadas con los clientes externos, proveedores y grupos de interés. En el mapa de procesos se muestran de manera gráfica todos los procesos identificados y la interrelación entre cada uno de ellos.
- **Paso 4:** Determinar la presencia y/o revisar las funciones y responsabilidades. Cada uno de los involucrados en el SGC debe contar con una definición de su perfil, para establecer nivel educacional, formación, experiencia y habilidades de cada uno. Además se debe identificar tareas, funciones y responsabilidades.
- **Paso 5:** Determinar la presencia y/o revisar la documentación donde se establezca la forma de comunicación interna.
- **Paso 6:** Determinar la presencia y/o revisar la política de calidad. En ella se establece el compromiso de todo el personal ante los requisitos de los clientes y la mejora continua de la eficacia del SGC. Es una declaración pública y documental del compromiso que asume la Dirección.
- **Paso 7:** Determinar la presencia y/o revisar los objetivos de calidad. Se presenta el compromiso de la organización con respecto a la calidad. Este complementa la política de calidad de manera cuantificada.

El orden de los pasos a seguir no es casual sino que se definen de manera que al grupo involucrado le sea más simple ya que la información documentada en revisión está relacionada con la evaluada en el paso anterior.

Una vez logrado el diagnóstico de situación, los autores entienden que la forma óptima para la gestión de la calidad, es seguir el Ciclo de Deming Planificar – ***Hacer*** - ***Verificar*** – ***Actuar*** [3].

- Planificar: establecer los objetivos y procesos necesarios para conseguir resultados de acuerdo con los requisitos del cliente y las políticas de la organización.
- Hacer: implementar los Procesos.
- Verificar: realizar el seguimiento y la medición de los procesos y los productos respecto a las políticas, los objetivos y los requisitos para el producto, e informar sobre los resultados.
- Actuar: tomar acciones para mejorar continuamente el desempeño de los procesos.

2.2 *Planificar* - Planificación de un SGC bajo ISO 9001

El hecho de contar con un proceso previo a la Planificación, permite a la organización visualizar claramente qué le falta para lograr alcanzar la certificación.

La ISO 9001 determina requisitos obligatorios para demostrar su capacidad de proporcionar productos o servicios que satisfagan los requisitos del cliente.

Dichos requisitos involucran la generación de documentación que evidencie la realización de actividades establecidas en los procesos y procedimientos que formalicen las acciones a seguir.

La documentación surge a medida que el SGC se ejecuta, pero los procedimientos documentados establecidos como obligatorios por la norma deben estar definidos con anticipación.

En función a la información obtenida por el RSI la organización está en condiciones de planificar la generación/actualización de los procesos, procedimientos, documentos, etc. en función de los requisitos de la norma.

2.3 *Hacer* - Puesta en marcha del SGC

Una vez definido y documentado el SGC, se pone en marcha según lo planificado.

La implementación consiste en la ejecución por parte de cada uno de los involucrados, de las actividades establecidas en los procesos que conforman el SGC. Esta es una fase crítica en la que se verifican la aceptación y el compromiso en el proceso de cada uno de los participantes, y las necesidades de capacitación específica.

Esta es una ardua labor, de la cual surgen los registros que evidencian dicha ejecución.

Pero además se deben elaborar los registros de los requisitos obligatorios determinados por la norma, tales como Auditorías o Revisión por la Dirección.

Todas y cada una de estas tareas deben ser realizadas en los tiempos que corresponden, determinados en los procedimientos documentados. No es simple el mantener todo el SGC en funcionamiento ante cada una de estas actividades.

Suele suceder que, ante la práctica de una auditoría, surgen diferentes *No conformidades* u *Observaciones*, por no contar con la evidencia de alguna de las actividades, y sólo por el olvido.

2.4 *Verificar* - Evaluación y mejora

En esta etapa se recopilan los datos y se realizan las mediciones, en el marco de auditorías internas y revisiones por parte de los responsables del SGC.

La evaluación se pone en marcha una vez que el sistema se encuentra en funcionamiento.

Las auditorías internas y la revisión del SGC ofician de autoevaluación. En el caso de desvíos, se analizan las causas y se implementan acciones para corregirlos.

El SGC debe mejorarse de manera continua, para lo cual debe analizarse el grado de satisfacción del cliente.

Para lograr la satisfacción del cliente es preciso capturar su percepción de la forma más precisa. Existen diferentes métodos para medir la satisfacción del cliente, la elección del más adecuado depende del conjunto y cantidad de usuarios a evaluar, el medio en que interactúa con el sistema, etc [4]. La primera clasificación de los métodos se relaciona con la forma de obtener los datos: a) ***Obtención de datos primarios:*** implica recolectar la información mediante el uso de alguna fuente para poder así operar con los mismos. Entre estas fuentes se pueden destacar los cuestionarios, la observación, entrevistas, entre otras. b) ***Obtención de datos secundarios:*** la información ya fue reunida previamente y es utilizada para un fin en particular, principalmente para obtener reportes estadísticos. A modo de ejemplo, se puede utilizar información recolectada en un censo, bases de datos de alumnos de una institución, etc. Al contar con la información ya recolectada, se realizan distintas operaciones con los datos y se obtiene un resultado, el cual puede ser de gran utilidad para medir la satisfacción de un cliente en los aspectos que se crean necesarios. Por ejemplo el nivel de uso de nuevas funcionalidades que simplifican la tarea, etc.

Dentro de la ***Obtención de datos primarios***, existen dos tipos de técnicas para obtener la información, denominadas *cualitativas* y *cuantitativas*. Las técnicas cualitativas apuntan a recolectar mucha información basándose en una muestra pequeña de la totalidad de la población. Caso contrario ocurre con las cuantitativas, las cuales obtienen poca información relevante pero de una muestra más grande, teniendo así una mayor cantidad de casos a evaluar. A continuación se enumeran las técnicas *cualitativas* más utilizadas. ***Grupos de discusión:*** Los grupos de discusión se basan en la obtención de información reuniendo a un grupo de personas, invitados puntualmente para armar un debate. De estos grupos se rescatan muchos datos valiosos utilizados para obtener una retroalimentación del sistema brindado. ***Entrevistas en profundidad:*** Este método de obtención es uno de los más utilizados hoy en día. Las entrevistas son un medio en el cual se comunican el dueño del producto con uno o más clientes. El objetivo es explorar en profundidad aspectos

relacionados con el sistema brindado, ya sea recibiendo sugerencias, opiniones, realizar preguntas puntuales, entre otros. ***Panel de usuario:*** El panel es otra de las herramientas, encargada de reunir a un grupo de usuarios experimentados en el sistema y obtener periódicamente una retroalimentación a partir de la experiencia de cada uno de ellos. Como desventaja, se puede mencionar que se descartan las opiniones de los nuevos usuarios y por ende, la "primera impresión" del mismo. ***Buzón de sugerencias:*** El buzón de sugerencias es una herramienta que le permite al usuario expresar su disconformidad, sugerencia o queja, sin necesidad de interactuar con un tercero. Es una opción inmediata con la que cuenta a la hora de expresar alguno de los casos mencionados anteriormente. Por el lado de las *cuantitativas* tenemos la ***Encuesta,*** que es la técnica más utilizada a la hora de medir la satisfacción del cliente. Cabe destacar que suelen aplicarse en una población grande, obteniendo así información útil principalmente para la realización de estadísticas. Es importante que las mismas se realicen de forma regular y periódica. En el caso de que los resultados no sean los esperados, el proveedor del sistema deberá ejecutar algún plan de acción de mejora y realizar una nueva encuesta. Por último se encuentra la ***Observación***, considerada una técnica mixta entre las *cualitativas* y las *cuantitativas*. Esta fuente de información suele ser utilizada cuando se quiere aprender acerca de la interacción de un grupo, el comportamiento de un individuo, o un fenómeno que se lleva a cabo, entre otros. Brinda información que otros métodos no pueden recolectar, por ser características que se observan y no que se dicen.

Lograr obtener de forma precisa el nivel de satisfacción del usuario, sus comentarios, sugerencias y el análisis cualitativo y cuantitativo de esa información es una componente esencial para encontrar las deficiencias del SGC e incorporarlas como futuras mejoras al sistema que nos permitirá permanecer dentro del ciclo de la mejora continua.

2.5 *Actuar* - Toma de decisiones en SGC

El proceso de toma de decisión debe nutrirse de las fuentes más confiables, como resultados de auditorías, revisiones por la dirección, diferentes tipos de evaluaciones realizadas a los clientes, etc. Las decisiones tomadas en esta etapa se planificarán para su ejecución.

3 Herramientas

En la actualidad existen varios programas diseñados para la administración de documentos, algunos de ellos exclusivamente para la industria IT.

Además, existen distintos software para SGC que administran su documentación de manera ágil, pero no abarcan todas las actividades establecidas para la creación, ejecución y mantenimiento del SGC.

Antes de la aparición de estas herramientas, la forma de administrar datos e información ha sido mediante planillas de cálculo y editores de texto. Sin embargo, la incorporación creciente de los sistemas de calidad e implementación de un manual de

gestión de la calidad, ha llevado a que la cantidad de documentación controlada y la forma en que se debe gestionar la información, hagan de estos programas muy poco amigables. La mayor cantidad de proyectos importantes y de gran envergadura junto a la mayor comunicación digital con los distintos actores en un proyecto de IT, ha determinado por incentivar la búsqueda de nuevas herramientas open source para el control de la información.

Se realizó una evaluación de las principales herramientas existentes, determinando lo que, para nuestra experiencia en gestión del SGC deberían contener.

Relevamiento de Situación Inicial: establecer lo que cuenta o le falta a la organización, respecto a los requerimientos obligatorios de la norma ISO 9001.

Generación de Documentos ISO 9001: permitir la generación de documentos que cumplan con los requerimientos de la norma ISO 9001 de manera óptima, realizando un manejo automático del versionado de los documentos, llevando un control del estado del mismo, ya sea elaborado revisado y/o aprobado, y los respectivos usuarios que llevaron a cabo dichas acciones.

Mejora Continua PDCA: permitir la mejora continua de la organización a través de la realización del PDCA (Plan, Do, Check, Act) realizando un seguimiento y control de las no conformidades, observaciones, oportunidades de mejora, acciones correctivas y preventivas, y de los indicadores.

Roles y Permisos: permitir el manejo de diferentes usuarios en el sistema y controlar sus respectivos permisos dentro del mismo. Especialmente se deberán controlar los permisos de edición, revisión y aprobación de los documentos.

Envío Automático de Notificaciones: permitir el envío automático de notificaciones a usuarios ya sea por eventos próximos, por vencimientos de plazos de no conformidades o por solicitudes de revisión o aprobación.

Calendario Integrado: permitir el uso de un calendario que integre todas las tareas a llevar a cabo permitiendo que el usuario pueda visualizarlo más fácilmente y enfocarse en las actividades que debe realizar en el día/semana/mes correspondiente.

De acuerdo a lo ya establecido se evaluaron ocho herramientas orientadas a la gestión de SGC.

El costo de las herramientas depende por lo general de la cantidad de usuarios que usen la herramienta, lo que las hace más difíciles de adquirir en organizaciones denominadas PyMES.

El resultado de dicha evaluación se muestra en la tabla 1.

Herramienta		Relevamiento de Situación Inicial	Generación Documentos ISO 9001	Mejora Continua PDCA	Roles y Permisos	Envío Automático de Notificaciones	Calendario Integrado
KMKey Quality	**[5]**	**X**	**A**	**A**	**A**	**P**	**X**
ISOlución	**[6]**	**X**	**A**	**A**	**A**	**P**	**P**
ISOXPRESS	**[7]**	**X**	**A**	**P**	**A**	**P**	**X**
QUTASK	[8]	**X**	**A**	**P**	**A**	**A**	**X**
ZenDoc Quality. Simplified.	**[9]**	**X**	**A**	**P**	**A**	**A**	**X**
FlinkISO Open Source QMS	**[10]**	**P**	**A**	**P**	**P**	**A**	**X**
bemus Plataforma de Gestión de Sistemas	**[11]**	**X**	**P**	**A**	**A**	**A**	**P**
VIVALDI SOFTWARE SOFTWARE THAT HELPS YOU IMPROVE	**[12]**	**X**	**A**	**P**	**A**	**A**	**X**

Table 1. Comparacion de Herramientas para SGC

A= Alcanzado**, P=** Pobremente Alcanzado, **X=** No Alcanzado

Se ha observado que ninguna de las herramientas brinda la posibilidad de realizar el Relevamiento de Situación Inicial excepto por FlinkIso, la cual ofrece informes con formato checklist pero no con el objetivo buscado [10].

La generación de documentos ISO 9001 es alcanzada satisfactoriamente por todas las herramientas salvo por Bemus que no controla el versionado de los documentos [11].

En cuanto a la mejora continua, gran parte de las herramientas lo alcanzan pobremente pues la gran mayoría no brinda la posibilidad de administrar los indicadores.

El control de roles y permisos es manejado satisfactoriamente en todas las herramientas con excepción de FlinkIso, mientras que el envío automático de notificaciones es también alcanzado por gran parte de las herramientas con excepción

de aquellas que no lo poseen automatizado totalmente o sólo se limitan a notificar al usuario dentro del sistema.

Finalmente, el calendario integrado sólo es parcialmente alcanzado por dos herramientas que si bien permiten el uso de agenda y cronograma no brindan la posibilidad de un calendario integrado que cumpla con lo esperado, mientras que las demás no logran alcanzarlo.

3.1 SIGC- Propuesta integradora para la administración de un SGC

Basado en el enfoque del PDCA se propone SIGC (Sistema Integral de Gestión de la Calidad), una herramienta integrada que facilite la puesta en marcha y ejecución de un SGC. La herramienta, cuenta con 5 módulos que interactúan sólo mediante la información que se crea o actualiza: Relevamiento de Situación Inicial (módulo cuya definición está basada en el proceso presentado en 2.1) - Planificación - Ejecución – Evaluación y mejora – Toma de decisión.

Luego, en base al principio de PDCA, SIGC se utiliza para la generación y mantenimiento del SGC. Cuando toda la información de carácter obligatorio (del SGC y propia del proceso) se encuentra aprobada, el sistema se considera en ejecución, repitiendo el ciclo para el mantenimiento del mismo.

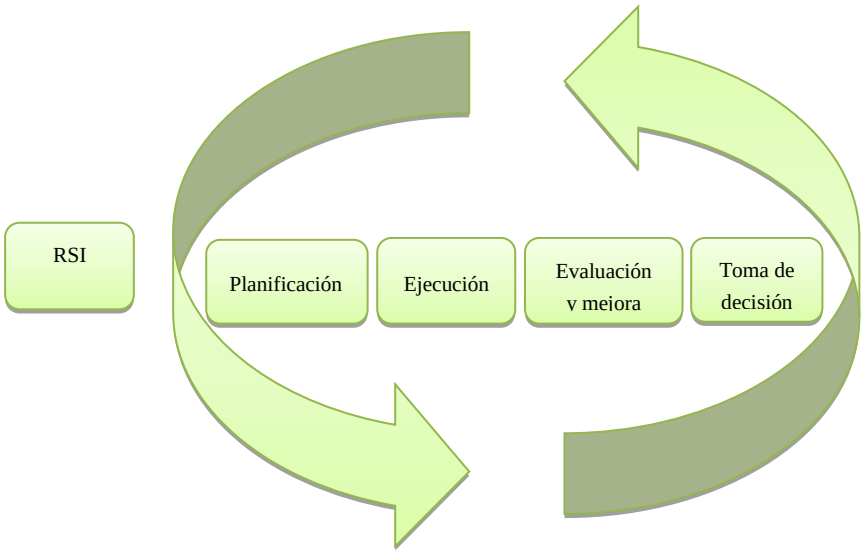


Fig. 1. SIGC

Para la visión integral del sistema, las acciones se deben dividir en dos etapas:

- Inicio y generación del SGC.
- Ejecución y Mantenimiento del SGC.

En la tabla 2 se presentan las principales funcionalidades de cada modulo

Módulo	
RSI	El usuario completa un Checklist de Documentación obligatoria, donde se logra establecer la situación de la organización.
Planificación	Generación /Actualización de: --Calendario --Alcance --Documentación obligatoria (Política, Objetivos, Procedimientos, etc) --Proceso --Perfiles y responsabilidades - Proveedores
Ejecución	Ejecución de cada uno de los pasos definidos en el proceso. Generación de Registros para establecer Evidencia.
Evaluación y mejora	Ejecución de métodos de medición: --Técnicas cualitativas: -Grupos de discusión -Entrevistas en profundidad -Panel de Usuario -Buzón de sugerencias --Técnicas cualitativas: (Encuestas Técnicas mixtas)
Toma de Decisión	Evaluación de la información recolectada: --Encuestas --Evaluaciones --Revisiones por la Dirección --Auditorías Definición de Obs./ NC /NCP / Mejoras

Table 2. Descripcion de los módulos - SIGC

RSI- Diagnosticar

Este módulo se creó tomando como base el proceso definido en 2.1. Sólo se realiza al inicio y generación del SGC. Está dirigido a la autoridad del grupo de Calidad. Tiene como objetivo ayudar a la toma de conciencia respecto al lugar en el cual se encuentra la organización respecto a toda la información documentada obligatoria establecida por la norma.

Planificación - Planning

Este módulo crea o actualiza la información dependiendo de la iteración que se esté ejecutando. En el caso de inicio de SGC, depende del resultado del módulo anterior.

En este módulo, se trabaja el alcance del SGC, las exclusiones, con la fundamentación que justifique la exclusión, los procesos, etc.

Para cada información documentada o proceso establecido se proporciona la opción de realizar un vínculo con un documento existente o por crear. En el caso que no exista, automáticamente se generará una entrada en la sección de control de documento para que se lo codifique de acuerdo a la forma que la organización lo haya establecido en el proc. De Control de Documentos.

Cada módulo puede tener asociado un gráfico describiendo su funcionamiento.

Además, cada documentación generada, posee automáticamente una entrada en el subsistema de control de documentos.

Toda la información registrada por el sistema permite definir el manual de la calidad, para lo cual el sistema, si es el inicio del SGC, presenta una estructura base que contiene todos los puntos requeridos.

Cada documento generado o actualizado posee un estado que puede ser: elaborado, revisado o aprobado. El estado de los documentos será utilizado por el subsistema de evaluación para medir el estado general del SGC.

Al momento de la creación de cada documento se permite realizar un guardado provisorio y uno definitivo, en caso de seleccionar el provisorio se tendrá en cuenta que pasado un determinado tiempo sin volver a ser editado se le enviará una notificación al autor del mismo como recordatorio de que el documento debe ser completado.

Cuando se concluye con la creación y/o modificación de información documentada, se actualiza el calendario del SGC, a través del cual se envía las notificaciones al responsable de cada tarea.

El usuario podrá acceder a los últimos documentos creados o modificados

Ejecución - Do

Durante la ejecución el sistema da cumplimiento a cada uno de los pasos determinados en el proceso específico definido. Esto da como resultado los Registros, necesarios para lograr la evidencia requerida por la norma. Cada vez que se cree o se actualice un registro automáticamente se generará o se actualizará una entrada en la sección de control de registro para que se lo codifique según corresponda.

Evaluación y mejora -Check.

El módulo de evaluación permite generar los diferentes tipos de herramientas de evaluación, por ejemplo encuestas, para poder obtener el nivel de satisfacción del cliente.

Una vez que el SGC incluya información histórica, la medición de la satisfacción del cliente se hará mediante técnicas más precisas. Por ejemplo, se podrán evaluar los indicadores exportando un gráfico que compare los valores obtenidos con los valores deseados. De esta manera se facilita el análisis de los resultados obtenidos. Así mismo, se podrán evaluar las no conformidades generadas exportando un gráfico que visualice los porcentajes de los diferentes orígenes de las mismas. Así, se podrá identificar fácilmente en donde se encuentran las debilidades.

Toma de decisiones en SGC- Act.

El representante de la dirección debe contar con un panel de control donde pueda ver a simple vista la última auditoria, la última revisión por la dirección, el resultado de las evaluaciones realizadas a los clientes, el estado de las No conformidades pendientes y el resultado de los indicadores.

Analizada la información presentada, y en base a la revisión por la dirección se registran las No Conformidades y Observaciones que considere pertinentes en cada caso.

Así mismo, en base a los resultados obtenidos en los indicadores se decidirá qué nuevos objetivos se van a establecer y cuáles serán modificados o cerrados.

Luego el módulo de planificación tomará esta información como entrada y permitirá planificar el tratamiento de la misma.

4 Conclusiones

Se presentó la problemática de la implementación de Sistemas de gestión de la Calidad bajo las normas ISO 9001. Si bien ISO liberaría la nueva versión de la misma en los próximos meses, se continúa trabajando con la versión vigente ya que las propuestas realizadas se asumen como válidas para la nueva versión.

El principal problema de los SGC es tanto la generación y actualización de la documentación requerida, como el control de lo planificado durante la ejecución.

Se evaluaron herramientas disponibles en el mercado bajo cinco criterios que, según la experiencia del grupo, resultan críticos en el SGC.

Se concluyó que ninguna de las herramientas analizadas cumplía con esos cinco criterios. En función de eso se decidió desarrollar SIGC, una herramienta, con cinco módulos, donde el primero se utiliza para conocer el estado de la organización a certificar, y los cuatro siguiente siguen los pasos de PDCA.

SIGC se encuentra en fase de prototipación y está siendo utilizada para la administración del SGC del Pre Ingreso de la Facultad de Informática.

5 Referencias.

1. Piatini Mario. et all. “Calidad de Sistemas de Información”. 2da Ed. Actualizada. Ediciones Ra-Ma. 2011.
2. ISO 9001:2008 Sistemas de gestión de la calidad. Requisitos.
3. Deming, W. Edwards, “Calidad, Productividad y Competitividad: La salida de la crisis”. Ediciones Díaz de Santos, 1989
4. Kumar Ranjit. “ Research methodology a step-by-step guide for beginners”. Editorial SAGE PUBNS INC (CA). 2005.
5. KMKEY Quality Software para la gestión de la calidad y proyecto. http://www.kmkey.com/ Agosto 2015.

6. ISOlucion Sistemas integrado de Gestión. http://www.isolucion.com.co/ Agosto 2015.
7. ISOPXPress ISO 9001 Quality Management http://www.isoxp.com/00Home/IsoXpHome.aspx Agosto 2015.
8. QUTASK http://qutask.com/es/ Agosto 2015.
9. ZenDoc Quality Management Software http://www.getzendoc.com/ Agosto 2015
10. FlinkISO Quality Management Software for ISO 9001 http://www.flinkiso.com/ Agosto 2015
11. bemus Plataforma de Gestión de Sistemas. http://www.bemus.eu/ Agosto 2015
12. VIVALDI SOFTWARE Software that helps you improve. http://www.vivaldisoftware.com/ Agosto 2015

Umbrales sugeridos para promedios de métricas de diseño de una aplicación en Java

Jorge Ramirez, Gustavo Gil, Carina Reyes

Universidad Nacional de Salta, Facultad de Ciencias Exactas, Centro de Investigación y Desarrollo en Informática Aplicada (CIDIA)
{ramirezj, gdgil,reyescarina}@cidia.unsa.edu.ar

Resumen. Las métricas de producto de software pueden aportar información relevante para la gestión de proyectos de software. La determinación de umbrales para esas métricas permitiría disponer de información rápida que advierta sobre potenciales problemas en los productos bajo análisis. Sin embargo, no existe amplio consenso en cuanto a estos indicadores. El Software Libre y de Código Abierto (F/OSS por sus siglas en inglés) ofrece la posibilidad de estudiar grandes cantidades de versiones a partir de la libre disponibilidad del código fuente que caracteriza a las licencias bajo las cuales se distribuyen esos productos. Este trabajo propone umbrales para métricas a partir del estudio de las distribuciones de frecuencia del promedio de la cantidad de métodos por clase y de la proporción de referencias a métodos de otras clases respecto del total de métodos definidos en la aplicación.

Palabras Clave: Métricas de Diseño, Software Libre y de Codigo Abierto. Umbrales para métricas.

1 Introducción

La caracterización de una pieza de software con miras a evaluar su potencial adopción o reutilización, requiere de disponer de elementos de juicio suficientes para la toma de decisiones. En particular, al momento de seleccionar un producto para incorporarlo a un nuevo desarrollo, es necesario contar con indicadores que orienten esta determinación.

No existe consenso general en la ingeniería de Software acerca de la vinculación entre métricas y los distintos atributos del software [1]; un ejemplo en este sentido es el estudio empírico presentado por Sjoberg y otros [2] que revela que varias métricas asociadas habitualmente a la mantenibilidad (el índice de mantenibilidad, distintas medidas de tamaño y cohesión) son inconsistentes entre sí. Fowler [3] considera que ninguna combinación de métricas puede rivalizar con la intuición humana informada; compartiendo en general ese punto de vista, no se busca postular aquí relaciones causales sino indicadores que actúen como advertencia de posibles debilidades en el diseño. Lanza y Marinescu [4], en tanto, señalaron que no existe un umbral perfecto, cuya superación signifique necesariamente un cambio cualitativo en el atributo que se pretende evaluar, pero pueden ser indicativos de utilidad en la práctica.

Desde esta perspectiva se aborda la posible determinación de umbrales de métricas de software, es decir, valores que al superarse alerten sobre posibles debilidades en el diseño o sobre riesgo potencial respecto de la mantenibilidad de código.

Para ello en este trabajo se analizan las distribuciones de frecuencias de un conjunto de métricas sobre una muestra de versiones de aplicaciones F/OSS escritas en Java pertenecientes a proyectos que han publicado 7 versiones o más, lo que refleja la evolución de estos proyectos.

El enfoque adoptado aquí se diferencia del que sigue la gran mayoría de los estudios que proponen umbrales, dado que se buscan elementos para evaluar el diseño en su conjunto mediante valores promedio en lugar de determinar valores que sirvan de referencia para módulos o clases particulares.

En concreto, se plantea detectar qué métricas referidas al diseño de una aplicación muestran una distribución de frecuencias tal que sugiera la conveniencia de adoptar determinados umbrales. El trabajo se organiza de la siguiente manera: en el apartado siguiente se ofrece una revisión de trabajos similares; a continuación, se describe la metodología utilizada en la investigación; posteriormente se analizan los datos obtenidos y se proponen umbrales para algunas métricas; a continuación, se comparan estos valores con los sugeridos en otros trabajos. Finalmente, en la sección 5 se exponen las conclusiones y los trabajos futuros.

2 Trabajos Relacionados

Se han propuesto diversos umbrales para métricas de diseño orientado a objetos definidas mediante diversos métodos. Por un lado, algunos trabajos utilizan técnicas estadísticas que intentan relacionar las mediciones obtenidas con atributos asociados a la mantenibilidad o a la propensión a errores de alguna parte de código o un elemento de diseño [5] [6] [7] [8] . Otros, en cambio, se basan en la distribución de los valores de las métricas, buscando revelar cuáles serían cifras atípicas para los indicadores que consideran [9] [10] [11] [4] [12] .

En la mayoría de los estudios las métricas que se abordan son las propuestas por Chidamber y Kemerer [13], y Brito e Abreu y Carapuça [14] entre otras de uso común.

2.1 Estudio de Umbrales en Base a Correlaciones Estadísticas

Los trabajos que se resumen a continuación apuntan a determinar valores de las métricas que pudieran advertir sobre debilidades en un módulo o una clase en concreto.

Benlarbi y otros [5] plantearon la determinación de umbrales para los valores de las métricas orientadas a objetos. En el estudio consideran a los umbrales como valores heurísticos que sirven para determinar rangos de valores deseables y no deseables para métricas de software. Los autores analizaron las métricas propuestas por Chidamber y Kemerer sobre dos sistemas de telecomunicaciones escritos en C++ en los que clasificaron a cada clase como “defectuosa” o “no defectuosa” según si se había reportado al menos una falla durante la operación del sistema.

Los autores concluyeron que el establecimiento de umbrales no aporta ventajas respecto de un modelo continuo.

Shatnawi realizó diversos estudios buscando determinar umbrales a partir de diferentes análisis estadísticos. En uno de ellos [6], emplea la curva ROC para proponer umbrales de un conjunto de métricas CK y de Lorentz. Ese método permite evaluar la pertinencia de un umbral para clasificar en dos categorías (en este caso, en defectuoso y no defectuoso). En un trabajo posterior, el mismo autor [7] empleó regresión logística (como hiciera anteriormente Benlarbi) para estudiar valores aceptables para un conjunto de métricas; los resultados por fuera de esos valores indicarían un mayor riesgo de presencia de defectos. Este autor analizó diferentes versiones del popular IDE Eclipse, realizando posteriormente una comparación con otros dos productos (Rhino y Mozilla); concluye que no pueden generalizarse las cifras postuladas en el primer caso.

El autor también compara los umbrales calculados con los propuestos por Rosenberg [9], concluyendo que éstos son menos seguros para estimar clases defectuosas.

Kaur [8] y otros obtuvieron las métricas CK y otras indicativas de posibles "bad smells" o antipatrones [15] sobre dos versiones del software JFreeChart. Emplearon regresión logística, estimando posibles umbrales en base al Nivel Aceptable de Riesgo (Value for Acceptable Risk Level, VAR) de acuerdo a la metodología elaborada por Bender [16] para analizar posibles "efectos de umbral". De este resumen se desprende que los objetivos buscados principalmente por los trabajos relevados se relacionan con la determinación de métricas que permitan caracterizar piezas individuales de software, más que caracterizar el diseño de las aplicaciones en su conjunto.

2.2 Estudio de Umbrales en Base a Distribuciones de Frecuencia de las Métricas

Rosenberg [9] utilizó histogramas para analizar el efecto de los valores de las métricas de Chidamber y Kemerer [13] sobre la calidad del software. Los valores de los umbrales se basan en el análisis de los histogramas, aunque no se analiza la posible relación entre esos valores y la probabilidad de que módulos presenten errores o deban modificarse.

Erni y Lewerentz [10] analizaron la distribución de métricas orientadas a objetos postulando como potenciales umbrales a la media ± la desviación estándar. Los autores plantean la adopción de uno de esos valores en función de los problemas que plantearía en cada caso cifras demasiado altas o demasiado bajas. Por ejemplo, si para una métrica se considera que un número alto es perjudicial, se adopta $\mu+\sigma$ como umbral (siendo μ la media y σ la desviación estándar).

Chidamber y otros [11] consideraron las distribuciones empíricas de las métricas sugiriendo que el tope para definir un valor como "alto" sería el 80º percentil (es decir, el 80% de las muestras alcanzarían valores por debajo del umbral señalado).

Lanza y Marinescu [4] también propusieron umbrales para métricas comunes partiendo de la distribución de los valores en un grupo de 45 aplicaciones escritas en Java. Clasificaron las medidas según sus frecuencias en bajo, medio, alto y muy alto.

Ferreira y otros [12] estudiaron las distribuciones de las métricas COF (Factor de Acoplamiento), cantidad de métodos públicos, cantidad de campos (atributos) públicos, LCOM (Falta de Cohesión en los métodos) y DIT (Profundidad del árbol de herencia)- Analizaron 40 aplicaciones escritas en Java, computando las métricas con

la herramienta Connecta, y las distribuciones de frecuencia se ajustaron a distribuciones teóricas con la herramienta EasyFit. Los autores dividieron la muestra según diferentes dominios, planteando umbrales distintos para cada uno de ellos.

Recientemente Filó y otros [17] presentaron un estudio de diversas métricas a partir de distribuciones de frecuencia encontradas en una muestra de 111 aplicaciones escritas en Java. A partir de ese análisis proponen umbrales para 17 métricas diferentes. El trabajo muestra similitudes con el que se expone aquí dado que se basa en una muestra públicamente accesible y emplea el análisis de distribución de frecuencia como metodología para detectar posibles umbrales; no obstante, el objetivo de ese estudio es proponer valores referidos a debilidades de clases o paquetes concretos, más que obtener indicadores referidos el diseño en su conjunto.

3 Metodología

Para este trabajo se tuvo en cuenta la infraestructura de investigación propuesta por Gaser y Scacchi [18] para los estudios empíricos sobre F/OSS. A fin de posibilitar la repetibilidad y revisión del trabajo, se utilizaron versiones públicamente accesibles que incluyan su código fuente; las herramientas empleadas siguen el mismo criterio: las mediciones se realizaron con iPlasma [19] y el tratamiento estadístico se efectuó con el entorno R [20].

Se descargó el código fuente de todas las versiones a analizar y se obtuvieron las métricas que se detallan en la tabla 1.

La selección de las métricas apunta a una visión global del software a estudiar; indagando en las características generales del código fuente. Los proyectos analizados tienen en común el haber producido numerosas versiones y que han sido utilizados de manera efectiva por muchos usuarios. Son proyectos "exitosos" en los términos de Crowston [21] y Weiss [22].

Por ello se eligieron las medidas propuestas por Lanza y Marinescu [4], quienes plantean una perspectiva de conjunto para caracterizar el diseño de una aplicación.

Tabla 1. Métricas computadas en el presente trabajo

Métrica	Descripción
CYCLO	Complejidad ciclomática, según la definición de McCabe[22]
LOC	Líneas de código, incluyendo líneas en blanco y comentarios
NOM	Cantidad de métodos y otras operaciones definidas en toda la aplicación
NOC	Cantidad de clases
NOP	Cantidad de paquetes
CALL	Cantidad de invocaciones a operaciones
FOUT	Suma de las clases referenciadas por cada una de las clases

Cada una de estas métricas arroja un valor global dependiente del tamaño del software. Es esperable que una aplicación más grande tenga un mayor número de

clases, mayor cantidad de líneas de código, etc. Lanza y Marinescu plantean entonces la caracterización a partir de las proporciones que se muestran en la tabla 2.

Tabla 2. Proporciones analizadas

Nombre	Descripción	Fórmula
Estructuración de alto nivel	Cantidad de clases por paquete	NOC/NOP
Estructuración de clases	Cantidad de operaciones en relación con le número de clases	NOM/NOC
Estructuración de operaciones	Promedio de la cantidad de líneas de código por operación	LOC/NOM
Complejidad de operación intrínseca	Cantidad de caminos independientes en relación con el tamaño en líneas de código	CYCLO/LOC
Intensidad de acoplamiento	Proporción de los métodos que invocan a cada operación en relación con la cantidad total de operaciones	CALL/NOM
Dispersión de acoplamiento	Proporción de las llamadas de cada clase a métodos de otra respecto del total de llamadas	FOUT/CALL

En este trabajo se computan las métricas mencionadas para 560 versiones de 28 proyectos diferentes escritos en Java. Si bien iPlasma no extrae WMC (métodos ponderados por clase) [13], se puede obtener a partir de la Complejidad Intrínseca, la Estructuración de Operaciones y la Estructuración de Clases. Analizamos la distribución de WMC dada de su amplia utilización.

En total, se computaron 371.934 clases que incluyen 3.907.564 métodos, cubriendo 43.842.838 líneas de código.

4 Análisis de los Datos

La tabla 3 resume las medidas de localización y dispersión [23] para las proporciones estudiadas:

Tabla 3. Medidas localización y dispersión para las métricas computadas

	1º Cuartil	Media	Mediana	3º Cuartil	Desviación estándar	Coef. de Variación	Coef. de asimetría
Complejidad Intrínseca	0,1727	0,2076	0,2053	9,2236	0,041	0,13	0,87
Estructuración de Operaciones	7,973	10,27	10,86	14,310	4,277	0,41	0,89
Estructuración de Clases	9,581	10,91	11,74	13,33	3,453	0,29	0,93
Estructuración de Alto Nivel	7,503	9,98	12,55	17,81	6,965	0,7	1,41
Intensidad de Acoplamiento	0,556	0,588	0,585	0.616	0,168	0,29	17,63
Dispersión de acoplamiento	2,534	2,992	3,024	3,399	0,82	0,27	0,13

Las distribuciones de estas proporciones muestran que la Complejidad Intrínseca, la Intensidad de Acoplamiento, la Estructuración de Clases, y la Dispersión de acoplamiento exhiben menor dispersión. De ellas, la Intensidad de acoplamiento presenta una distribución de frecuencias con una asimetría más marcada.

Se analizan a continuación las gráficas de distribución que se muestran en las figuras 1 a 4; posteriormente, se considera la variación entre deciles, a fin de observar en cuál de ellos se constata mayor distancia entre los valores.

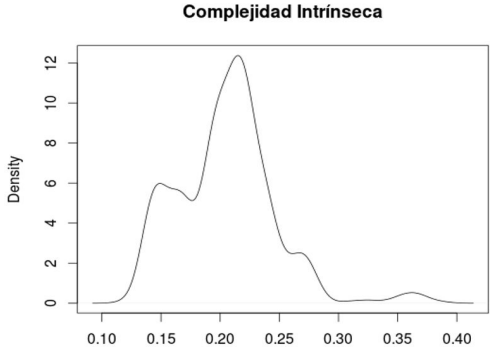


Fig. 1. Distribución de la Complejidad Intrínseca.

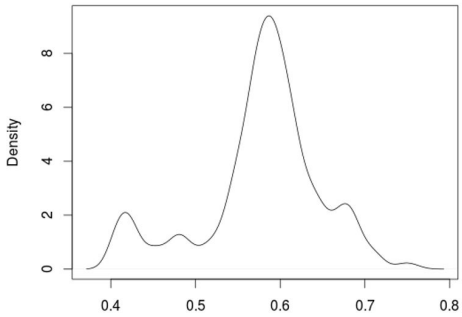


Fig. *2*. *Distribución de la Intensidad de acoplamiento*

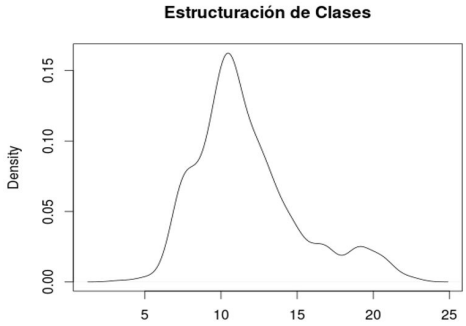


Fig. *3*. *Distribución de la Estructuración de Clases*

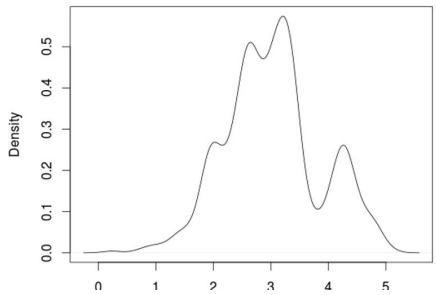


Fig. *4.* *Distribución de la Dispersión de acoplamiento*

Respecto de la Estructuración de clases, la gráfica muestra una alta concentración en torno a la moda. En tanto, la Estructuración de Operaciones exhibe varios picos, con una caída pronunciada en las frecuencias de valores mayores a 20.

Las distribuciones de frecuencias de la Estructuración de Clases y de la Intensidad de Acoplamiento muestran claramente una zona modal a partir de las cuales las frecuencias decrecen en ambos sentidos. Analizando la variación de los deciles podemos proponer umbrales superiores buscando cambios pronunciados entre los valores sucesivos de los mismos, lo que implicaría un descenso rápido de las frecuencias. La tabla 4 resume los deciles para las distribuciones de las métricas consideradas.

Tabla 4. Valores que correspoden a los diferentes deciles para las distribucionesobservadas en las proporciones analizadas.

Decil	C.I.	E.C.	E.O.	D.A.	I.A.	WMC
"0%"	0,121	3,375	3,832	0,234	0,406	7,346
"10%"	0,148	7,721	5,707	1,968	0,468	9,774
"20%"	0,166	9,001	6,876	2,372	0,544	16,233
"30%"	0,191	9,950	8,218	2,603	0,567	18,103
"40%"	0,196	10,458	9,425	2,750	0,575	20,463
"50%"	0,208	10,905	10,272	2,992	0,588	22,000
"60%"	0,216	11,730	11,301	3,172	0,594	23,772
"70%"	0,221	12,639	13,622	3,337	0,607	26,589
"80%"	0,233	14,092	14,820	3,665	0,625	36,062
"90%"	0,251	16,936	15,830	4,270	0,668	45,683
"100%"	0,385	22,773	29,321	5,084	0,759	182

En la Fig. 5 Observamos también la distribución de WMC, calculada a partir de otras métricas según se señaló más arriba.

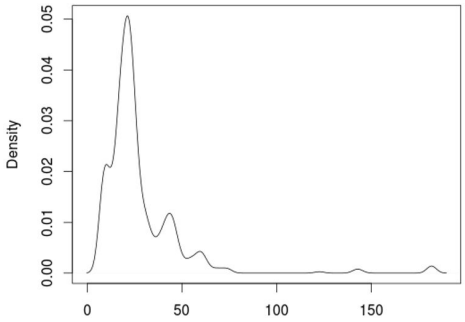


Fig. 5. *Distribución de WMC (Métodos ponderados por clase)*

Respecto de la estructuración de clases, se observa que entre el tercer y el octavo decil (60% de los casos) tienen en promedio entre 9 y 14,09 métodos por clase en promedio. Por otra parte, analizando el histograma (sesgado hacia la izquierda) se observa que la frecuencia de los valores superiores a 13 decrece significativamente, lo que convierte a ese valor en un posible umbral.

En cuanto a la Intensidad de acoplamiento, el decrecimiento más pronunciado comienza con el noveno decil , por lo que un valor superior a 0,668 podría reflejar acoplamiento excesivo.

Si contemplamos WMC, vemos que la curva de distribución de frecuencias tiene un máximo posterior a la moda; recién luego del 9º decil se verifica un descenso constante, por lo que el umbral para esta métrica sería de 45,68.

4.1 Umbrales Propuestos en Otros Trabajos

La tabla 5 muestra los valores propuestos en diversos trabajos para la métrica WMC

Tabla 5. Valores propuestos en otros trabajos como umbral para WMC.

Métricas	Shatnawi [7]	Rosenberg [9]	Lanza y Marinescu [4]	Herbold et al [24]
WMC	20	100	47	100

Como puede observarse, las cifras postuladas son muy disímiles. Cabe señalar, no obstante, que el valor planteado por Lanza y Marinescu es similar al obtenido en este trabajo sobre una muestra mucho mayor.

5 Conclusiones y Trabajos Futuros

Este trabajo adopta dos enfoques novedosos respecto de la búsqueda de umbrales útiles de métricas de diseño orientado a objetos. Por un lado, se buscan referencias para valores promedio, indicativos del diseño general de una aplicación o de un componente de software, en lugar de proponer referencias para clases particulares o módulos particulares.

Por otra parte, se retoma el análisis de distribuciones de frecuencia de los valores de las métricas, buscando topes identificables a partir de las curvas de distribución de frecuencias.

En concreto, una aplicación que presente más de 14 métodos por clase, o un promedio de WMC por encima de 46, muestra una estructuración muy poco común, con clases con demasiados métodos; de esto no se concluye que tal o cual clase se potencialmente defectuosa, sino que el diseño en su conjunto tiende a agrupar demasiadas funciones sobre las clases.

Si bien la métrica compuesta denominada Intensidad de Acoplamiento muestra un comportamiento que sugiere un tope viable, es una medida que pocas herramientas computan. No obstante, puede advertir que el diseño tiende a que las clases requieran de muchas otras para realizar sus tareas.

Teniendo en cuenta estos aspectos, sería conveniente adoptar la métrica Estructuración de Clases (promedio de la cantidad de métodos por clase) por su sencillez, o WMC, que es captada por una amplia cantidad de herramientas. Cabe señalar, sin embargo, que al basarse los umbrales propuestos en valores promedio, éstos pueden no ser relevantes para aplicaciones pequeñas, dadas las características del promedio como medida de tendencia central influenciada por valores extremos.

En futuros trabajos se prevé analizar otras métricas relacionadas con el acoplamiento, y estudiar su comportamiento con relación a las distintas etapas de la evolución de una aplicación. De esta forma, esperamos conocer diversos aspectos de la evolución del software y generar más elementos para una evaluación preliminar de una aplicación F/OSS a partir de la información que se puede obtener de su código fuente.

6 Referencias

1. Bourque, P., Wolff, S., Dupuis, R., Sellami, A., Abran, A.: Lack of Consensus on Measurement in Software Engineering: Investigation of Related Issues. In: 14th International Workshop on Software Measurement (IWSM) IWSM-Metrikon 2004, Konigs Wusterhausen. pp. 321–333. Shaker-Verlag, Magdeburg, Germany (2004).
2. Sjøberg, D.I.K., Anda, B., Mockus, A.: Questioning Software Maintenance Metrics: A Comparative Case Study. In: Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement. pp. 107–110. ACM, New York, NY, USA (2012).
3. Fowler, M., Beck, K., Brant, J., Opdyke, W., Roberts, D.: Refactoring: Improving the Design of Existing Code. Addison-Wesley Professional (1999).
4. Lanza, M., Marinescu, R., Ducasse, S.: Object-Oriented Metrics in Practice. Springer-Verlag New York, Inc., Secaucus, NJ, USA (2005).
5. Benlarbi, S., Emam, K.E., Goel, N., Rai, S.N.: Thresholds for Object-Oriented Measures. In: ISSRE. pp. 24–39. IEEE Computer Society (2005).
6. Shatnawi, R., Li, W., Swain, J., Newman, T.: Finding software metrics threshold values using ROC curves. J. Softw. Maint. Evol. Res. Pract. 22, 1–16 (2010).
7. Shatnawi, R.: A Quantitative Investigation of the Acceptable Risk Levels of Object-Oriented Metrics in Open-Source Systems. IEEE Trans. Softw. Eng. 99, 216–225 (2010).

8. Kaur, S., Singh, S., Kaur, H.: A Quantitative Investigation Of Software Metrics Threshold Values At Acceptable Risk Level. Int. J. Eng. Res. Technol. 2, (2013).
9. Rosenberg, L.: Applying and Interpreting Object Oriented Metrics. Presented at the Software Technology Conference (1998).
10. Erni, K., Lewerentz, C.: Applying Design-metrics to Object-oriented Frameworks. In: Proceedings of the 3rd International Symposium on Software Metrics: From Measurement to Empirical Results. p. 64–. IEEE Computer Society, Washington, DC, USA (1996).
11. Chidamber, S.R., Darcy, D.P., Kemerer, C.F.: Managerial Use of Metrics for Object-Oriented Software: An Exploratory Analysis. IEEE Trans Softw Eng. 24, 629–639 (1998).
12. Ferreira, K.A.M., Bigonha, M.A.S., Bigonha, R.S., Mendes, L.F.O., Almeida, H.C.: Identifying Thresholds for Object-oriented Software Metrics. J Syst Softw. 85, 244–257 (2012).
13. Chidamber, S.R., Kemerer, C.F.: A Metrics Suite for Object Oriented Design. IEEE Trans Softw Eng. 20, 476–493 (1994).
14. Brito e Abreu, F., Carapuca, R.: Object-Oriented Software Engineering: Measuring and Controlling the Development Process. In: Proc. Int'l Conf. Software Quality (QSIC) (1994).
15. Fowler, M., Beck, K.: Bad Smells in Code. In: Refactoring: Improving the design of existing code. pp. 75–88. Addison-Wesley (1999).
16. Bender, R.: Quantitative Risk Assessment in Epidemiological Studies Investigating Threshold Effects. 41, 305–319 (1999).
17. Filó, T., Bigonha, M., Ferreira, K.A.M.: A Catalogue of Thresholds for Object-Oriented Software Metrics. In: SOFTENG 2015. pp. 48–55 (2015).
18. Gasser, L., Scacchi, W.: Towards a Global Research Infrastructure for Multidisciplinary Study of Free/Open Source Software Development. In: Russo, B., Damiani, E., Hissan, S., Lundell, B., and Succi, G. (eds.) Open Source Development, Communities and Quality. pp. 143–158. Springer, Boston (2008).
19. Marinescu, C., Marinescu, R., Mihancea, P.F., Ratiu, D., Wettel, R.: iPlasma: An Integrated Platform for Quality Assessment of Object-Oriented Design. In: ICSM (Industrial and Tool Volume). pp. 77–80 (2005).
20. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2013).
21. Crowston, K., Annabi, H., Howison, J.: Defining Open Source Software Project Success. In: in Proceedings of the 24th International Conference on Information Systems (ICIS 2003). pp. 327–340 (2003).
22. Weiss, D.: Measuring success of open source projects using web search engines. In: First International Conference on Open Source Systems (2005).
23. Devore, J.L.: Probabilidad y estadística para Ingeniería y ciencias. CENGAGE Learning (2005).
24. Herbold, S., Grabowski, J., Waack, S.: Calculation and Optimization of Thresholds for Sets of Software Metrics. Empir. Softw Engg. 16, 812–841 (2011).1. Bourque, P., Wolff, S., Dupuis, R., Sellami, A., Abran, A.: Lack of Consensus on Measurement in Software Engineering: Investigation of Related Issues. In: 14th International Workshop on Software Measurement (IWSM) IWSM-Metrikon 2004, Konigs Wusterhausen. pp. 321–333. Shaker-Verlag, Magdeburg, Germany (2004).

Verification and validation of domain specific languages using Alloy

Ana Garis[1] and Alejandro Sánchez[1]

Universidad Nacional de San Luis, Argentina
{agaris,asanchez}@unsl.edu.ar

Abstract. A *domain specific language* (DSL) focuses on a particular problem domain, facilitating the specification of its instances. Since they are frequently defined using imprecise languages, such as UML, they present ambiguities and their *verification and validation* (V&V) becomes complex. This paper proposes an approach to the precise definition of DSLs using Alloy – a formal language with tool-support that enables its V&V. The approach is illustrated with a DSL for Software Architecture.

1 Introduction

A *domain specific language* (DSL) [18] restricts its primitives to a problem domain, usually aiming at facilitating the development of specifications by domain experts. DSLs are typically defined through a metamodel that includes the language elements and the relationships among them [17]. Usually, *UML class diagrams* (CD) [11] are used for this [3], with the main disadvantage of an imprecise semantics, and the lack of a tool-supported mechanism for checking the quality of the resulting metamodel.

This paper proposes an approach that aims at addressing such disadvantage. The approach uses Alloy, a formal language designed for performing automatic analysis, and whose models resemble class diagrams [9]. Alloy includes friendly tool-support for V&V, which is based on a bounded SAT analyser. The mechanism assumes a metamodel specified as a CD, that can optionally be enriched with formulas expressed in the *Object Constraint Language* (OCL) [12]. It prescribes translating the CD and associated OCL formulas into Alloy, verifying and validating the model in Alloy's framework, and then translating it back into a CD and OCL formulas. Domain experts iterate improving the model's quality using Alloy. Therefore, it is possible to generate instances of the model, establish if it is inconsistent, find a counterexample for some assertion, study generated instances and modify the model accordingly, extend it with further formulas that need to be verified, and generate instances again.

A DSL for Software Architecture called Archery [14, 15] is used to illustrate the approach. The language is designed for modelling, animating, analysing, and verifying software architectures in terms of architectural patterns. The language semantics are given by an encoding into a process algebra, but its metamodel is described using a CD, which permits specifying ill-defined models. We study Archery's metamodel and suggest improvements.

The work is framed within the approach described in [7]. The translations the approach relies on were prototyped in Haskell [8] and are available at [13].

Although, we sketched the proposed approach in [7], it omitted the required specifications to be applied in particular domains. This work details how use it for V&V of DSLs.

The contribution of the paper lies in a tool-supported approach based on Alloy to the V&V of DSL metamodels and in the development of an example in the domain of Software Architecture.The rest of the paper is structured as follows: sections 2 and 3 briefly describe the Archery language and Alloy's framework, respectively; section 4 describes how models in Alloy are obtained from CDs; section 5 illustrates the approach by performing the V&V of Archery's metamodel; section 6 describes related work; and section 7 concludes the publication.

2 The Archery language

The Archery language is for modelling the structure and behaviour of software architectures. It allows defining the basic building blocks of architectures: (architectural) patterns and (architectural) elements. A pattern consists of a set of elements representing either component or connector types, that are specified in terms of their interfaces and behaviours. Interfaces are sets of ports, atomic events of interaction, and behaviours are sequential processes that describe how the activities in element's instances take place. The client-server pattern, and the pipes and filters pattern, for instance, are shown in Listing 1.

```
1  pattern ClientServer()
2  element Server()
3    interface in rreq; out sres; act cres;
4    proc Server() = rreq.cres.sres.Server();
5  element Client()
6    interface in rres; out sreq; act prcs;
7    proc Client() = prcs.sreq.rres.Client();
8  end
9  pattern PipeFilter()
10 element Pipe()
11   interface in acc; out fwd;
12   proc Pipe() = acc.fwd.Pipe();
13 element Filter()
14   interface in rec; out snd; act trans;
15   proc Filter() = rec.trans.snd.Filter();
16 end
```

Listing 1. Client-server and pipes and filters patterns

Architectures are built out of defined patterns and elements. An architecture is regarded as a pattern instance that describes a particular configuration of element instances through a set of attachments linking their ports, and a set of renamings changing the externally visible names of ports. Both patterns and elements act as types for instances, which are kept and referenced through typed

variables. A variable has an identifier and a type that must match an element or pattern name. Allowed values are instances of a type, that do not necessarily need to match the variable's own type. An attachment includes a port reference to an output port and a port reference to an input port. A port reference is a pair of identifiers that identify a variable, and a port of the variable's instance. In addition, the language supports hierarchical composition by allowing the definition of configurations where attachments indifferently link ports of pattern and element instances. For example, the architecture in Listing 2 defines a server, which is hierarchically composed of instances of the pipes and filters pattern.

```
cs : ClientServer = architecture ClientServer()
instances
  s1 : Server = architecture PipeFilter()
  instances
    f1:Filter=Filter(); p1:Pipe=Pipe(); f2:Filter=Filter();
  attachments
    from f1.send to p1.accept;
    from p1.forward to f2.rec;
  interface
    f1.rec as rreq;
    f2.send as sres;
  end
end
```

Listing 2. A hierarchically composed server

The CD specifying the ARCHERY's metamodel for architectures is shown in Figure 1. It is taken from [15] with the exception of `PortType`. A class specifying the distinction the language makes among ports was omitted in the original CD, but it is included at this stage to enable a richer analysis.

The metamodel is underspecified and presents several issues. We illustrate the approach by making evident and addressing some of them.

3 Alloy

In Alloy, a *signature declaration* denotes a set of atoms. An *atom* is a unity with three basic properties: it is indivisible (it cannot be divided into smaller parts), it is immutable (its properties remain over time), and it is uninterpreted (it does not have inherent properties). Signature declarations can introduce *fields*, which represent a relation among signatures. Listing 3 shows the signature declarations that constitute ARCHERY's metamodel in Alloy.

```
sig Instance { prt: some Port }
sig ElementInstance extends Instance {
   act: set Act }
sig PatternInstance extends Instance {
   att: set Attachment,
   ren: set Renaming,
```

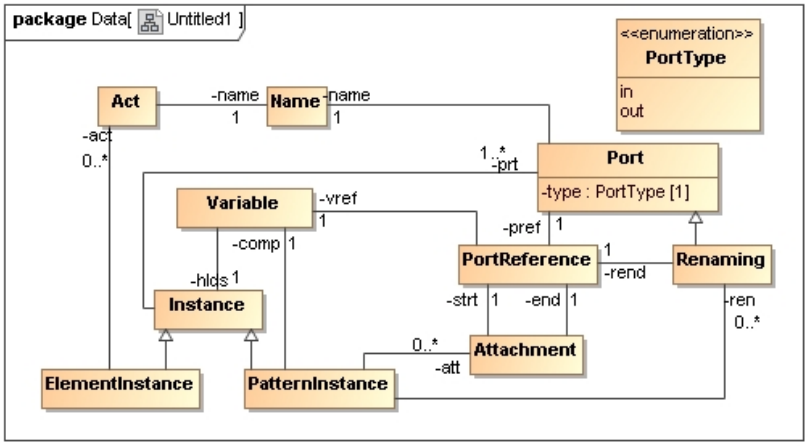


Fig. 1. Class diagram for ARCHERY's architectural specifications

```
7     comp: some Variable }
8  sig Attachment {
9     strt: one PortReference,
10    end:  one PortReference }
11 sig PortReference {
12    pref: one Port,
13    vref: one Variable }
14 sig Port {
15    name: one Name,
16    type: one TypePort }
17 sig Renaming extends Port {
18    rend: one PortReference }
19 sig Variable { hlds: one Instance }
20 sig Act { name: one Name }
21 sig Name{}
22 enum TypePort { In, Out }
```

Listing 3. Metamodel of Archery's architectures in Alloy

The cardinality in relationship between a signature and another can be constrained using the keywords as follows: **lone** for zero or one, **one** for exactly one, **some** for one or more, and **set** for any number.

Facts, *predicates* and *functions* describe invariants, named constraints, and named expressions, respectively. The difference between a fact and a predicate is that the former always holds, while the latter is only verified when invoked. Invariants can also be defined in the context of each signature. *Assertions* allows to express properties that are expected to hold as consequence of specified facts.

The analyser is instructed through *commands* **run** and **check**. Command **run** checks model consistency by requesting a valid instance, and command

check verifies an assertion by searching for a counterexample. Both commands optionally define a scope, overriding the default bound of the number of instances allowed for each signature.

Since in Alloy everything is a relation, it defines the typical set's relational operations: `+` (union), `-` (difference), `&` (intersection), `.` (join), `->` (cartesian product). It also provides two important operators over binary relations, that make its logic more expressive than first-order logic: `^` (transitive closure) and `*` (transitive-reflexive closure).

4 Modelling domain specific languages in Alloy

The translations into Alloy and back are organised in four modules [5]. Prototype tools `CD2Alloy` and `OCL2Alloy` accept a CD and an OCL specification, respectively, and yield the corresponding Alloy model. The dual prototype tools are `Alloy2CD` and `Alloy2OCL`, which accept an Alloy model, and produce a CD and a OCL specification, correspondingly. CD and OCL specifications are handled using the OMG standard XML Metadata Interchange (XMI) format. Listing 3 depicts the model generated from the CD corresponding to the metamodel of ARCHERY's architectures.

A signature is defined for each class in the CD as follows: `ElementInstance`, `PatternInstance`, `Instance`, `Attachment`, `PortReference`, `Renaming`, `Variable`, `Port`, `Act`, `Name` and the class enumeration `TypePort`. Note the inheritance relation is represented in Alloy using keyword **extends** and abstract classes are marked with keyword **abstract**.

Associations corresponds to fields. Signature `Instance`, for example, declares a field `prt` and signature `ElementInstance` a field `act`. The multiplicity in association ends, also has an equivalence in Alloy: `0..*` is **set**; `1..*` is **some**; `0..1` is **lone**, `1` is **one**; and if it is absent, the default is **set**. Therefore, keyword **some** in the declaration of field `prt`, for instance, indicates that each `Instance` has one or more ports.

5 Verifying and validating domain specific languages

The V&V of the DSL's metamodel takes place once its Alloy model is obtained. It is performed as iterations in which domain experts generate instances, modify the model, and generate instances again to confirm consequences of changes. Modifications can be to correct relations, or to add formulas that forbid incorrect instances. Instances are generated either to show examples of the model, or to show counterexamples that disproof a given formula.

It is suggested to start with the generation of examples, allowing domain experts to gain a better understanding of the model. This is done with the **run** command, which in its most simple variant (**run** {}), is usually enough to simulate and find problems or underspecifications in the model, namely, the model allowing instances that are forbidden in the domain.

Subsequently, the model is extended with formulas removing incorrect instances. Such formulas are checked with **`check`** commands, which generate counterexamples when they are not verified, and allow domain experts to learn where a given formula is weak.

These steps are illustrated with the Alloy model of ARCHERY's metamodel. Figure 2 shows an instance generated with the analiser that is enough to find several problems. The instance shows:

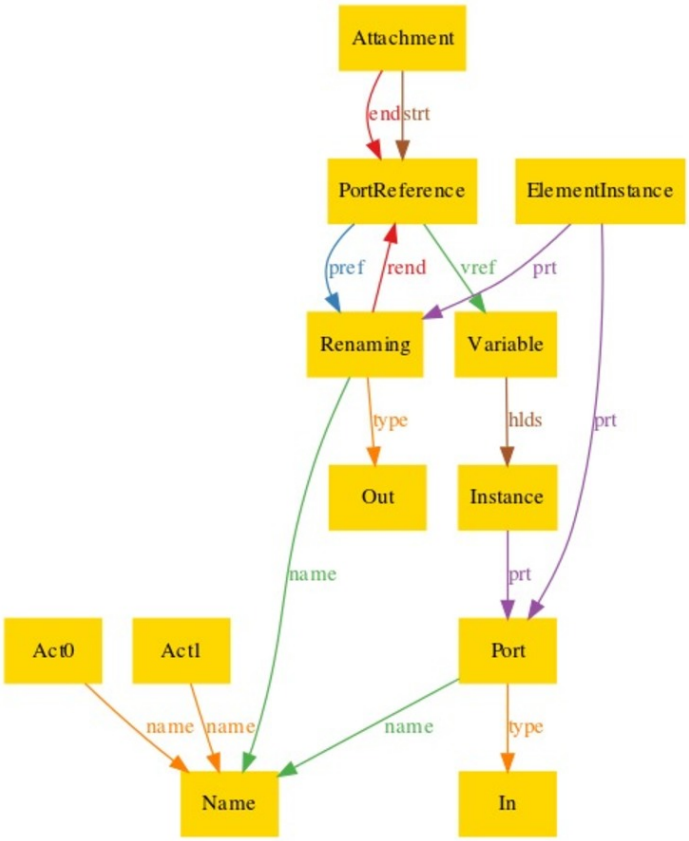


Fig. 2. An instance found with Alloy analyzer.

(i) An instance of class `Instance` that is a superclass of `ElementInstance` and `PatternInstance`. This class is an abstraction of element and pattern instance, which represent components and architectures, respectively. Direct occurrences of this abstraction are incorrect.

(ii) Actions (`Act0` and `Act1`) unrelated to an element instance. All actions must be contained in an element instance, since they represent an observable transition in the state of the component.
(iii) A name (`Name`) is shared between ports (`Renaming` and `Port`) that belong to the same element instance. However, names cannot be shared between ports (including renamings) or actions of the same instance.
(iv) An attachment references the same port twice. The fields `strt` and `end` connect `Attachment` to the same `PortReference`. This makes no sense, since a component cannot interact with itself at the same port. Moreover, communication flows from outward ports to inward ports.
(v) A renaming references to itself. A renaming changes the name of a port, and no cycle can be created by the reference it creates.

Other similar detected problems, such as `Attachment` not having an associated pattern instance, are omitted in this description.

We correct problems (i) to (v) by modifying the model and adding constraints to it. Line 1 addresses problem (i) by declaring signature `Instance` as **`abstract`**: only instances of `ElementInstace` or `PatternInstance` are permitted. Problems (ii) to (v) are addressed including facts in lines (2) to (5), one fact in each respective line, making the model stronger.

```
1 abstract sig Instance { prt: some Port }
2 fact { all a: Act | a in ElementInstance.act }
3 fact { all e: ElementInstance | no e.act.name & e.prt.name }
4 fact { all a: Attachment | no a.end & a.strt }
5 fact { all r: Renaming | not r.^(rend.pref) in Renaming }
```

Problem (iv) unveils a more involved issue requiring further treatment. Attachments have a direction that must be respected. We verify if the model ensures such principle by executing a **`check`** command with the assertion as follows,

```
assert attachInOut { all a:Attachment |
  a.strt.pref.type=Out and a.end.pref.type=In }
check attachInOut
```

which requires attachments to connect `Out` ports to `In` ports. The analyser finds the counterexample shown in Figure 3. Attachments connect `PortReference1` to `PortReference0` through fields `strt` and `end`. However, they are referencing to the same out port, thus violating the assertion. Moreover, the counterexample also shows that the essence of problem (iv) was not addressed by the fact in line (4): attachments may include port references that are different, but that refer to the same port. Adding the assertion as a fact addresses both issues.

Iterations continue, as further issues can be observed, until a satisfactory point is reached. Then, the model can be translated back into a CD enriched with OCL. For example, the facts previously presented, are translated (with `Alloy2OCL`) into the OCL specification as follows.

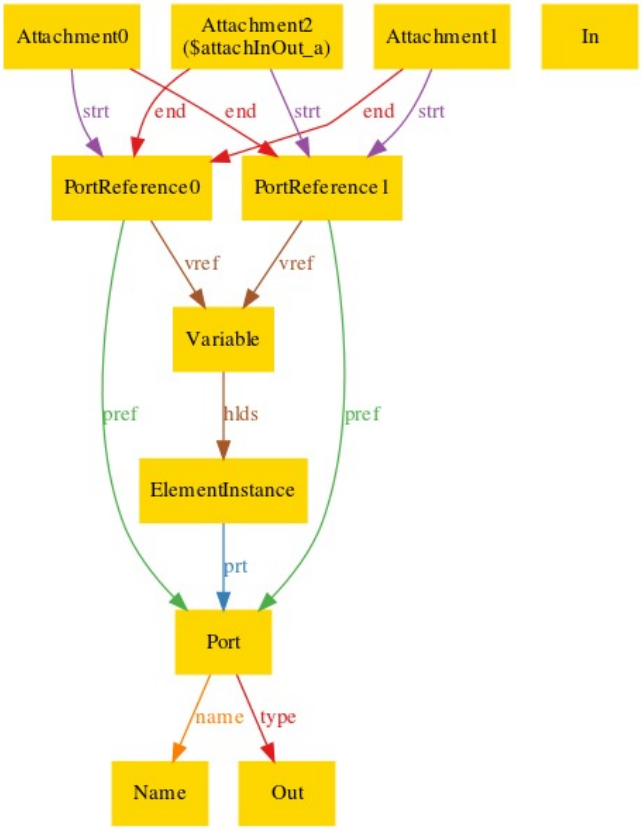


Fig. 3. Counterexample found with Alloy analyzer.

```
context Port inv:
 Instance.allInstances()->exists(v1 | v1.prt->includes(self))
context Act inv:
 ElementInstance.allInstances()->exists(v3 | v3.act->includes(
    self))
context ElementInstance inv:
 (Name.allInstances()->select(v4 | (Act.allInstances()->exists(
    v5 | (self.act->includes(v5) and (v5.oclIsKindOf(Port) and
    v5.oclAsType(Port).name->includes(v4)))) and Port.
    allInstances()->exists(v7 | (self.prt->includes(v7) and v7.
    name->includes(v4)))))->size() = 0)
context Attachment inv:
 (PortReference.allInstances()->select(v9 | (self.end->includes(
    v9) and self.strt->includes(v9)))->size() = 0)
```

6 Related work

Several works, including [1, 2, 10], use formal frameworks to address issues that emerge in the definition of DSLs. Amrani presents a formal specification of Kermeta [1], a metamodelling framework for modeling of DSLs. Its formal specification allows making DSL definitions more precise. Bodeveix et al. combine Bossa (a DSL for process schedulers) and formal method B for ensuring correctness [2]. In particular, B is used to define the correctness of a Bossa specification and to produce certified schedulers. James et al. [10] introduce a methodology for including formal methods in DSLs. The methodology is based on a formal algebraic specification language named CASL. The DSL is first modeled with a CD, then it is automatically translated into a formal specification in CASL.

Alloy potential for DSL modelling has been studied in [16, 4, 6]. Sen et al. expose an approach for using Alloy in order to improve DSLs definition [16]. A set of recommendations are generated to complete partial models which represent DSLs. The completion feature is centred around Alloy. Challenger et al. establish a formal semantics of a DSL for Semantic Web enabled Multi-agent Systems employing Alloy. Static and dynamic aspects of the interaction between agents and semantic web services are considered. Additionally, they explain how to perform automatic analysis for checking these models. Unlike our approach, these works propose the representation of a specific DSL in Alloy, instead of a general approach for modelling DSLs. Alloy is used to support the Lightning tool which allows the representation of modelling languages [6]. The paper describes Lightning's capabilities for verification of a DSL related to structured business processes. This work uses Alloy for supporting the Lightning tool instead of applying directly Alloy's tool-support in order to V&V a DSL metamodel.

7 Conclusion and future work

This paper presented an approach for modelling, verifying and validating domain specific languages (DSLs) using Alloy. It detailed a concrete mechanism for automatically obtaining Alloy models from UML class diagrams (CDs) enriched with formulas specified in the Object Constraint Language (OCL), and a mechanism for translating the models back into CDs enriched with OCL. The approach allows domain experts to gain understanding of their metamodels, modify them, possibly specifying restrictive formulas, so incorrect instances that emerge are prevented. An illustrative example in the domain of Software Architecture was developed.

Future work includes the extension of the approach to study instances of DSLs, and the development of other case studies in order to adjust the approach and validate it further.

References

1. Moussa Amrani. A formal semantics of kermeta. In *Formal and Practical Aspects of Domain-Specific Languages: Recent Developments*, 2012.

2. Jean-Paul Bodeveix, Mamoun Filali, Julia Lawall, and Gilles Muller. Formal methods meet domain specific languages. In Judi Romijn, Graeme Smith, and Jaco van de Pol, editors, *Integrated Formal Methods*, volume 3771 of *Lecture Notes in Computer Science*, pages 187–206. Springer Berlin Heidelberg, 2005.
3. Achim D. Brucker and Jürgen Doser. Metamodel-based UML notations for domain-specific languages. In Jean Marie Favre, Dragan Gasevic, Ralf Lämmel, and Andreas Winter, editors, *4th International Workshop on Software Language Engineering (ATEM 2007)*. 2007.
4. Moharram Challenger, Sebla Demirkol, Sinem Getir, Marjan Mernik, Geylani Kardas, and Tomaz Kosar. On the use of a domain-specific modeling language in the development of multiagent systems. *Eng. Appl. of AI*, 28:111–141, 2014.
5. Alcino Cunha, Ana Garis, and Daniel Riesco. Translating between Alloy specifications and UML class diagrams annotated with OCL. *Software & Systems Modeling*, pages 1–21, 2013.
6. Loıc Gammaitoni, Pierre Kelsen, and Fabien Mathey. Verifying modelling languages using lightning: a case study. *11th Workshop on Model Driven Engineering, Verification and Validation MoDeVVa 2014*, page 19, 2014.
7. Ana Garis and Alejandro Sanchez. Especificación formal de lenguajes específicos del dominio utilizando Alloy. In *Proceedings of the XVII the Workshop de Investigadores en Ciencias de la Computación*, WICC'15, 2015.
8. Haskell website http://www.haskell.org.
9. Daniel Jackson. *Software Abstractions: Logic, Language, and Analysis*. MIT Press, revised edition, 2012.
10. Phillip James and Markus Roggenbach. Encapsulating formal methods within domain specific languages: A solution for verifying railway scheme plans. *Mathematics in Computer Science*, 8(1):11–38, 2014.
11. OMG. UML Superstructure, version 2.4.1, 2011.
12. OMG. Object Constraint Language, version 2.4, 2014.
13. Translations MDA - Alloy http://sourceforge.net/projects/alloymda.
14. Alejandro Sanchez, Luis Barbosa, and Daniel Riesco. A language for behavioural modelling of architectural patterns. In *Proceedings of the Third Workshop on Behavioural Modelling*, pages 17–24, 2011.
15. Alejandro Sanchez, Luis S. Barbosa, and Daniel Riesco. Specifying structural constraints of architectural patterns in the archery language. In Theodore E. Simos and Charalambos Tsitouras, editors, *Proceedings of the International Conference on Numerical Analysis and Applied Mathematics 2014 (ICNAAM-2014)*, volume 1648, pages 310008(1) – 310008(5). AIP Proceedings, 3 2015.
16. Sagar Sen, Benoit Baudry, and Hans Vangheluwe. Towards domain-specific model editors with automatic model completion. *Simulation*, 86(2):109–126, 2010.
17. James Willans Tony Clark, Paul Sammut. *Applied metamodelling: A foundation for language driven development*. Ceteva, second edition edition, 2008.
18. Arie van Deursen, Paul Klint, and Joost Visser. Domain-specific languages: An annotated bibliography. *SIGPLAN Not.*, 35(6):26–36, 2000.

Modelo para aplicaciones sensibles al contexto (MASCO): Formalización de las capas relacionadas con datos tomados desde el contexto.

María del Pilar Galvez Díaz[1,2], Nélida Raquel Caceres[1]

1 Universidad Nacional de Jujuy, Facultad de Ingeniería, San Salvador de Jujuy, Jujuy, Argentina
2 Universidad Católica de Santiago del Estero, Departamento Académico San Salvador, San Salvador de Jujuy, Argentina
{mdpgalvezdiaz,nrcaceres}@fi.unju.edu.ar

Resumen. El modelo de Aplicaciones Sensibles al Contexto (MASCO) presenta una arquitectura en capas que provee servicios para aplicaciones sensibles al contexto, considera el tratamiento de más de una variable de contexto, la relación de una entidad con más de una variable de contexto, como así también la interacción entre variables de contexto y entidades. El modelo fue validado usando diferentes experimentaciones las cuales permitieron su refinamiento y determinar sus atributos de calidad. El modelado se realizó usando UML, al ser un lenguaje visual, sus diagramas no siempre son lo suficientemente expresivos como para especificar todos los aspectos que son relevantes. En este trabajo se presenta una introducción a la formalización de MASCO utilizando OCL de forma que el modelo no presente ambigüedades en su interpretación y uso como framework en el desarrollo de sistemas Context Aware.

Palabras Clave: Context Aware, modelado, UML, OCL, MASCO.

1 Introducción

El grupo de investigación de Aplicaciones Sensibles al Contexto (GRISECO) ha desarrollado un Modelo para Aplicaciones Sensibles al Contexto (MASCO), orientado a objetos basado en capas.

MASCO considera el tratamiento de más de una variable de contexto, la relación de una entidad con más de una variable de contexto, como así también la interacción entre variables de contexto y entidades. El modelo fue validado usando diferentes experimentaciones las cuales permitieron su refinamiento y determinar sus atributos de calidad como modelo de arquitectura.

MASCO fue modelado usando UML, al ser un lenguaje visual, los diagramas de UML no siempre son lo suficientemente expresivos como para especificar todos los aspectos que son relevantes en una especificación, por ejemplo la especificación de restricciones adicionales con respecto a los objetos que pertenecen a un modelo. Ante la falta de un lenguaje de especificación formal de restricciones, estas se describen usando lenguaje natural lo cual introduce ambigüedades. El propósito de este trabajo

es presentar una introducción a la formalización de MASCO utilizando OCL de forma que el modelo no presente ambigüedades en su interpretación y aplicación como framework en el desarrollo de sistemas Context Aware.

En el apartado 2 se describe MASCO, en el apartado 3 se describe OCL, en el apartado 4 se comienza con la formalización del modelo, en el apartado 5 se presentan las conclusiones, y finalmente en el apartado 6, se detallan las referencias.

2 MASCO

MASCO es un modelo de arquitectura en capas, modelado con UML, orientado a objetos, que fue desarrollado con el objetivo de ser usado como framework para desarrollar sistemas sensibles al contexto, Fig. 1, y fue evolucionando y refinándose a través de los trabajos presentados en [1] [2], [3], [4] y [5].

En MASCO se identifican cinco capas: Application Layer: se encuentran los objetos del dominio de la aplicación; Context Layer: contiene los objetos necesarios para procesar la información de contexto; Service Layer: contiene los objetos necesarios para proveer servicios tanto internos como externos al sistema; Sensing Concerns Layer: se encarga de interpretar o traducir los datos que provienen de la capa Hardware Abstractions; Hardware Abstractions Layer: esta capa agrupa los objetos que representan los sensores y actuadores.

Los trabajos realizados permitieron determinar además los atributos de calidad de MASCO y atributos o requisitos técnicos de calidad no funcionales: Integridad Conceptual, Corrección, Compleción, Capacidad de realización, Modificabilidad, Testeabilidad, Simplificación de la funcionalidad, Reusabilidad, y Portabilidad [5].

Descripción del funcionamiento de las capas relacionadas con datos tomados desde el contexto. Cada vez que una entidad cambia su posición o una variable de contexto ingresa un nuevo valor, situaciones que se reflejan en un objeto IRport, un objeto SensingConcern es notificado. SensingConcern abstrae las reglas de sensado, representadas por una subclase SensingPolicy. El objeto SensingPolicy analiza el valor y, si es necesario, produce alguna transformación en los datos y, posteriormente, notifica a ContextAggregator. Si se trata de un valor de ubicación busca a qué objeto pertenece esa ubicación en Context Layer (ej: Office, Corridor). En esta capa se estableció la clase Context (Fig. 1) con la subclases Location (para considerar la variable de contexto Ubicación tanto de un usuario como otras entidades relevantes para la aplicación) [1], y AnotherVariable1, AnotherVariable2… AnotherVariableN (para considerar otras variables de contexto tales como presión, temperatura, nivel de agua, que son inherentes, por ejemplo, a los procesos industriales o aplicaciones médicas). En tiempo de implementación Sensing Concern actúa como un patrón Strategy y un patrón Observer y ContextAggregator como un patrón Observer [6].

En Application Layer la clase Discoverer es la responsable de mantener la información sobre los procesos, entidades y variables inherentes a una aplicación y la clase EntityAggregator es la responsable de coordinar la interacción entre objetos Entity [3]. Esta clase, en tiempo de implementación, constituye un patrón Mediator. Cada objeto Process puede estar relacionado con uno ó más objetos Entity. A su vez,

cada uno de éstos puede estar relacionado con un objeto ContextAggregator. Cuando una variable de contexto cambia su valor éste determina de qué manera deben reaccionarse las variables restantes. Una vez definido el comportamiento, solicita a ServiceCoordinator la ejecución de un determinado servicio.

En muchas oportunidades la actuación de ServiceCoordinator involucra la solicitud de un actuador (ej: abrir una compuerta, elevar temperatura); por lo tanto se puede hacer una distinción entre objetos Actuator y Sensor que aparecen en el modelo como subclases de IRPort.

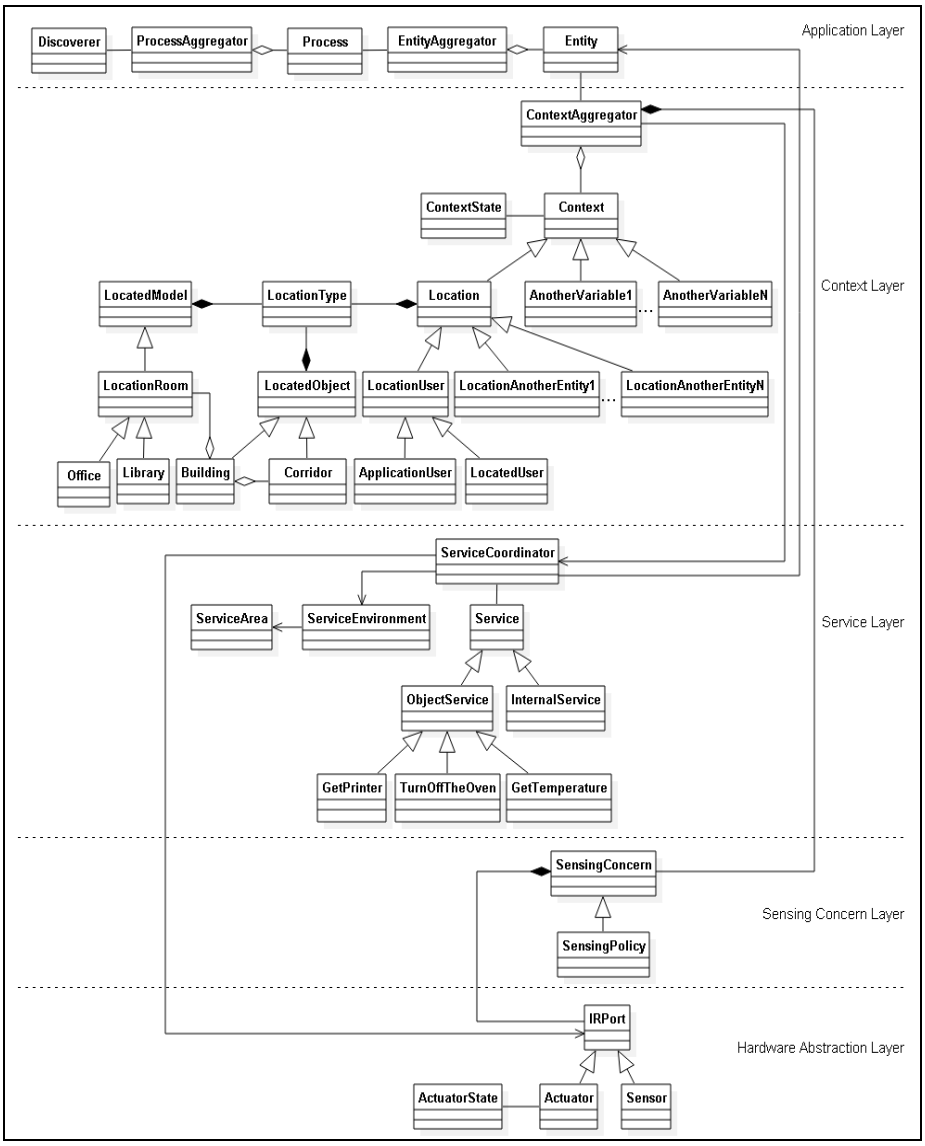


Fig. 1. Modelo MASCO

Cada clase que representa un objeto Actuator contiene operaciones que hacen referencia a los servicios que brinda el objeto real.

La clase Actuator representa la generalización de los actuadores y éstos en general accionan en base a sus estados, por esta razón se incorporó la clase ActuatorState, que en tiempo de implementación actúa como un patrón State [6] [7].

3 OCL (Object Constraint Language)

Un lenguaje gráfico de definición de modelos, como UML, proporciona una idea general e intuitiva, generalmente fácil de comprender por parte del usuario, de los conceptos del dominio de referencia que se está modelando y de las relaciones entre ellos. Sin embargo, estos lenguajes no son lo suficientemente expresivos como para definir toda la información relevante del dominio de referencia, estos aspectos hacen referencia a requisitos establecidos en éste y también tienen que ser definidos en el modelo ya que de lo contrario la definición del dominio sería incompleta.

Por este motivo, el uso de lenguajes gráficos de modelado se complementa con el uso de otros lenguajes, normalmente textuales aunque basados en un formalismo lógico, que permitan definir esta información de manera precisa y no ambigua. El lenguaje OCL (Object Constraint Language), lenguaje de restricción de objetos es el más conocido de ellos. OCL ha sido propuesto por el Object Management Group [8], y recientemente ha sido adoptado como estándar por el ISO/IEC [9].

Las características principales de OCL que se definen en [10] son: a) Es un lenguaje de definición de expresiones, a partir de las cuales pueden definirse condiciones que un modelo debe satisfacer; b) Es un lenguaje formal, en el sentido que define expresiones con una semántica precisa y no ambigua, y que por lo tanto pueden ser evaluadas mediante herramientas automáticas; c) Es un lenguaje tipado, ya que cada expresión corresponde a un tipo de datos determinado y solo se pueden combinar en OCL expresiones de tipos compatibles entre sí; d) Es un lenguaje de especificación y no de programación.

OCL puede ser utilizado tanto para definir aspectos estructurales (o estáticos) de un modelo como para definir aspectos dinámicos (o de comportamiento). En la parte estructural de un modelo, OCL se usa para especificar las restricciones de integridad que el diagrama de clases debería satisfacer y también para definir la información derivada de este diagrama. En la parte del comportamiento, OCL se usa para definir el efecto de la ejecución de una operación sobre la base de información (mediante expresiones que especifican las precondiciones y las postcondiciones de dicha operación) y para definir consultas sobre la base de información.

Este lenguaje es un lenguaje puro de especificación, por lo que las expresiones escritas en él no tienen efectos secundarios, es decir, no modifican el estado del modelo consultado [11]. Por lo tanto OCL es una extensión estándar a UML que permite realizar lo siguiente: [12]

- Al ser un lenguaje de consulta: especificar consultas para acceder a elementos de modelo y sus valores, y definir operaciones de consulta.
- Indicar restricciones en elementos de un modelo al definir reglas de negocio como restricciones en elementos de un modelo.

4 Planteamiento de Restricciones en MASCO

Las restricciones son planteadas en lenguaje natural, y a modo de introducirlas posteriormente en lenguaje OCL, se puede determinar las siguientes:

- **Restricciones del dominio**
  El modelo sólo puede ser utilizado por aplicaciones context-aware.
- **Restricciones del modelo**
  El modelo solo trabaja con objetos y con patrones de diseño
  El modelo permite trabajar con N variables de contexto
  El modelo permite trabajar con N entidades.
  El modelo permite trabajar con N procesos
  La clase Actuator existirá dependiendo del tipo de aplicación.

4.1 Restricciones en Hardware Abstraction Layer

El modelo trabaja con N variables de contexto, las cuales deben ser sensadas y de acuerdo a estos valores sensados se determina la intervención de un actuador, por lo cual es necesario cumplir con lo siguiente:

- Deben existir 1 a N sensores
- Deben existir 1 a N actuadores
- La actuación de los actuadores va a depender de los valores que sean sensados.

Los atributos y métodos son determinados de forma genérica para realizar la formalización de éste modelo y se muestran en la Fig. 2. Las restricciones de esta capa se presentan a continuación:

- **Clase IRPort**
  - *Atributos*
    - Los atributos en común del Sensor y actuador deben colocarse en esta clase.
  - *Métodos*
    - Los métodos en común del sensor y el actuador deben colocarse en esta clase.
- **Clase Sensor**
  - *Atributos*
    - Identificar el sensor del cual proviene el dato.
    - Identificar el tipo de dato sensado.
    - El valor sensado debe ser del tipo numérico real.
    - Deben existir 1 a N sensores que proveen los datos.
    - Toda identificación debe ser única.
    - La ubicación del sensor debe ser única.
  - *Métodos*
    - El valor leído desde el sensor no debe pertenecer al conjunto vacío.
- **Clase Actuator**
  - *Atributos*
    - Determinar el actuador al cual se envía la orden.
    - Pueden existir 1 a N actuadores.
    - Toda identificación debe ser única.
    - La ubicación del actuador debe ser única.

- *Métodos*
  - El cambio de estado se va a producir dependiendo del estado actual del actuador definido en StateActuator. Se utiliza el patrón diseño State que permite que un objeto modifique su comportamiento cada vez que cambie su estado interno [6]:
    - Si el estado del actuador no se corresponde con el estado actual del actuador la orden debe ser ejecutada.
    - Si el estado actual del actuador se corresponde con la orden de actuación, la aplicación debe anular la orden. Por ejemplo, si un motor está encendido no se puede solicitar que se lo encienda, en este caso la orden queda sin efecto.

- **Clase StateActuator**
  - *Atributos*
    - Representa el estado actual del actuador.
    - El estado debe ser booleano (por ejemplo, pueden encontrarse los estados: abierto/cerrado, encendido/apagado, activado/desactivado, entre otros).
  - *Métodos*
    - Realiza la consulta del estado actual del actuador.
- **Relación de Asociación**
  - Es necesario definir la relación de asociación entre los objetos Actuator y StateActuator: "Un estado corresponde a 1 o N actuadores" (por ejemplo, el estado "encendido" puede corresponder a distintos actuadores: encendida la ventilación, encendida la luz, entre otros).

Especificación con OCL. Luego de plantear las restricciones en lenguaje natural, se puede declarar el contexto especificando el elemento del modelo en el cual será definida la restricción. Para las invariantes la declaración del contexto será una clase. Para las precondiciones y postcondiciones el contexto será una clase seguida de la signatura de la operación [13], utilizando las clases representadas en la Fig. 2 se puede iniciar una especificación usando OCL de la siguiente forma:

- **Context IRPort**
  inv: self.NumIdentification >0
  inv: self.NumIdentification notEmpty()
  inv: self.NumIdentificationÚnique
  inv: self.Address notEmpty()
  inv: self.LocationUnique
  --a cada sensor o actuador le corresponde una única identificación
  inv: UniqueNumIdentification:IRPort.AllInstances( ) → is Unique (NumIdentification)
  --a cada sensor o actuador le corresponde una única ubicación
  inv: UniqueAddress:IRPort.AllInstances( ) → is Unique(Address)
- **Context Sensor**
  inv: self.SensingValue notEmpty()
  inv: self.SensingValue = real
  inv: self.VariableType = "Digital" or self.VariableType = "Analogic"
  inv: self.ReadValue() = real implies notEmpty()
- **Context Actuator**
  inv: self.StateActuator
  --Es el que implementa el patrón de diseño State

inv: self.ChangeState() = true implies self.State = false or self.ChangeState() = false implies self.State = true

- **Context StateActuator**

  inv: self State = true or self State = false

  --*Es el que revisa en qué estado se encuentra el actuador*

  Context.StateActuator :: CheckStatus()

  Pre: state notEmpty()

  Post: state = true or state = false

En la Fig. 2 se puede observar que las clases Sensor y Actuator heredan desde la clase abstracta IRPort, lo cual puede ser representado en la tabla 1 según lo propuesto en [14]:

Tabla 1. Herencia expresada en OCL de Hardware Abstraction Layer.

Clase Abstracta	Subclases
IRPort:	**Sensor:**
Context IRPort	*Context Sensor*
Inv: self.oclIsKindOf (IRPort) = true	Inv: self.oclIsKindOf (IRPort) = true
Inv: self.oclIsTypeOf (IRPort) = true	Inv: self.oclIsTypeOf (IRPort) = false
Inv: self.oclIsTypeOf (Sensor) = false	Inv: self.oclIsTypeOf (Sensor) = true
Inv: self.oclIsKindOf (Sensor) = false	Inv: self.oclIsKindOf (Sensor) = true
Inv: self.oclIsTypeOf (Actuator) = false	Inv: self.oclIsTypeOf (Actuator) = false
Inv: self.oclIsKindOf (Actuator) = false	Inv: self.oclIsKindOf (Actuator) = false
	Actuator:
	Context Actuator
	Inv: self.oclIsKindOf (IRPort) = true
	Inv: self.oclIsTypeOf (IRPort) = false
	Inv: self.oclIsTypeOf (Actuator) = true
	Inv: self.oclIsKindOf (Actuator) = true
	Inv: self.oclIsTypeOf (Sensor) = false
	Inv: self.oclIsKindOf (Sensor) = false

La asociación "*corresponds to*" ("corresponde a") entre las clases StateActuator y Actuator se expresa de la siguiente forma:

- **Context StateActuador**

  Inv: self.State= true or self.State= false implies self.Set → notEmpty()

  Self.Set → collect (State)

4.2 Restricciones en Sensing Concern Layer

Esta capa recibe los valores sensados en la Hardware Abstraction Layer y los transforma, si es necesario, en datos con los cuales pueda trabajar la aplicación, por lo que las clases que se muestran en la Fig. 2 deben estar presentes:

- **Clase SensingConcern**
  - *Atributos*
    - Es necesario conocer el Identificador del sensor que está siendo leído.
- **Clase SensingPolicy**

o *Métodos*

- Los valores sensados llegan a esta capa a medida que son leídos por los sensores, en tiempo de implementación se utiliza el patrón de diseño Observer. [6].
- El valor sensado debe ser transformado si es necesario.
- La clase ContextAggregator en Context Layer debe ser notificada al finalizar la transformación (si fuera necesario) de los valores sensados.
- En caso de que el valor sensado necesite ser transformado debe aplicarse la política de conversión más adecuada. Para la situación en que sean leídos más de una variable de contexto o de ubicación, en tiempo de implementación es necesario utilizar el patrón de diseño Strategy [6] que determina cual es la política de transformación adecuada para cada uno de los datos sensados que necesite ser transformado.

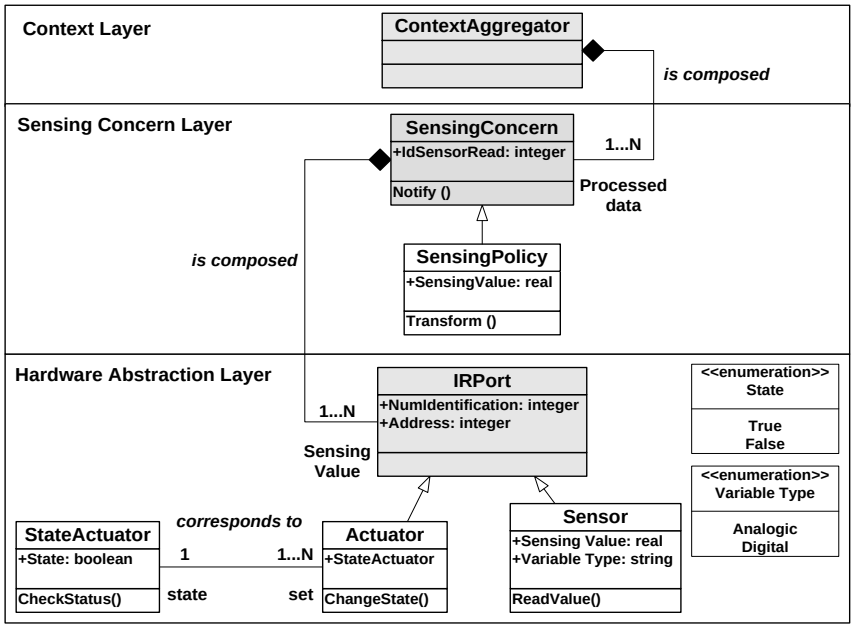


Fig. 2. MASCO con atributos, métodos genéricos y relaciones entre capas..

Especificación con OCL. Utilizando las clases representadas en la Fig. 2 se puede iniciar una especificación usando OCL de la siguiente forma:

- **Context SensingConcern**
  Inv: self.IdSensorRead notEmpty()
  Inv: self.IdSensorRead = integer
  Inv: self.Notify()
  --*Con este método se notifica a Context Layer los datos procesados*
- **Context SensingPolicy**
  Inv: self.SensingValue notEmpty()
  Inv: self.SensingValue = real

Inv: self.Transform()
--Debe existir un valor a transformar
Pre: = SensingValue notEmpty()
--El valor sensado es transformado, el cual deriva en un dato procesado
Post: Context.SensingConcern :: SensingValue: real
Derive
Transformer(SensingValue) → processed Data

En la Fig. 2 se puede observar que la clase Sensing Policy hereda desde la clase Sensing Concern, lo cual puede ser representado en la tabla 2 según lo propuesto en [14]:

Tabla 2. Herencia expresada en OCL de Sensing Concern Layer.

Clase Abstracta	Subclases
SensingConcern:	**SensingPolicy:**
Context SensingConcern	*Context SensingPolicy*
Inv: self.oclIsKindOf (SensingConcern) =true	Inv: self.oclIsKindOf (SensingConcern) =true
Inv: self.oclIsTypeOf (SensingConcern)=true	Inv: self.oclIsTypeOf(SensingConcern)=false
Inv: self.oclIsTypeOf (SensingPolicy) = false	Inv: self.oclIsTypeOf (SensingPolicy) = true
Inv: self.oclIsKindOf (SensingPolicy) = false	Inv: self.oclIsKindOf (SensingPolicy) = true

4.3 Especificación con OCL de las relaciones entre las capas del modelo

Para representar las relaciones entre las capas del modelo se plantearon asociaciones cualificadas [12] y [14]. En la Fig. 2 se observan las siguientes composiciones:

- La clase SensingConcern en Sensing Concern Layer está compuesta por uno o varios valores que son enviados desde la clase IRPort en Hardware Abstraction Layer.
- La clase ContextAggregator en Context Layer está compuesta por uno o varios datos procesados que son notificados desde la clase SensingConcern en Sensing Concern Layer, para éste envío se emplea el método Notity().

Las restricciones expresadas en OCL son las siguientes:

- **Context SensingConcern**
  --Deben existir uno o varios valores sensados para que pueda ser procesado
  Inv: self.SensingValue [N].NumIdentification notEmpty()
- **Context ContextAggregator**
  --Los datos procesados deben ser notificados en Context Layer
  Inv: self.Processed data [N].SensingValue → Notify()

5 Conclusiones

La formalización de MASCO utilizando OCL, permite obtener un modelo libre de malas interpretaciones, con una arquitectura definida, con atributos de calidad determinados, el cual ya fue probado y validado en distintos ámbitos de aplicación, para ser usado en el desarrollo de aplicaciones Context Aware.

Se comenzó éste trabajo definiendo para el modelo solo los aspectos estructurales (o estáticos) de las capas Hardware Abstraction y Sensing Concern y las asociaciones entre ellas, continuando en futuros trabajos con la formalización completa de MASCO, considerando que además de los aspectos estructurales serán incorporados los aspectos de comportamiento (consultas).

6 Referencias

1. Quincoces, V.E., Gálvez, M.P., Cáceres, N.R., Vega, A.A., Ramos, H.O.: Extensión de un modelo en capas que provee servicios para aplicaciones sensibles al contexto. En: Investigaciones en Facultades de Ingeniería del NOA, Vol I, pp. 35-40, Cap. IV. EUNSa, Argentina (2009)
2. Gálvez, M.P., Quincoces, V.E., Cáceres, N.R., Vega, A.A.: Refinamiento de un Modelo en Capas que Provee Servicios de Ubicación para Aplicaciones Sensibles al Contexto. En: III Congreso Internacional de Telecomunicaciones, Tecnologías de la Información y las Comunicaciones, Quito (2010)
3. Gálvez, M.P., Brouchy C., González O., Cáceres, N.R., Quincoces, V.E.: Modelo que provee servicios para aplicaciones sensibles al contexto (MASCO): Interacción entre entidades. En: Investigaciones en Facultades de Ingeniería del NOA, pp. 1103-1109. Científica Universitaria, UNCa, Argentina (2011)
4. Gálvez, M.P., Cáceres, N.R., Brouchy C., Velázquez C.E., González O.M, Guzmán A.N., Romero, N.D., Quincoces, V.E.: Modelo que provee aplicaciones sensibles al contexto (MASCO): validación e inicio de evaluación. En: Investigaciones en Facultades de Ingeniería del NOA, pp. 146. Grupo Loza Impresiones S.R.L., UNT, Tucumán (2012)
5. Gálvez, M.P., Cáceres, N.R, Velázquez, C. E., Guzmán, A. N.: Atributos de calidad del modelo para aplicaciones sensibles al contexto MASCO. En: 5to Simposio Internacional de Investigación: "Interdisciplinariedad, Multidisciplinariedad y/o transdisciplinariedad: en la búsqueda de respuestas desde las experiencias de investicación". Universidad Católica de Santiago del Estero, DASS, Jujuy (2013)
6. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Elements of Reusable OO Software. Addison Wesley, USA (1995)
7. Velázquez, C. E., Guzmán, A. N., Gálvez, M.P., Cáceres, N.R.: Modelo para aplicaciones sensibles al contexto (MASCO): Un caso de estudio para validación. En: XIX Congreso Argentino de Ciencia de la Computación, Mar del Plata (2013)
8. Object Constraint Language (OCL), http:// www.omg.org/spec/OCL/
9. ISO/ IEC 19505-2:2012 – OMG UML superstructure 2.4.1, http/www.iso.org/
10. Garcia Molina, J., García Rubio, F., Pelechano, V., Vallecito, A., Vara, J. M., Vicente-Chicote, C.: Desarrollo de software dirigido por Modelos: conceptos, métodos y herramientas. Alfaomega, Rama, Mexico (2014)
11. UML 2.0 Superstructure Specification, 2003, http://www.omg.org/docs/ptc/03-08-02.pdf
12. Arlow, J., Neustadt, I.: UML 2. Anaya, Madrid (2005)
13. Vidal, D. E., Vidal, S. C.: Como reforzar Diagramas de Clases UML aplicando OCL y Object-Z: un caso práctico, http://eventos.spc.org.pe/jpc2007/MyReview/FILES/p23.pdf
14. Warmer, J. B., Kepple, A. G.: The Object Constraint Languaje: Getting Your Models Ready for MDA. Pearson Education, USA (2003)

Usabilidad con AOP: Comparación de Enfoques y Herramientas.

Sandra Casas[1], Natalia Trejo[1], Juan Enriquez[1] y Roberto Farias[1],
,

[1] GISP. Instituto de Tecnología Aplicada, Universidad Nacional de la Patagonia Austral,
Campus Universitario, Piloto Lero Riversa S/N,
9400 Río Gallego, Argentina
{scasas, ntrejo, jenriquez, rfarias}@unpa.edu.ar

Abstract. Este trabajo realiza un recorrido por distintas propuestas que usan técnicas de programación orientadas a aspectos, para la captura automática de datos para la evaluación de usabilidad. Se propone un conjunto inicial de criterios de análisis (técnicos y no técnicos) que luego son usados para plantear el estudio comparativo. El objetivo es orientar a los potenciales adoptantes de los enfoques en cuanto a la variación de las características, y a los investigadores de estas áreas respecto de que ejes requieren mayor profundización o abordaje.

Keywords: Usabilidad; Programación Orientada a Aspectos, Logging; AspectJ

1 Introducción

La evaluación de la usabilidad [1] consiste en realizar pruebas para obtener información que permitan observar e identificar debilidades relacionadas al uso de las aplicaciones de software. Las cuatro formas básicas de evaluación son: automática (se calculan las métricas durante la ejecución de la aplicación), empírica (la usabilidad es evaluada testeando la aplicación con usuarios reales), formal (se usan modelos formales y fórmulas para el cálculo de medidas de usabilidad), e informal (se basan en reglas generales y la habilidad y experiencia de los evaluadores). Este proceso de evaluación implica varias actividades en función del método empleado, pero en general se agrupan en 3 pasos básicos [2]: Captura de datos, Análisis y Crítica.

Cualquiera sea la forma de evaluación que se adopte o combinación de ellas, la recolección de los datos es una tarea fundamental. En lo que respecta a la captura de datos automatizada se ha caracterizado por usar archivos de logs. Sus ventajas son varias, como: a) soportan la ejecución de las pruebas en laboratorios de usabilidad o en el ambiente natural del usuario; b) son transparentes para los usuarios; c) permiten con facilidad obtener información de múltiples usuarios; d) permiten detectar errores comunes y actuales; etc. Los logs aportan gran cantidad de datos a los evaluadores de usabilidad, además de poder contrastar comportamientos entre diferentes usuarios. Sin embargo, tienen desventajas, al emplear los logs generados por servidores webs o sistemas operativos, se obtiene una gran cantidad de información que resulta

irrelevante para la evaluación de usabilidad, son incapaces de registrar información específica. Por otro lado, añadir en las aplicaciones el código que realiza el logging de las actividades / tareas de usabilidad que interesan al evaluador de usabilidad, implica invadir la estructura interna de las aplicaciones agregando cientos de líneas de código, cuya remoción posterior es muy compleja.

A partir de la aparición de las técnicas de separación de concerns [3], y en particular de la Programación Orientada a Aspectos (AOP) [4], es posible diseñar módulos de código (aspectos) que realicen el seguimiento y log sin invadir y/o alterar la estructura interna de los sistemas. La posibilidad de realizar la recolección de los datos en forma dinámica, transparente para el usuario y sin invadir el código, ha provocado interés y varios enfoques se han presentado que aplican AOP a la evaluación de usabilidad.

Este trabajo realiza un recorrido por las distintas propuestas que usan la AOP para la captura automática de datos para la evaluación de usabilidad. Se propone un conjunto inicial de criterios de análisis (técnicos y no técnicos) que son empleados en el estudio comparativo. El objetivo es orientar a los potenciales adoptantes de los enfoques en cuanto a la variación de las características, y a los investigadores de estas áreas respecto de los ejes que requieren mayor profundización o abordaje.

Este trabajo se estructura de la siguiente manera, en la Sección 2 se presenta brevemente los conceptos básicos de la AOP, en la Sección 3 se presenta un resumen de los enfoques que usan AOP en la evaluación de la usabilidad, en la Sección 4 se presenta el conjunto de criterios de comparación, en la Sección 5 se presentan los resultados del análisis y comparación de los enfoques según los criterios escogidos y, en la Sección 6 se presenta la discusión y conclusiones.

2 Programación Orientada a Aspectos (AOP)

La AOP [4] es una técnica de programación que surgió hace una década y su objetivo es encapsular la funcionalidad transversal que genera código mezclado y diseminado, generalmente relacionado a requerimientos no funcionales, técnicos y/o de calidad. Ejemplos típicos de funcionalidad transversal son logging, traza, seguridad y persistencia. Con el fin de encapsular e implementar este tipo de funcionalidad la AOP introduce cuatro nuevas abstracciones: join-point, pointcut, advice y aspectos:

- Un join-point es un punto bien definido en la ejecución de un programa, por ejemplo la invocación a un método.
- Un pointcut es una expresión que especifica las condiciones de ejecución de un segmento de código (advice), es decir, determina el conjunto de join-points que serán interceptados por el aspecto, y expone algunos de los valores del contexto de los mismos.
- Un advice es un segmento de código similar a un método que se ejecuta en cada join-points especificado en los pointcuts.
- Un aspecto es un tipo transversal que encapsula pointcuts, advices y características transversales estáticas. Un aspecto es la unidad de modularización de la AOP.

Los aspectos se integran (componen) en el sistema por medio de un proceso especial llamado tejedor [5]. Actualmente hay extensiones de lenguajes

convencionales que soportan AOP para las aplicaciones implementadas con esos lenguajes de programación convencionales. AspectJ [6] es uno de los soportes AOP para Java.

3 Resumen de Enfoques

En esta Sección se describen brevemente los trabajos presentados que usan AOP en la evaluación de la usabilidad en los últimos años.

A. Framework de Diseño

Tarta y Moldovan [7] proponen un diseño de aspectos que soporta la evaluación automática de la usabilidad en aplicaciones de escritorio. Se plantea que mediante una jerarquía de aspectos se pueden reusar pointcuts que compartan los mismos puntos de entrada y salida (join-points). Los aspectos derivados definen nuevos pointcuts para la intercepción de errores, completitud de tareas, captura de pantallas y adquisición de datos por medio de cuestionarios. Los aspectos concretos por medio de los advices realizan el log de los datos. La estructura es específica de cada aplicación en la que se use. Se presentan simples diagramas y código de ejemplo en AspectJ.

Shekh y Tyerman [8] presentan el desarrollo de un framework AOP para la evaluación de la usabilidad con AOP. Las autoras registran los eventos de IU, por ejemplo los eventos del mouse. Respecto de las características del framework y diseño de aspectos no se brindan detalles, pero se indica que han seguido las recomendaciones de [7]. El framework esta desarrollado en AspectJ. Se presentan resultados de experimentos controlados y pautados en un laboratorio sobre la aplicación software ACIE Player.

B. Trayectoria de Eventos de Interface

Holzinger y otros [9] proponen la implementación de aspectos para seguir la trayectoria de algunos eventos de interface (entrada del teclado, menues y acciones arrastrar y soltar). Se plantea usar OBjective-C, agregando los aspectos a la jerarquía de objetos usando la tecnología "method swizzling", y extendiendo de clases. El diseño se centra en un aspecto que realiza el logging de manera centralizada y tres aspectos son necesarios para identificar los eventos de interface mencionados.

C. Generación automática de aspectos

Kronbauer y otros [10] [11] presentan una metodología para generar automáticamente los aspectos que realizan la captura de datos de usabilidad en aplicaciones móviles. La AOP se emplea en dos instancias y funciones bien definidas del enfoque. El código de los aspectos es generado mediante las especificaciones y acciones realizados en dos unidades (pasos) para lo que se ofrecen herramientas. La primer herramienta permite "la descripción automática de tareas", el ingeniero de usabilidad ejecuta esta herramienta en paralelo con la aplicación móvil a evaluar. Por medio de la ejecución en simultáneo se capturan las acciones de la interacción del ingeniero de usabilidad, que configuran una tarea determinada. Un aspecto va registrando los métodos en el camino de la tarea, los cuales se almacenan en un archivo XML. Además se definen

los parámetros de evaluación de usabilidad para dicha tarea. Luego con el archivo XML generado en el paso anterior (corresponde a la definición de una tarea), el ingeniero de usabilidad, a través de la herramienta "creación de aspectos de usabilidad" define que métricas se aplicarán en la evaluación, a partir de una biblioteca de métricas disponible. De esta segunda especificación se generan los aspectos para la evaluación de la usabilidad en AspectJ. La persistencia de los datos se trata en los aspectos. Los aspectos se usan para capturar datos para métricas de usabilidad convencionales (eficiencia, efectividad y satisfacción), también se puede recolectar información del contexto aportada por sensores (luminosidad-desplazamiento-posición) y de los perfiles de los usuarios (edad-sexo-estudios-etc.). Se presentan pruebas realizadas sobre aplicaciones móviles reales.

D. Toolkit para experimentos de usabilidad

Lettner y Holzmann [12] presentan un conjunto de herramientas para realizar experimentos relacionados al campo de la usabilidad sobre aplicaciones móviles. La propuesta consiste en posibilitar la conexión en forma transparente a los datos de interacción de usuario y calcular diversas métricas de bajo nivel durante los estudios de campo sin supervisión a los usuarios finales. El Toolkit se divide en tres partes principales: (i) el framework móvil para Android, (ii) un servidor back-end basado en Google App Engine y (iii) un servidor front-end basado en Google Web Toolkit.

La AOP se usa en el framework móvil. Como se describe en [13] los métodos del ciclo de vida de una aplicación Android, onCreate(), onResume(), onPause() y onDestroy() son interceptados por los aspectos para capturar el comportamiento de las actividades (Activity). Los datos se registran a nivel actividad y refieren esencialmente al comportamiento de navegación. Las métricas de bajo nivel que se recogen corresponden a errores de navegación, tiempo de sesión, llamadas a pantallas, clicks de botones, etc., las cuales son segmentadas y agregadas por grupo usuarios o usuarios individuales. Estas métricas se combinan con estadística para derivar métricas relacionadas a la usabilidad.

E. Trazas de Eventos

Traza de Eventos MVC

Yonglei [14] propone AOP para capturar automáticamente eventos de la interface de usuario en aplicaciones con arquitectura MVC (modelo-vista-controlador). Se propone una jerarquía de aspectos, en la cual se logra reusar solo el método que realiza el reporte (hora-fecha y evento ocurrido), los pointcuts y advices deben ser redefinidos en cada caso y aplicación, por ejemplo actualización de observer, manejadores y notificadores de eventos, cuadros de diálogo. Se presentan códigos de ejemplo en AspectJ y un caso de estudio simple.

Traza de Eventos IU

Un esquema muy similar al anterior, es el que presenta Yonglei [15], ya que este enfoque se basa en técnicas OA para ejecutar las trazas de los eventos de interface de usuario y recolectar información contextual, en aplicaciones de escritorio (WIMP). Propone una jerarquía de aspectos, la cual consiste básicamente en un método que contiene la lógica que permite hacer el reporte, los sub-aspectos definen los eventos a interceptar en los pointcuts y los advice que invocan al método que ejecuta el reporte. Se presentan simples códigos de ejemplo en AspectJ.

F. Instrumentación Interactiva

Bateman y otros [16] presentan el enfoque "Instrumentación Interactiva de Usabilidad" (IIU). Los evaluadores de usabilidad especifican qué acciones serán registradas (log) interactuando con los elementos de la interfaz de la aplicación objeto de la evaluación, eliminando la necesidad de apoyo adicional de programación. Esta primera actividad se apoya en AOP, y permite la instrumentación a través de la interacción directa con la aplicación a ser instrumentada; además de su simplicidad, especificación interactiva también proporciona un mecanismo natural para decidir qué elementos de un sistema están relacionados con tareas particulares y problemas de usabilidad. La herramienta UMARA da soporte al enfoque que ha sido probado en los softwares "PDF Split and Merge" y "MegaMek".

4. Criterios de Comparación

A continuación se presenta el conjunto de criterios utilizados en la próxima Sección para comparar los enfoques mencionados previamente. La compilación de este conjunto está centrada en la variabilidad de las diferentes características (técnicas y no técnicas). Como tal, el objetivo es de obtener un marco de clasificación y comparación flexible que ayude a los potenciales usuarios de los enfoques para seleccionar el más adecuado, y que ayude a otros investigadores a razonar respecto de las diferencias entre sus propios trabajos y las características existentes de otros. El conjunto de criterios permite contrastar las diferentes restricciones de cada enfoque e impone las fortalezas y debilidades a nivel de automatización y flexibilidad. Estos criterios forman un marco comparativo inicial que puede evolucionar siguiendo los nuevos desarrollos en los campos de usabilidad y/o AOP.

Plataforma. ¿A qué tipo de plataforma de software el enfoque está dirigido? Se distinguen tres tipos de plataforma: escritorio, móvil y web.

Nivel de abstracción. ¿A qué nivel de abstracción es posible realizar la captura de los datos de usabilidad? Se distinguen tres niveles: bajo, se interceptan eventos de manera general sin posibilidad de que sean acotados a una tarea del usuario específica, alto: los eventos observados se acotan a una tarea específica del usuario, y medio: algunos eventos pueden ser acotados a nivel tarea y otros no.

Soporte a métricas. ¿Los datos capturados aportan datos para calcular que clase de métrica? Usabilidad (efectividad, eficiencia-satisfacción), contexto (lectura de sensores o captura de pantallas), perfil de usuario (edad-sexo-formación-etc.), patrones (traza de eventos-navegación).

Uso de Aspectos. ¿Qué responsabilidad/función cumplen los aspectos en el enfoque? Se distinguen las siguientes:

-Identificación interactiva de tareas/elementos de IU. En un estadio previo a la recolección de datos, los aspectos se emplean para identificar y/o seleccionar los eventos que en ejecuciones posteriores serán observados.

- Traza de IU. Los aspectos interceptan determinados elementos de IU. Los elementos que suelen ser monitoreados son: eventos producidos por la acción del usuario como

presionar botones, movimiento de ratón, etc. y/o la ejecución de métodos de componentes gráficos como mostrar los cuadros de dialogo.
- Logging de datos. Esta actividad es realizada en los advices y/o métodos de los aspectos y consiste en registrar datos, que posteriormente se emplean para calcular métricas.
- Traza de tareas. Los aspectos por medio de pointcuts interceptan determinados puntos de ejecución (métodos) que configuran el inicio y fin de una tarea de usuario.
- Incorporación de cuestionarios: Los aspectos incorporan cuestionarios para obtener información subjetiva principalmente relacionada al factor satisfacción o perfil del usuario.
- Contexto: Los aspectos registran información relacionada al contexto (datos obtenidos de los sensores del dispositivo o captura de pantalla)

Aporte Empírico. ¿Cuál es la contribución empírica del enfoque? Una herramienta o un framework de diseño. ¿Cuál ha sido la validación empírica del enfoque? El enfoque se ha aplicado a sistemas reales o ejemplos pequeños.

Precondiciones. ¿Qué requisitos impone el enfoque para su aplicación y/o adopción? Conocimientos de la estructura interna de la aplicación cuya usabilidad será evaluada (diseño, arquitectura, implementación), conocimiento de AOP (conceptos y/o lenguaje OA), características de la aplicación a evaluar (lenguaje/arquitectura/ plataforma).

Usuarios Final. ¿Quién es el usuario final del enfoque? los desarrolladores de software o los ingenieros de usabilidad.

5. Resultados

Basados en los criterios introducidos en la Sección anterior comparamos los diferentes enfoques que usan AOP para capturar los datos para la evaluación de usabilidad. En la Tabla 1, se presenta el título abreviado del enfoque que será usado posteriormente.

Tabla 1.

Título Abreviado	**Identificador**
Framework de Diseño	A
Trayectoria de Eventos de Interface	B
Generación automática de aspectos	C
Toolkit para experimentos de usabilidad	D
Traza de Eventos MVC	E
Traza de Eventos IU	E
Instrumentación Interactiva	F

La Tabla 2 presenta una síntesis del análisis y comparación de los enfoques según todos los criterios definidos.

TABLA 2. ANALISIS Y COMPARACION DE ENFOQUES SEGUN LOS CRITERIOS PROPUESTOS

	Framework de Diseño (A)	**Trayectoria de Eventos de Interfa-ce (B)**	**Generación Automática de Aspectos (C)**	**Toolkit para experimentos de Usabili-dad (D)**	**Traza de Eventos MVC (E)**	**Traza de Eventos IU (E)**	**Instrumentac ión Interac-tiva (F)**
Plataforma	Escritorio	Escritorio	Móvil	Móvil	Escritorio	Escritorio	Escritorio
Abstracción	Media	Baja	Alta	Media	Baja	Baja	Baja
Métricas	Usabilidad Contexto	Patrones	Usabilidad Contexto Perfiles de Usuario	Patrones	Patrones	Patrones	Patrones
Uso de Aspectos	Traza de IU Logging de datos	Traza de IU Logging de datos	Identificación interactiva de tareas y parametrización Traza de tareas Logging de datos Incorporación de cuestionarios Contexto	Traza de tareas(activida des) Logging de datos	Traza de elementos de IU Logging de datos	Traza de elementos de IU Logging de datos	Identificación interactiva de elementos de IU. Traza de elementos de IU Logging de datos
Propuesta	Jerarquía de Aspectos	Indicaciones de desarro-llo y configura-ción.	Herramientas Automatic Task Description y Creation Usability Aspect	Toolkit - Framework	Jerarquía de Aspectos	Jerarquía de Aspectos Herramienta Prueba de conceptos	Herramienta UMARA
Validación Empírica	Dos aplicaciones de ejemplo	Una aplicación de ejemplo	Gingerbread Shuffle Mileage Cubed	ScotDroid	AccountMa nager (Deytel)	Una aplicación de ejemplo	PDF Split and Merge MegaMek
Precondicio-nes	Conocimient o del diseño /arquitectura de la aplicación Conocimien-tos de AOP Aplicaciones Java-AspectJ	Aplicacio-nes ObjectC Conocimien to del diseño /arquitectura de la aplicación Conocimien tos de AOP	Aplicaciones Java (Android) –	Conocimiento del diseño/arquite ctura de la aplicación Conocimiento s de AOP Aplicaciones Java (Android)– AspectJ.	Conocimie nto del diseño/arqu itectura de la aplicación Conocimie ntos de AOP Aplicacion es Java-AspectJ Arquitectur a MVC	Conocimien to del diseño/arqui tectura de la aplicación Conocimien tos de AOP Aplicacione s Java-AspectJ	Aplicaciones Java-AspectJ
Usuario Final	Desarrolla-dor de Software	Desarrolla-dor de Software	Ingeniero de Usabilidad	Desarrollador de Software	Desarro-llador de Software	Desarrolla-dor de Software	Ingeniero de Usabilidad

6. Discusión y Conclusiones

Trabajos Relacionados. Tarby y otros [17] comparan las AOP y los agentes interactivos para la evaluación temprana de la usabilidad de aplicaciones interactivas usando trazas. Ambos enfoques requieren ser incluidos en los estudios preliminares y de factibilidad del proyecto, especificación y codificación especifica. AOP no requiere diseño arquitectónico, a diferencias de los agentes dado que son parte de tales estructuras. Es decir AOP permite mantener intacto el código inicial.

Existen varios estudios comparativos y estados del arte, sobre los métodos de la evaluación de usabilidad, algunos incluso presentan taxonomías [2][18][19][20], la diferencia con este trabajo es que parten de alguna segmentación que en este trabajo forma parte de algún criterio de comparación. Por ejemplo, cuando se clasifican aplicaciones WIMP (escritorio) o cuando se analizan los métodos de captura de eventos. Asimismo no se toma como punto de partida la técnica de implementación como es el caso de este trabajo, que se focaliza en el uso de AOP.

Discusión. Los ejes de la discusión se plantean sobre los criterios de comparación y los aspectos más relevantes que del estudio surgen:

Plataforma. Llamativamente no se han encontrado en la literatura enfoques que usen AOP para la evaluación de la usabilidad de aplicaciones WEB.

Nivel de Abstracción. De los 7 trabajos analizados sólo uno proporciona medios para definir las tareas de usuario y asociar a estas los datos capturados. En los enfoques que han calificado con un nivel de abstracción "medio" se logra en forma restringida. Así el Framework de Diseño (A), define tareas solo para analizar ciertos eventos y el Toolkit para experimentos de usabilidad (D), se emplea el concepto de "actividad", que en realidad esta mas asociado a las características estructurales de las aplicaciones Android, que a tareas de usuario. También se observa que los datos recolectados por aspectos en forma general (nivel de abstracción bajo), luego son de alguna forma asociados a tareas, como ocurre en Traza de Eventos MVC (E). La evaluación de la usabilidad requiere la posibilidad de vincular los datos/eventos obtenidos con la intención del usuario, y para ello es necesario contextualizarlos en el marco de una tarea. De otra forma, disminuye el significado y las posibilidades de la interpretación de los resultados. La traza/captura de eventos proporciona información de bajo nivel que puede ser usada en todo caso, para definir pruebas posteriores.

Soporte a métricas. Relacionada a la observación anterior, la usabilidad es débilmente evaluada ya que no es posible obtener datos para evaluar la efectividad, por ejemplo al no ser posible medir la completitud de las tareas. Por otro lado, el soporte para capturar y medir información contextual y de perfil de usuario, contribuye positivamente a comprender las acciones del usuario.

Uso de aspectos. El uso de aspectos se presenta como muy versátil, ya que pueden ser usados tanto para las funciones básicas de la captura de datos (traza y logging), y también pueden realizar otras funciones como posibilitar que el usuario identifique las tareas/eventos en forma interactiva o bien pueda calcular otras métricas.

Aporte empírico. Surge una diferencia muy notable al contrastar la validación empírica de los distintos aportes, lo que permite distinguir los diferentes niveles de madurez de las propuestas y posibilidades de adopción.

Precondiciones. Las precondiciones más fuertes están dadas para aquellos enfoques cuyo usuario final es el desarrollador de software, dado que debe conocer la aplicación a evaluar y AOP. En 6 de 7 casos se presenta una limitación importante respecto del lenguaje de la aplicación a evaluar (Java) ya que se ha utilizado AspectJ. AspectJ es una herramienta de propósito general, madura y consolidada en estos días, sin embargo es necesario abarcar otros lenguajes OA, para que los enfoques se puedan aplicar a software codificado en lenguajes que no sean Java.

Usuario Final. Los enfoques que proporcionan una herramienta que permite al Ingeniero de Usabilidad ser independiente para el desarrollo de las pruebas de usabilidad, en contra partida son menos flexibles, ante la necesidad de implementar cambios mínimos. Toda métrica y/o dato no soportado por la herramienta requiere que el generador sea modificado.

Conclusiones. En este trabajo se ha definido un conjunto de criterios de comparación de enfoques que usan AOP para la evaluación de la usabilidad. Estos pueden ser ampliados para completar la propuesta, también para plantear marcos de clasificación/categorización y/o para conformar un benchmark.

Los criterios de comparación se han aplicado a 7 enfoques, de este proceso emergen algunas conclusiones y ejes de trabajo futuro:

a) La captura de datos para la evaluación de la usabilidad con AOP en aplicaciones web es un tema cuyo tratamiento está pendiente.

b) Se deben abordar mecanismos para dar mayor y mejor soporte a la medición de datos/eventos asociados a tareas de usuario. Una hipótesis de las limitaciones actuales podría ser el uso de AspectJ, que es un lenguaje de propósito general, por lo que no dispone de mecanismos apropiados para el seguimiento de una tarea. Una posibilidad sería explorar el uso de lenguajes de aspectos de dominio específico (DSAL) que brinde soporte específico a la usabilidad.

c) La AOP es muy versátil en cuanto a las posibilidades que brinda a la evaluación de la usabilidad, ha demostrado capacidad para registrar diversos tipos de datos a diferentes clases de métricas, o bien ha posibilitado el proceso de identificación de tareas/eventos a evaluar. Los posibles usos de los aspectos no parecen estar agotados en estas propuestas y mucho menos aún el uso de otras técnicas de Separación de Concerns, como podría ser la aplicación de la programación orientada a features (FOP).

d) En cuanto al aporte empírico es necesario definir un benchmark común, basado en aplicaciones reales y de ser posible con usuarios reales, lo que facilitará el análisis de cada trabajo.

Referencias

1. ISO 9241-11. Ergonomic requirements for office work with visual display terminals. ISO, 1998.
2. Ivory, M. y Hearst, M., The State of the Art in Automating Usability Evaluation of User Interfaces. ACM Computing Surveys, vol. 33, no. 4, 2001.
3. Hürsch, W. y Lopes, C., Separation of Concerns. Northeastern University, TR NU-CCS-95-03, USA, 1995.

4. Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C. y Loingtier, J. Aspect-oriented Programming. Proceedings of the European Conference on Object-Oriented Programming (ECOOP), 1.997, LNCS 1241, Springer-Verlag
5. Piveta, E. y Zancanela, L., Aspect Weaving Strategies. Journal of Universal Computer Science, vol.9, no. 8, 2003.
6. Kiczales, G., Hilsdale, E., Hugunin, J., Kersten, M., Palm, L. y Griswold, W., An overview of AspectJ. Proceedings of the 15th European Conference on Object-Oriented Programming (ECOOP). 2001.
7. Tarta, A. y Moldovan, G., Automatic Usability Evaluation Using AOP. Automation, Quality and Testing, Robotics, IEEE International Conference, vol. 2, 2006.
8. Shekh A. y Tyerman, S., An Aspect-Oriented Framework for Event Capture and Usability Evaluation. Communications in Computer and Information Science: Evaluation of Novel Approaches to Software Engineering, vol. 69, Springer, pp. 107-119, 2010.
9. Holzinger, A., Brugger, M. y Slany, W., Applying Aspect Oriented Programming in Usability Engineering Processes - On the Example of Tracking Usage Information for Remote Usability Testing. Proceedings of the 8th. International Conference on electronic Business and Telecommunications. España pp. 53-56, 2001
10. Kronbauer, A. y Santos, C., Um modelo de avaliação da usabilidade baseado na captura automática de dados de interação do usuário em ambientes reais. Proceedings of the 10th Brazilian Symposium on on Human Factors in Computing Systems y 5th Latin American Conference on Human-Computer Interaction – pp. 114-123, 2011.
11. Kronbauer, A., Santos, C. y Vieira, V., Um Estudo Experimental de Avaliação da Experiência dos Usuários de Aplicativos Móveis a partir da Captura Automática dos Dados Contextuais e de Interação. Braziliam Symposium on Human Factors in Computing Systems (IHC). Brazil, 2012
12. Lettner, F. y Holzmann, C., Automated and unsupervised user interaction logging as basis for usability evaluation of mobile applications. 10th International Conference on Advances in Mobile Computing & Multimedia (MoMM), Indonesia, pp. 118-127, 2012.
13. Lettner F. y Holzmann, C., Sensing mobile phone interaction in the field. Proceedings of the 4th International Workshop on Sensor Networks and Ambient Intelligence, 2012.
14. Yonglei T., Automated Data Collection for Usability Evaluation in Early Stages of Application Development. 7th WSEAS in ACACOS 08, China.2008
15. Yonglei,T., Aspect-Oriented Instrumentation for Capturing Task-Based Event Traces. International Journal on Control System and Instrumentation, vol. 03, no. 01, 2012.
16. Bateman, S., Gutwin, C., Osgood, N. y McCalla, G., Interactive Usability Instrumentation. Symposium on Engineering Interactive Computing Systems, USA, 2009.
17. Tarby, J., Ezzedine, H., Rouillard, J., Tran, C., Laporte P. y Kolski, C., Traces Using Aspect Oriented Programming and Interactive Agent-Based Architecture for Early Usability Evaluation: Basic Principles and Comparison. HCI (1), Vol. 4550, pp. 632-641, doi:10.1007/978-3-540-73105-4_70, 2007.
18. Coutaz, J., Evaluation techniques: Exploring the intersection of HCI and software engineering. In R.N. Taylor and J. Coutaz, Eds., Software Engineering and Human-Computer Interaction, LNCS, pp. 35–48. Heidelberg, Germany: Springer-Verlag, 1995.
19. Whitefield A., Wilson F. y Dowell, J. A framework for human factors evaluation. Behaviour and Information Technology vol. 10, no. 1, pp. 65–7, 1991.
20. Balbo, S., Automatic evaluation of user interface usability: Dream or reality. Proceedings of the Queensland Computer-Human Interaction Symposium (Queensland, Australia, August). Bond University. 1995.

Modelo de Aplicaciones Sensibles al Contexto (MASCO), caso de estudio: Robot móvil recolector de objetos

Evelina C. Velázquez[1], María del Pilar Gálvez D.[1], Ariel N. Guzman Palomino[1]

[1] Facultad de Ingeniería, Universidad Nacional de Jujuy, Provincia de Jujuy, Argentina.
caro_eve@hotmail.com, mdpgalvezdiaz@fi.unju.com.ar, nel_gp@hotmail.com.

Resumen. El presente trabajo tiene como finalidad la adaptación del Modelo de Aplicaciones Sensibles al Contexto (MASCO) del Grupo de Investigación GRISECO de la Facultad de Ingeniería de la Universidad Nacional de Jujuy, al caso de estudio: Robot móvil recolector de objetos. Se aprovecharon las características del modelo para desarrollar un prototipo robótico cuya función es el proceso de recolección de un objeto que es detectado en la trayectoria de movimiento de un robot dentro de una habitación.

Keywords: Contexto, MASCO, Robótica, Robot.

1 Introducción

La Robótica ha sufrido un gran progreso desde sus inicios en muy pocas décadas, esto es debido a la capacidad de los robots de cumplir tareas críticas y también tediosas para el hombre, en una cada vez más extensa variedad de áreas. Esto es posible a través de sensores y actuadores que conectan a un robot con su entorno. El presente trabajo consiste en la adaptación del Modelo MASCO al desarrollo de un robot de movimiento que posee un brazo robótico con el cual realiza el agarre de objetos, cuya ubicación es determinada por un sensor sonar incluido en el robot, los objetos son colocados en un compartimento en el mismo robot, para que éste cumpla su función de recolección.

El apartado 1 corresponde a la introducción, el 2 consiste en una introducción a los conceptos básicos de Robótica, el apartado 3 define las capas del Modelo MASCO, el cual es adaptado en el apartado 4 al caso de estudio, donde además se indican consideraciones generales de los movimientos que realiza el robot y el cálculo de los ángulos que posiciona al brazo para realizar el agarre del objeto. El apartado 5 contiene las conclusiones y el 6 las referencias bibliográficas.

2 Robótica

La Robótica ha cobrado gran importancia en la actualidad, debido a sus múltiples usos en diferentes áreas, desde usos espaciales, medicina, industria y también usos sociales, entre otros. Esta se define "como el diseño y construcción de máquinas capaces de desempeñar tareas realizadas por el ser humano o que requieren del uso de inteligencia" [1]

En el ámbito social, hay innovaciones con respecto a las aplicaciones de la Robótica, tanto de asistencia a personas con capacidades disminuidas y también en tareas diarias.

Todas las innovaciones se lograron a partir de la percepción de variables a través de dispositivos como sensores, y de las funciones que pueden llevar a cabo los robots. Para clarificar, se define robot como: *"*Manipulador multifuncional y reprogramable, diseñado para mover materiales, piezas, herramientas o dispositivos especiales, mediante movimientos programados y variables que permiten llevar a cabo diversas tareas" según la definición de Robot del Instituto Norteamericano de Robótica. Dado que esta definición se aplica sobre todo a robots industriales, sería más apropiado definirlo como: "un dispositivo electromecánico multipropósito capaz de reaccionar a estímulos captados por sensores, los que son interpretados y que componen el entorno donde se encuentra, mediante el uso de actuadores" [3].

Utilizando sensores y actuadores, los robots pueden ejecutar funciones con información del ambiente que los rodea, tomando variables de interés del contexto, estas son procesadas y tras una decisión, se ejecutan funciones a través de actuadores que permiten al robot interactuar con un ambiente. Se definen los sensores como dispositivos capaces de captar y traducir información en valores que son compresibles u operables, y los actuadores dispositivos de salida que permiten al robot realizar acciones. Así los sensores otorgan al robot la capacidad de tomar valores de interés del ambiente que lo rodea, y los actuadores le permiten realizar tareas en consecuencia para cumplir un objetivo.

3 Modelo de Aplicaciones Sensibles al Contexto (MASCO)

El Modelo para aplicaciones Sensibles al Contexto (MASCO) es una solución para desarrollar sistemas sensibles al contexto (Fig. 1), realizado por el grupo de investigación GRISECO (Grupo de Ingeniería de Aplicaciones Sensibles al Contexto) de la Facultad de Ingeniería de la Universidad Nacional de Jujuy. [4]

MASCO consta de 5 niveles:

- Aplication Layer: consta de los objetos del dominio de la aplicación.
- Context Layer: en esta capa se encuentran los objetos que procesan la información de las otras capas.
- Service Layer: los objetos de esta capa brindan servicios.
- Sensing Concerns Layer: posee objetos que se encargan de la interpretación los datos de la capa inferior adyacente.
- Hardware Abstractions Layer: contiene a los objetos que representan los sensores y actuadores.

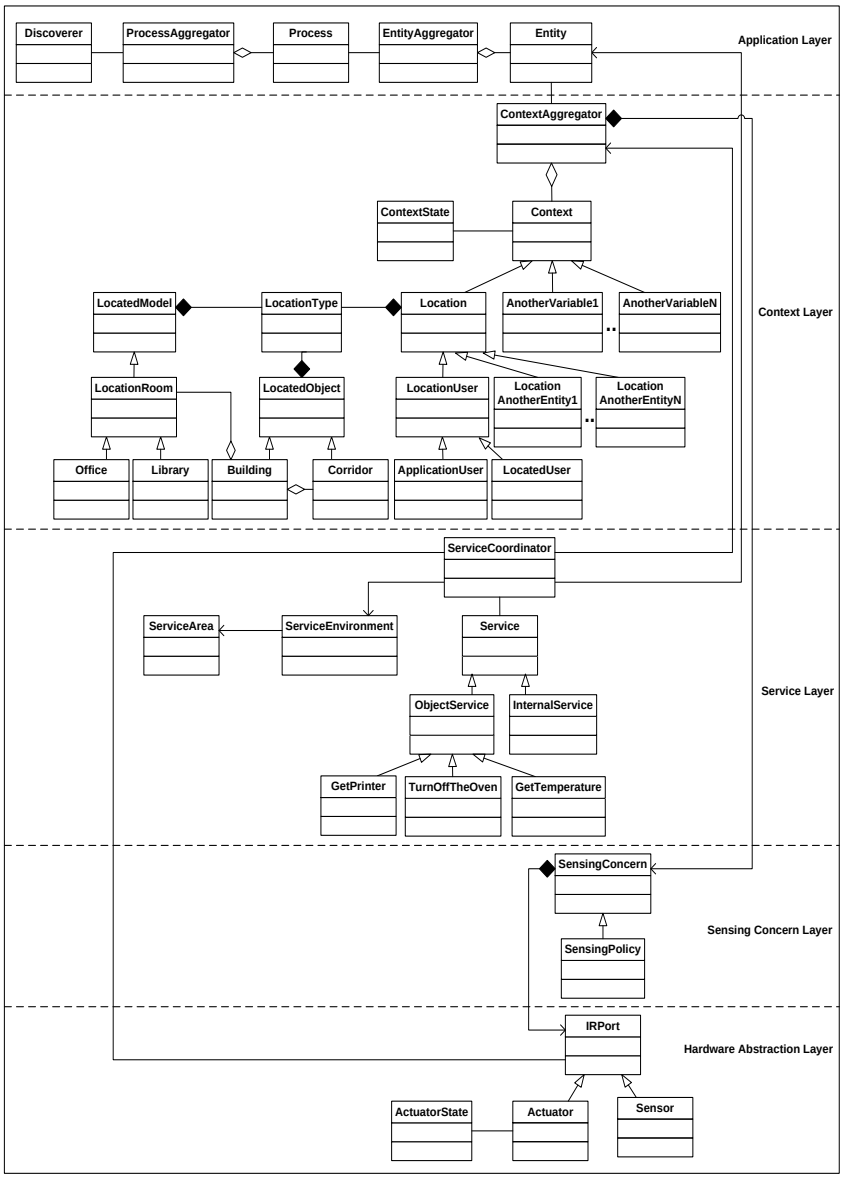


Fig. 1. Modelo MASCO

El Modelo procesa un cambio de valor de una o más variables de contexto, esto se refleja en la Clase IRPort, que se implementa como un patrón Observer. Un Objeto SensignConcern es notificado en la siguiente capa, para analizar y transformar las variables de contexto a valores comprensibles por la aplicación, esta clase se implementa como un patrón Strategy [5]. Posteriormente se notifica a la Clase ContextAggregator de la Context Layer, que se implementa como un patrón Observer

y Mediator [5], si la variable de contexto corresponde a la "ubicación" se la incorpora al objeto correspondiente en Context Layer, que es enviada a un objeto Location que representa la posición actual de la entidad, indicando un cambio de posición de las entidades sensadas.

La capa Service se encarga de la solicitud de servicios, donde ServiceCoordinator, constituye una interface común a todos los servicios, la Clase Service posee dos subclases: ObjectService e InternalService, la primera representa los servicios que son provistos a los objetos en general y la segunda es un servicio que es solicitado por otros servicios, como por ejemplo un cálculo intermedio. Para cada posible servicio hay una subclase concreta de ObjectService. Los servicios relacionados con ubicación, ServiceArea y ServiceEnvironment colaboran con ServiceCoordinator. ServiceArea tiene el conocimiento de la relación del objeto Location con los servicios que son provistos en el área, mientras que ServiceCoordinator conoce la relación con el área en la cual es localizado, por esto ServiceEnvironment actúa como un Patrón Mediador entre ServiceCoordinator y ServiceArea.

Los servicios solicitados por ServiceCoordinator se pueden brindar a entidades, manipulando los actuadores representados en Hardware Abstraction Layer por la clase Actuator, y su estado mediante ActuatorState, para esto se la implementa como un patrón State [5]. En Application Layer la clase Discoverer es responsable de mantiener la información sobre los procesos, entidades y variables de importancia en una aplicación. La clase EntityAggregator coordina la interacción entre objetos Entity, por esto, se implementa como un patrón Mediator [5], y se relaciona con un objeto Process, en caso de haber varios procesos. Cada uno de estos se corresponde a un objeto Discoverer (responsable del mantenimiento de la información sobre los objetos Entity y sus variables de contexto que trabajan en el proceso) y pueden estar relacionados con uno o más objetos Entity, además a cada uno de éstos se los puede relacionar con un objeto ContextAggregator. Cuando una variable de contexto cambia su valor EntityAggregator determina de qué manera deben relacionarse las variables restantes. Una vez definido el comportamiento, solicita a ServiceCoordinator la ejecución de un determinado servicio. Este servicio requiere la solicitud de un actuador, por lo que la subclase Actuator de IRPort contiene operaciones que hacen referencia a los servicios que brinda el objeto real.

4 Caso de Estudio

El caso de estudio corresponde a un robot de movimiento de avance, que recorre una habitación y detecta objetos a través del uso de un sensor sonar para determinar su ubicación. Ésta, está dada por los grados y la distancia a los que se encuentra el objeto en referencia a la ubicación actual del robot.

El robot realiza movimientos de avance, de giros: derecho e izquierdo, retroceso y se detiene. Además realiza la detección de un objeto luego, el robot se ubica de manera que el objeto es asido y depositado en un compartimento del mismo, cumpliendo su objetivo: ser un robot recolector.

Para determinar la ubicación de un objeto el robot posee un sensor sonar que a través de ondas sonoras permite determinar la distancia al objeto, además el sensor

recorre un espectro de 180°, los que se dividen en 4 secciones de 45° para determinar los grados a los que se encuentra (Fig.2).

Al determinar la presencia de un objeto, se procesa la información de su ubicación y se decide los servicios a ser ejecutados para realizar el agarre del mismo mediante un brazo robótico incluido como parte del robot (Fig. 3), formado por 4 secciones: hombro, brazo posterior, antebrazo y mano.

Los movimientos que realiza el robot son los siguientes (Fig. 4): Avanzar, Girar a la Izquierda, Girar a la Derecha, Detener, Retroceder.

Los movimientos que realiza el brazo robótico son los siguientes: a) Mover Hombro: Movimiento de rotación de hasta 180°; b) Mover Brazo Posterior: Movimiento de traslación de hasta 180°; c) Mover Antebrazo: Movimiento de traslación de hasta 160°; d) Movimiento de mano: Incluye movimiento de rotación de hasta 180° y de agarre.

Los movimientos del brazo robótico consisten en posicionar al mismo para que las pinzas (extremidades de la mano) puedan sujetar un objeto con características conocidas pero ubicación desconocida. [3]

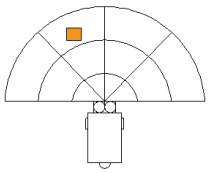


Fig. 2. Detección de Objetos mediante Sonar. [6]

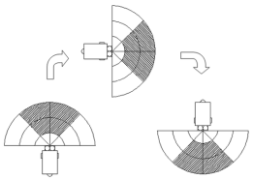


Fig. 4. Movimientos del robot. [6]

El brazo posterior, antebrazo y la distancia entre ambos forma un triángulo a partir del cual se puede definir la posición de ambos con respecto a los ángulos de traslación, para que la mano, que tiene una posición horizontal con respecto al eje x, pueda asir el objeto. (Fig. 5)

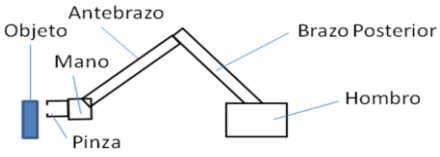


Fig. 3. Brazo Robótico.

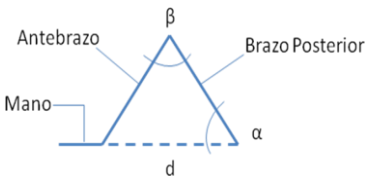


Fig. 5. Posición de agarre de un objeto.

La distancia desde el robot al objeto, definida como "D", se determina a través del sensado que realiza el sensor y procesamiento para convertir tiempo de retardo en esa distancia y los grados a los que se encuentra, sin embargo, se puede definir esta distancia como la suma entre el largo de la sección mano (que incluye a las pinzas de agarre) y una distancia d a determinar, como se indica en (1):

$$D = a + d \quad \textbf{(1)}$$

Siendo:

D: Distancia entre el robot y el objeto detectado, valor sensado.
a: longitud de la sección mano, valor conocido.

d: distancia entre el brazo y el objeto para realizar el agarre de un objeto.

Para calcular d, se obtiene (2) al despejar "d" de (1):

$$d = D - a \quad \textbf{(2)}$$

Dada las longitudes conocidas del antebrazo y brazo posterior y la distancia "d" calculada en (2), es posible determinar los grados α y β, que corresponden a los grados de giro de las dos secciones mencionadas, utilizando (3)

$$\text{Tan } \gamma = Co/Ca \quad \textbf{(3)}$$

Siendo:

γ: Ángulo a determinar.
Co: Cateto opuesto, valor conocido.
Ca: Cateto adyacente, valor conocido.

Así para calcular el ángulo de movimiento de traslación del brazo posterior se utiliza (4):

$$\alpha = \text{Arcotan } (l_a/d) \quad \textbf{(4)}$$

Siendo:

α: Ángulo de giro para realizar el movimiento de traslación del brazo posterior.
l_a: longitud de la sección antebrazo, valor conocido.
d: distancia entre el brazo y el objeto para realizar el agarre de un objeto. Calculada en (2)

Para el cálculo del ángulo de movimiento de traslación del antebrazo se utiliza (5):

$$\beta = \text{Arcotan } (d/l_b) \quad \textbf{(5)}$$

Siendo:

β:Ángulo de giro para realizar el movimiento de traslación del antebrazo.
l_b: longitud de la sección brazo posterior, valor conocido.
d: distancia entre el brazo y el objeto para realizar el agarre de un objeto. Calculada en (2)

4.1 Hardware Abstraction Layer

La Hardware Abstraction Layer (Fig. 6) consta de la Clase IPPort, que se implementa como un sensor o un actuador con un patrón Observer [5]. La Clase Actuator posee un estado, esto es, para que al momento de definir un servicio, éste ejecute una acción permitida de acuerdo al estado actual del actuador, mediante un patrón State [5]. Se definen las clases Port para representar la conexión con los servomotores físicos de un robot, donde cada servomotor se implementa como un objeto Servo que realizan movimientos de avance o de agarre. La Clase Sonar representa el sensor que detecta un objeto a través de la emisión de ondas sonoras. [6]

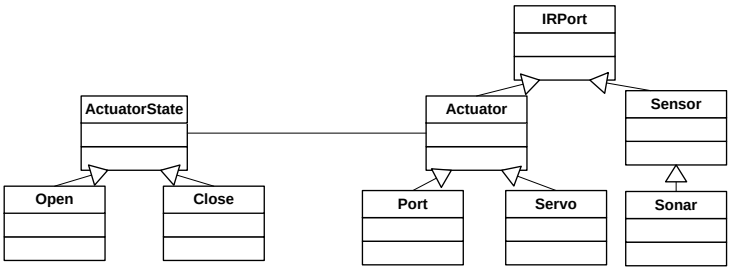


Fig. 6. Hardware Abstraction Layer adaptada al caso de estudio.

4.2 Sensing Concern Layer

La Sensing Concern Layer (Fig. 7) corresponde a la capa donde se realiza el procesamiento de las variables sensadas para producir un valor que pueda ser comprensible por las demás capas de la aplicación. La Clase SensingConcern se implementa como un patrón Strategy [5]. Se establece la clase LocationObjectPolicy que transforma el valor sensado en la distancia entre el objeto y el robot, como se indica:

$$D = V_s * T/2 \quad \textbf{(6)}$$

Siendo:

D: Distancia entre el robot y el objeto.
V_s: Velocidad del Sonido, conocida.
T: Tiempo de retardo. Valor sensado.

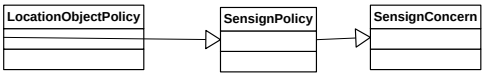


Fig. 7. Sensing Concern Layer adaptada al caso de estudio.

4.3 Context Layer

En la Context Layer (Fig. 8) se toma la decisión de qué servicios deben ser solicitados en la siguiente capa, Service, para esto la Clase ContextAggregator se implementa como un patrón Mediator [5]. Las clases Distance y Degree representan las dos variables de contexto del caso de estudio, ambas determinan la ubicación del objeto a ser asido, que se registra en el objeto InitialLocationObject. También se define la Clase RobotLocation [6] para definir la ubicación del robot dentro de la habitación. La Clase FinalLocationObject representa la ubicación final del objeto, en el compartimento del robot.

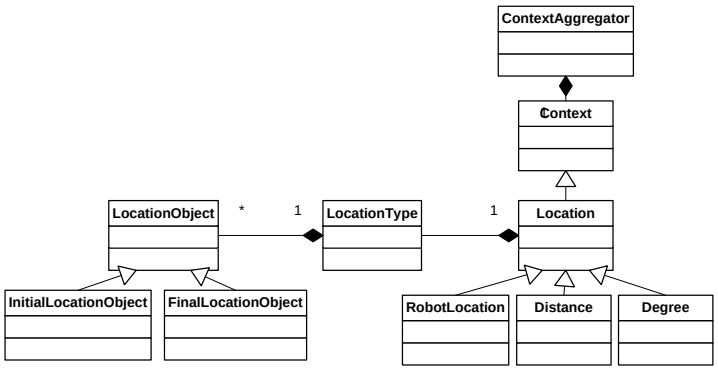


Fig. 8. Context Layer adaptada al caso de estudio.

4.4 Service Layer

La capa Service Layer (Fig. 9) posee la clase ServiceCoordinator que se implementa como un patrón Mediator [5], y coordina los servicios para realizar un movimiento del robot o el agarre de un objeto. La clase ObjectService establece los servicios de movimiento del robot: de giros hacia la izquierda y derecha, de movimientos adelante, atrás y de parada que consume el servicio interno de la Clase Motion, y de agarre que corresponde a los servicios para mover el brazo: MoveSoulder, MoveRearArm, MoveForeArm, MoreHand, HoldObject, DropObject, los que a su vez utilizan los servicios internos, que corresponden a las clases Traslation y Rotation.

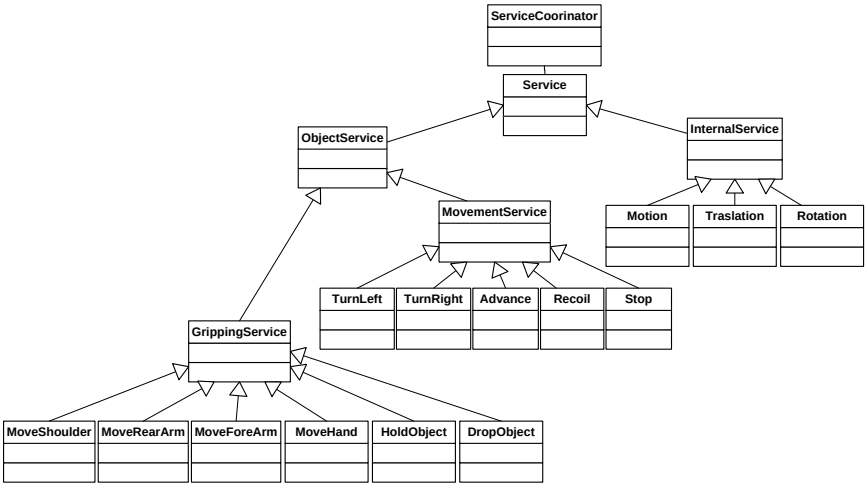


Fig. 9. Service Layer adaptada al caso de estudio.

4.5 Application Layer

La capa Application Layer (Fig. 10) gestiona la información de las entidades más importantes. Para el caso de estudio se representa el brazo robótico como una entidad, ya que posee secciones que deben coordinarse para realizar el agarre de un objeto. Se define una Clase por cada sección para su coordinación.

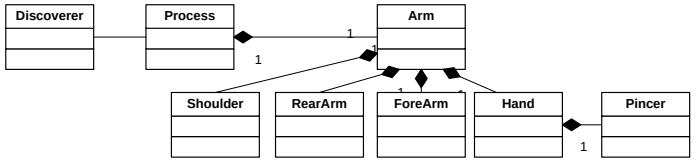


Fig. 10. Application Layer adaptada al caso de estudio.

Finalmente, el Modelo adaptado resultante (Fig. 11) contempla el movimiento de un robot, la detección y agarre de objetos.

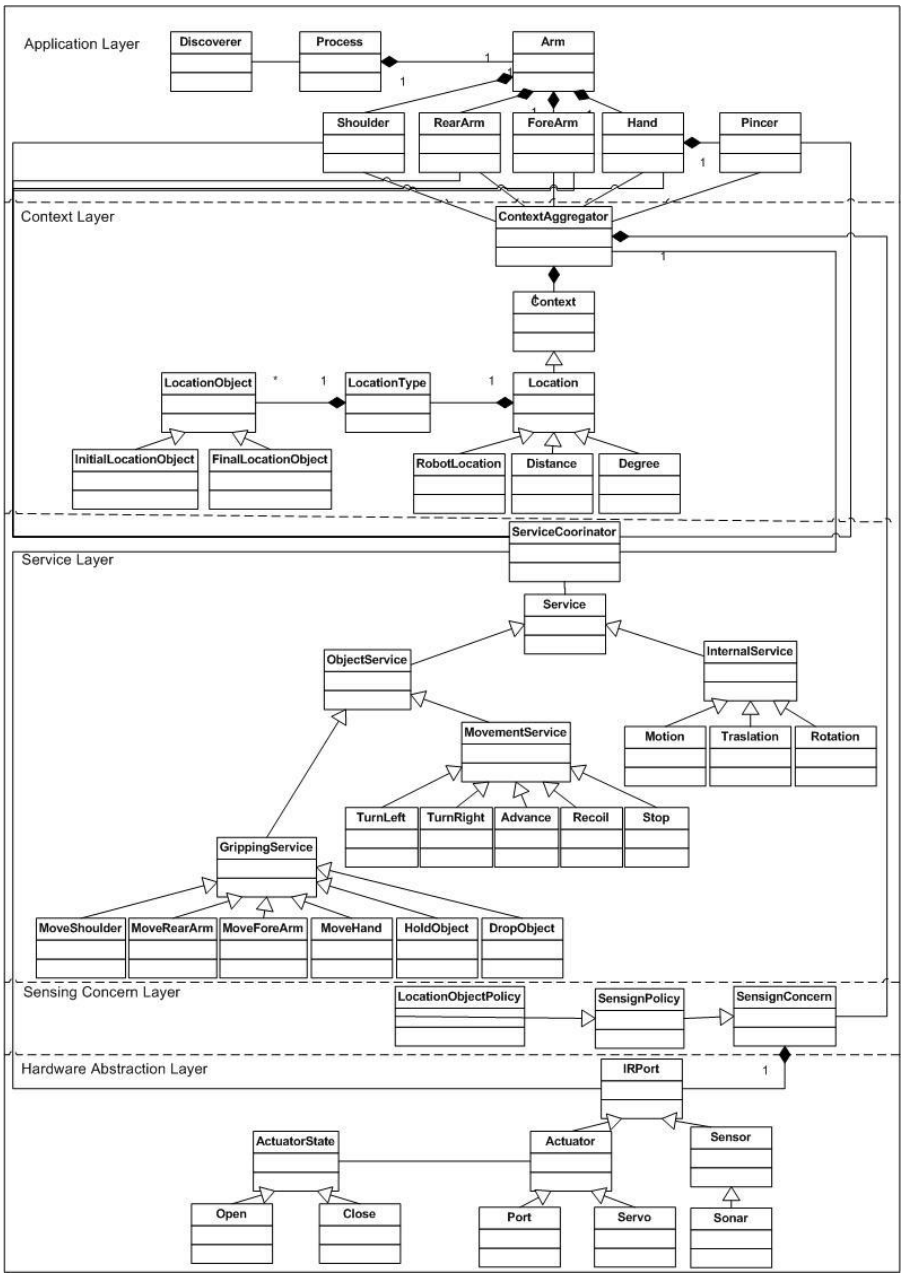


Fig. 21. Modelo MASCO adaptado al caso de estudio

5 Conclusiónes

El presente trabajo concluye con la adaptación del Modelo MASCO al caso de estudio: Robot de avance recolector de objetos, que se implementó utilizando un robot a escala con una placa Arduino Uno, manejado por un sistema sofware desarrollado en Visual Studio 2013 Community de licencia gratuita.

Para esto se adaptaron las capas de MASCO con Clases que representan sensores y actuadores necesarios para su funcionamiento, y también servicios para que el robot cumpla con su objetivo: asir objetos con una posición indefinida inicialmente, pero con un peso aceptable para posteriormente depositarlos en un compartimento propio. También permite el movimiento del robot en una habitación para encontrar estos objetos. Esto fue posible debido a que MASCO es un Modelo en capas, que posee las conexiones necesarias para que cada capa se conecte con aquellas que hacen un procesamiento necesario. Estas capas en general no son adyacentes para el proceso de decisión y ejecución de servicios, esto permite mayor rapidez en el funcionamiento del robot, esto es debido al uso de patrones en el diseño de MASCO.

Además al ser Orientado a Objetos es flexible para representar entidades como las secciones del brazo que forman parte del robot y las variables de contexto que determinan la posición de un objeto.

Se concluye que MASCO no sólo es aplicable al dominio de la Robótica como se demostró en [6] sino que además permite el manejo de entidades y varias variables de contexto, como así también servicios muy diferentes: los referidos a movimientos de avance y los de agarre, ampliando su utilización como una solución de Software.

Referencias

1. http://robotica.wordpress.com/about/
2. Zabala, G.: Robótica, Guía Teórica y Práctica. User Power, Argentina (2011)
3. Velázquez, E.: Sistema de Comando de Movimiento de Brazo Robótico, Trabajo Final de Carrera. Facultad de Ingeniería, UNJU, Jujuy (2015)
4. Gálvez, M.P., Cáceres, N.R, Velázquez, C. E., Guzmán, A. N.: Atributos de calidad del modelo para aplicaciones sensibles al contexto MASCO. En: 5to Simposio Internacional de Investigación: "Interdisciplinariedad, Multidisciplinariedad y/o transdisciplinariedad: en la búsqueda de respuestas desde las experiencias de investicación". Universidad Católica de Santiago del Estero, DASS, Jujuy (2013)
5. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Elements of Reusable OO Software. Addison Wesley, USA (1995)
6. Velázquez, E., Guzmán, A., Gálvez, M.P.: Modelo en capas sensibles al contexto aplicada a la movilidad de un robot. En: Cuarto Simposio Internacional de Investigación. Libro de Resúmenes. Pág. 117. Universidad Católica de Santiago del Estero, DASS, Jujuy (2011)

Datasheet Modeler: Una herramienta de soporte para el desarrollo de funcionalidades en LPS

Mailen Mancuso[1] *, Agustina Buccella[12], Alejandra Cechich[1], Maximiliano Arias[12], and Matias Pol'la[12]

[1] GIISCO Research Group
Departamento de Ingeniería de Sistemas - Facultad de Informática
Universidad Nacional del Comahue
Neuquén, Argentina
{mailen.mancuso,agustina.buccella,maximiliano.arias,matias.polla,
alejandra.cechich}@fi.uncoma.edu.ar
[2] Consejo Nacional de Investigaciones Científicas y Técnicas - CONICET

Abstract. Las líneas de productos de software (LPS) son un paradigma basado en el reuso definido mediante la especificación de aspectos comunes y variables para la creación flexible de productos dentro de un dominio en particular. En este trabajo, y dentro de esta misma línea, hemos definido un marco de desarrollo para dar soporte al diseño de funcionalidades, adaptadas a nuestra metodología LPS propuesta en trabajos previos. A su vez, hemos implementado una herramienta de soporte llamada *Datasheet Modeler*, la cual permite a los ingenieros de software diseñar funcionalidades dentro del marco definido. Por último, presentamos el diseño de una nueva funcionalidad en un contexto real.

Keywords: Líneas de Producto de Software, Gestión de Variabilidad, Herramientas de Software

1 Introducción

Las líneas de productos de software (LPS) [2, 8] han surgido como un paradigma de reuso muy útil cuando estamos analizando dominios particulares. El paradigma se sustenta en base a dos ingenierías: la de *dominio* y la de *aplicación*. La primera se basa en la definición de servicios o características comunes y variables que conformarán parte de una plataforma LPS. La segunda, en base a esta plataforma, permite personalizar los servicios variables y a su vez crear nuevos servicios, de manera de cumplir con las particularidades de cada organización. Dentro de ambas ingenierías, un aspecto importante es el denominado *gestión de variabilidad*, ya que engloba todas actividades necesarias para identificar, analizar, seleccionar, modelar, implementar e instanciar la misma dentro de la plataforma LPS.

* Este trabajo está parcialmente soportado por el proyecto UNCOMA F001 "Reuso Orientado a Dominios" como parte del programa "Desarrollo de Software Basado en Reuso".

En la literatura existen muchos trabajos relacionados con la variabilidad, los cuales proponen diferentes enfoques para dar soporte a las actividades involucradas en la misma. Por ejemplo, los trabajos presentados en [6, 7] analizan diferentes aspectos como el modelo utilizado, tipos, actividades del desarrollo LPS soportadas, etc. Dentro del modelado de la variabilidad, el cual es un aspecto en el que estamos interesados, las propuestas se enmarcan en el modelo de variabilidad soportado. Entre los más comunes, están: el modelo de características, el modelo de decisión, extensiones UML y el modelo ortogonal OVM (Orthogonal Variability Model) [7]. A su vez, varias de estas propuestas proveen herramientas de soporte para asistir al modelo y a las diferentes particularidades que se desprenden de los mismos. Por ejemplo, herramientas como XFeature[3] y fmp (Feature Modeling Plug-in)[4], proveen soporte para el diseño de variabilidades dentro del modelo de caractarísticas, y VarMod[5] y PLUM[6] implementan el modelo OVM.

En este trabajo estamos interesados en las tareas de modelado y diseño de las LPSs como parte de la ingeniería de dominio. Nuestra propuesta surge como una continuación de trabajos realizados previamente [3–5] para la definición de un proceso de desarrollo de LPSs basado en recursos semánticos, definidos como artefactos de software. En este trabajo presentamos el marco de desarrollo necesario para llevar a cabo el diseño de funcionalidades, junto con su herramienta de soporte que implementa el mismo. De esta manera, el diseño de funcionalidades se efectúa en una forma mas rápida y ágil, beneficiando así, al proceso de desarrollo en general.

El artículo se estructura de la siguiente manera. En la sección siguiente se describen los antecedentes de nuestra propuesta, enfocándonos en las bases de nuestra metodología y en los artefactos de software creados. Luego, en la Sección 3, se describe el marco de desarrollo en el cual se basa el diseño de funcionalidades en la plataforma LPS junto con la herramienta de soporte desarrollada. En la Sección 4 se ilustra dicho proceso mediante el desarrollo de una nueva funcionalidad dentro de una LPS. Por último se describen las conclusiones y trabajos futuros.

2 Antecendentes

En trabajos previos [3–5] hemos presentado una metodología para el desarrollo de LPS basada en niveles de dominios. En particular, la metodología define recursos semánticos mediante representaciones específicas enmarcadas dentro del enfoque de modelado OVM y en una implementación anotativa y composicional. Dichos recursos asisten a la comunicación entre los participantes y definen reglas para la construcción de artefactos de software en cada una de las etapas del desarrollo. En la Figura 1 podemos observar algunas de las actividades de la

[3] http://www.pnp-software.com/XFeature/
[4] http://gsd.uwaterloo.ca/fmp
[5] http://www.swpl.de/SEGOS-VM-Tool/index.html
[6] http://www.esi.es/plum/index.php

fase de ingeniería de dominio de dicha metodología, junto con los artefactos de software creados. La metodología posee más actividades que construyen otros artefactos de software necesarios para dar soporte a la ingeniería de dominio, los cuales pueden ser vistos en [5].

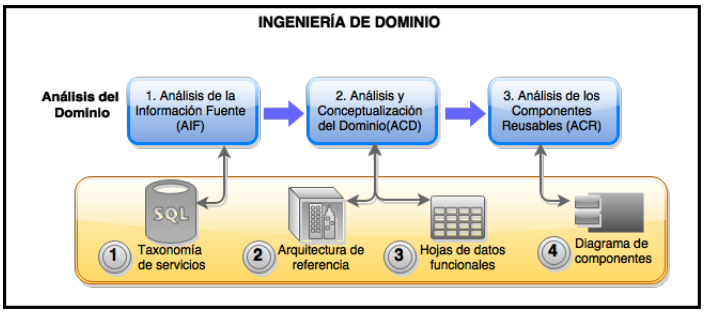


Fig. 1. Parte de las actividades involucradas en la ingeniería de dominio de las LPSs

En la parte inferior de la Figura 1 podemos observar los cuatro artefactos de software (numerados del 1 al 4) que deben ser desarrollados dentro de las tres actividades involucradas en el análisis del dominio. El primero de ellos es la taxonomía de servicios [4] creada en la actividad de análisis de la información fuente (AIF). La misma es una estructura jerárquica, la cual posee en los nodos superiores las categorías por las cuales clasificar los servicios. El segundo artefacto es la definición de una arquitectura de referencia como parte de la actividad de análisis y conceptualización del domino (ACD). Dicha arquitectura debe especificar una estructura preliminar para la interacción de los servicios definidos en la taxonomía. A su vez, en esta actividad se debe crear el tercer artefacto, las hojas de datos funcionales, las cuales poseen la especificación de cada funcionalidad. Por último, el cuarto artefacto es una estructura preliminar de componentes de software basados en la especificación previa de las hojas de datos funcionales. Dicha estructura debe crearse considerando la clase de servicios definidos (comunes, variantes y puntos de variación) y sus interacciones. La misma se realiza en la actividad de análisis de los componentes reusables (ACR).

Para dar soporte a la construcción de los artefactos mostrados en la Figura 1, hemos definido las pautas fundamentales para dar la construcción de los artefactos de software involucrados. Así, en [1] hemos desarrollado la herramienta llamada *Service Mapper*, la cual asiste a los ingenieros de software en la búsqueda de los servicios de la taxonomía más adecuados para cumplir con los requerimientos del domino (formulados por los usuarios expertos). Luego, en este trabajo describimos el marco de trabajo y los recursos definidos para la creación de las hojas de datos funcionales por cada funcionalidad de la LPS. A su vez, aquí también se crean los archivos XML que sirven para describir, en un formato

legible por una computadora, las relaciones y servicios definidos en cada una de las funcionalidades. Por último, estos archivos son leídos y analizados para crear la estructura de componentes preliminar que deberá ser analizada por los ingenieros de software y desarrolladores, para luego conformar la arquitectura de la plataforma.

3 Diseño de Funcionalidades en LPSs

Como hemos descripto en la sección anterior, el diseño de funcionalidades de la plataforma LPS se basa en la creación de las hojas de datos funcionales, las cuales poseen una serie de ítems para la especificación de las dependencias y variabilidades de los servicios de la taxonomía. Los ítems a completar dentro de cada hoja de datos son: la identificación o código de la funcionalidad con su descripción textual, el dominio donde la funcionalidad está incluida, la lista de servicios utilizados, un modelo gráfico que muestre la interacción de los servicios para lograr dicha funcionalidad y, finalmente, un conjunto de archivos XML que describen dicha interacción [5].

Para el modelo gráfico dentro de la hoja de datos definimos, a su vez, una serie de dependencias que nos permiten representar las interacciones posibles entre los servicios, junto con su variabilidad, las cuales se basan en una extensión del enfoque OVM. Las mismas son:

- *Uso* (<*MandatoryVP*>): especifica una dependencia entre servicios comunes que no necesariamente están asociados a puntos de variación.
- *Punto de Variación Obligatorio* (<*MandatoryVP*>): indica la obligatoriedad de seleccionar los servicios variantes.
- *Punto de Variación Opcional* (<*OptionalVP*>): indica que pueden seleccionarse cero o más servicios variantes.
- *Punto de Variación Alternativo* (<*AlternativeVP*>): indica que debe seleccionarse sólo un servicio variante (relación XOR).
- *Punto de Variación Variante* (<*VariantVP*>): indica que debe seleccionarse al menos un servicio variante (relación OR).
- *Requiere* (*dependency:Requires = "serviceName"*): especifica que si un servicio variante es seleccionado, requiere que otro servicio también lo sea, independientemente del punto de variación en donde estén asociados.
- *Excluye* (*dependency:Excludes = "serviceName"*): es el opuesto al anterior, en donde se excluye a un servicio variante si otro es elegido.
- *Punto de Variación Global* (<*GV*>): especifica que si se instancia el punto de variación de una manera específica, la misma será aplicada a todas las funcionalidades que contengan dicho punto.
- *Punto de Variación Específico* (<*SV*>): especifica que la instanciación de un punto de variación es particular a cada funcionalidad.

En la descripción previa, podemos observar la etiqueta XML utilizada para la creación de los archivos XML que describen la interacción de los servicios en el modelo gráfico. Estos archivos, que también son parte de la hoja de datos, son creados para garantizar un formato legible por la computadora y así permitir el

uso de herramientas de software en etapas posteriores. Los tres tipos de archivos XML creados a partir de cada una de las hojas de datos funcionales son: *información del servicio*, el cual especifica información por cada servicio involucrado como su identificación y nombre, entre otros; *interacción de servicios*, especificando todas las dependencias entre servicios del diseño en la notación gráfica de la hoja de datos; y *restricción de servicios* que posee las restricciones de *excluye* y *requiere*, cuando sea necesario.

3.1 Herramienta de soporte para el Diseño de Funcionalidades

Para diseñar funcionalidades de la plataforma desarrollamos una herramienta de soporte, llamada *Datasheet Modeler*, que permite a los ingenieros de software o diseñadores interactuar con los servicios de la taxonomía y con la arquitectura de referencia (artefactos 1 y 2 de la Figura 1) para definir las hojas de datos funcionales con todos sus ítems. La herramienta permite entonces, crear cada funcionalidad y a su vez generar los archivos XML, según las etiquetas definidas para las dependencias. En la Figura 2 podemos observar los componentes que conforman la herramienta.

El componente de *Definición de Hojas de Datos* es el responsable de solicitar al usuario los ítems generales por lo cuales está compuesta cada hoja de datos, siendo ellos: la identifición de la funcionalidad, el nombre y el dominio en donde estará incluida la funcionalidad. Como podemos observar en la Figura 3, la interfaz gráfica le permite al diseñador completar esos ítems mediante un formulario y además, la interfaz posee dos botones que le permiten al usuario realizar el diseño gráfico de la funcionalidad y luego, realizar la transformación XML para generar los archivos correspondientes al diseño.

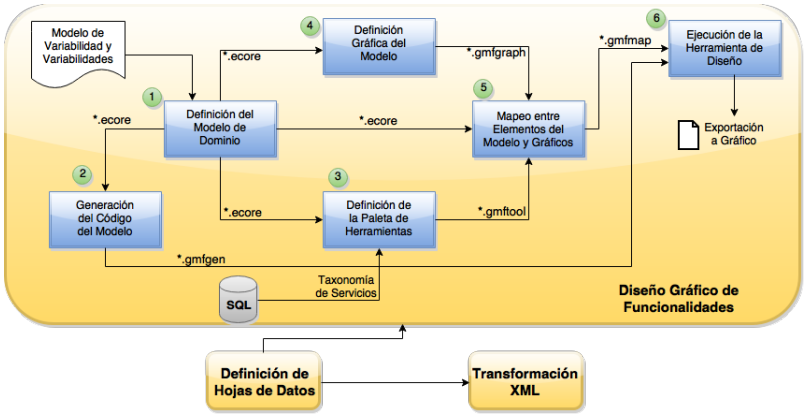


Fig. 2. Proceso para la creación de la herramienta de soporte *Datasheet Modeler*

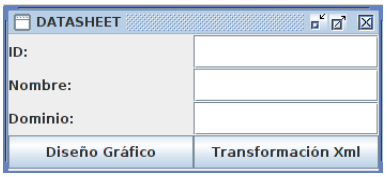
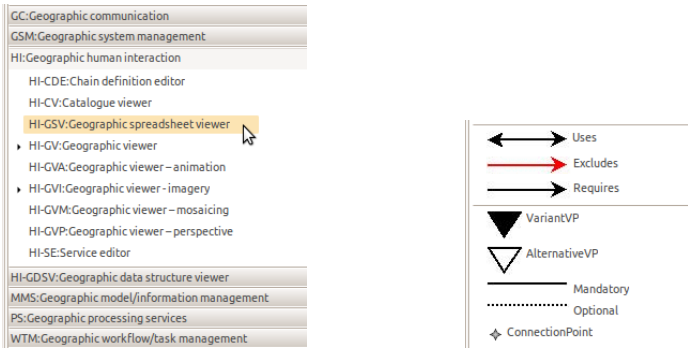


Fig. 3. Interfaz gráfica para la definición de hojas de datos

Al presionar el botón *Diseño Gráfico* se invoca a los componentes involucrados en la herramienta de diseño gráfico en sí misma, la cual fue desarrollada utilizando el framework *GMF (Graphical Modeling Framework)*[7] provisto por la plataforma de desarrollo Eclipse[8], implementado en JAVA. En la parte superior de la Figura 2, podemos observar un conjunto de seis componentes que representan cada una de las etapas necesarias para cumplir con todos los requerimientos que implica el desarrollo del editor gráfico. Estas etapas son:

1. *Definición del Modelo*: Aquí se genera el *modelo de dominio base* para el editor gráfico. Todo lo que el modelo base permita y restringa, será exactamente lo que el asistente de GMF permita realizar para diseñar las hojas de datos. Así, considerando la naturaleza de los servicios de la taxonomía, las dependencias y variabilidades permitidas, definimos un modelo conceptual que incluye las abstracciones necesarias para interactuar con todos ellos. Como podemos observar en la Figura 3, la salida del componente es un archivo **.ecore* del cual se derivarán el resto de los subprocesos, generando así una serie de transformaciones de modelos para la implementación final de la herramienta.
2. *Generación del Código del Modelo*: Este componente genera un archivo con extensión **.gmfgen*, a partir del archivo **.ecore* anterior. Permite transformar automáticamente el modelo de dominio base, a su correspondiente código fuente. Este proceso se lleva a cabo mediante la aplicación de patrones de transformación, y el resultado es un conjunto de clases Java.
3. *Definición de la Paleta de Herramientas*: El tercer componente, que utiliza el archivo **.ecore*, se encarga de especificar los elementos que componen la paleta gráfica para diseñar las hojas de datos. En nuestro caso, definimos una serie de nodos para especificar los servicios y enlaces para sus dependencias, ya que todos ellos representan el comportamiento de las variabilidades entre los diferentes servicios. A su vez, aquí creamos las imágenes correspondientes a los elementos (íconos) de la paleta. De esta manera, la paleta contiene todos los recursos que se pueden utilizar para crear el modelo gráfico de las hojas de datos. La parte (b) de la Figura 4 muestra dicha paleta, según está implementada en la herramienta, con sus respectivas imágenes y descripciones. Este componente genera como resultado un archivo **.gmftool*.

[7] https://www.eclipse.org/gmf-tooling/
[8] https://www.eclipse.org



(a) Servicios de la Taxonomía (b) Variabilidades

Fig. 4. Paleta

A su vez, es importante resaltar que durante la ejecución de la herramienta (paso posterior) se podrá contemplar en la paleta, la taxonomía de servicios proveniente de una base de datos (Parte (a) de la Figura 4). Así, la paleta incorpora los servicios reales de forma dinámica, con los cuales el diseñador puede generar nuevas funcionalidades, siendo servicios actualizados y existentes en una base de datos previamente definida.

4. *Definición Gráfica del Modelo*: El cuarto componente permite especificar la definición gráfica del modelo de dominio base, a partir del archivo **.ecore*. Esto significa que, para cada uno de los elementos del dominio, habrá una manera particular de graficarlos. En concreto, aquí diseñamos las diferentes figuras que cada uno de nuestros elementos debe poseer. Así, por cada una de las primitivas definidas, realizamos su correspondiente figura. Este componente genera como resultado un archivo **.gmfgraph*.
5. *Correspondencia entre Elementos del Modelo y Gráficos*: En este componente se realiza el mapeo entre los resultados obtenidos por los componentes 1, 3 y 4. El mismo consiste en establecer, para cada elemento del modelo del dominio base, su respectiva representación gráfica en la paleta (según el archivo **.gmftool*) y su correspondiente definición gráfica (según el archivo **.gmfgraph*). Este componente genera como resultado un archivo **.gmfmap*. A su vez, en esta etapa se establecen las restricciones (OCLs) necesarias, limitadas a la etapa de diseño únicamente. Algunas de estas restricciones fueron agregadas para evitar valores nulos en varios atributos de las clases, especificar obligatoriamente servicios origen y destino para las dependencias, controlar que los puntos de variación estén correctamente definidos y no permitir ciclos en estas relaciones. Todas estas restricciones evitan que la transformación XML posterior produzca errores e inconsistencias en los modelos generados.
6. *Ejecución de la Herramienta de Diseño*: La ejecución del modelo **.gmfgen*, genera una plataforma Eclipse que permite poner en ejecución el editor construido, para realizar los diseños gráficos de las funcionalidades. Siempre y

cuando se respeten las reglas explicitadas en los pasos anteriores, el asistente de GMF se encargará de graficar cada uno de los elementos selectos de la paleta, según se hayan definido. Así, la paleta presentada en la Figura 4 es utilizada por el diseñador para generar las hojas de datos. Aquí, el diseñador podrá ver los servicios de la taxonomía mostrados en un árbol jerárquico y podrá seleccionar los que desee para diseñar sus funcionalidades, respetando las restricciones del modelo. Al finalizar con el diseño de cada funcionalidad, la herramienta permite exportar el modelo gráfico a un formato de imagen (png o gif).

Por último, una vez ingresados los ítems de una hoja de datos funcional y habiendo creado el modelo gráfico de la misma, la herramienta permite generar los archivos XML que describen la funcionalidad. Presionando el botón de *Transformación XML* (de la Figura 3) se invoca al componente del mismo nombre (Figure 2), el cual crea cada uno de los archivos. De esta manera, por cada funcionalidad el componente crea el archivo XML de *interacción de servicios*, el cual se genera a partir de la obtención de los datos relevantes del archivo generado automáticamente por GMF, como los servicios involucrados, sus dependencias, restricciones y variabilidades. También crea un archivo XML de *información del servicio* por cada servicio involucrado en la funcionalidad. Por último, el componente crea un archivo de *restricción de servicios* en caso de haber colocado restricciones de *excluye* o *requiere* entre servicios.

4 Diseñando una Nueva Funcionalidad

La herramienta de soporte para el diseño de funcionalidades fue aplicada para el desarrollo de una nueva funcionalidad, a fin de mostrar su funcionamiento. En particular, describimos el diseño de la funcionalidad *Análisis de Datos Biológicos* incluida dentro del dominio de Ecología Marina. Esta funcionalidad permite a los usuarios del producto final, seleccionar un conjunto de datos almacenados y elegir formas gráficas de visualización de los mismos, como por ejemplo, mediante histogramas, diagramas de tortas, etc.

En la Figura 5 se muestra una parte del modelo gráfico para la funcionalidad analizada. El ingeniero de software puede definir componentes arquitectónicos (en este caso capas de una arquitectura) y arrastrar servicios y sus dependencias desde la paleta hacia el editor de la herramienta. En la figura podemos ver que la paleta contempla los servicios (en la parte superior derecha), estructurados jerárquicamente según la taxonomía, y todas las posibles dependencias, restricciones y relaciones de variabilidad (en la parte inferior derecha) como vimos en la Figura 4. En este caso, el ingeniero ha ido seleccionando y relacionando una serie de servicios comunes y variantes, colocando también componentes arquitectónicos. Se han adicionado dos capas, “Human interaction” y “User Processing”, y agregado el servicio PS-S5.1 (análisis de datos biológicos) que posee una dependencia de uso con el servicio HI-GDV (visor gráfico de datos). Éste último se define a su vez, como punto de variación de dos servicios variantes HI-GSV

(visor geográfico de hoja de cálculo) y HI-GCV (visor geográfico), el cual es también un punto de variación de tres servicios variantes HI-GCV1 (mostrar información como histograma), HI-GCV2 (mostrar información como gráfico de área) y HI-GCV3 (mostrar información como diagrama de torta).

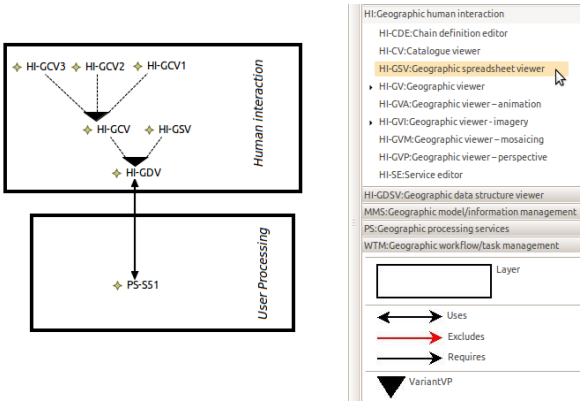


Fig. 5. Parte del diseño de la funcionalidad Análisis de Datos Biológicos utilizando la herramienta

Una vez finalizado el diseño de la funcionalidad y completado el resto de los datos de la hoja de datos funcional, generamos los archivos correspondientes mediante la transformación XML. Como producto de dicha transformación obtenemos al menos siete archivos que corresponden a cada uno de los servicios incluidos en la funcionalidad (Figura 5). A su vez, se genera también el archivo de interacción de servicios de la funcionalidad con todos los servicios que la integran y sus respectivas dependencias, restricciones y variabilidades. Este archivo puede observarse en la Figura 6. En este ejemplo, el proceso de transformación XML no creó el archivo de restricciones, ya que no hay ninguna restricción de requiere y/o excluye en la funcionalidad diseñada.

Como hemos descripto en la Sección 2 y específicamente en la Figura 1, estos archivos XML son los que se utilizan para crear la estructura de componentes preliminar (cuarto artefacto). La misma debe ser luego analizada por los ingenieros de software para conformar la arquitectura de la plataforma.

5 Conclusión y Trabajo Futuro

En este trabajo hemos presentado el marco de desarrollo necesario para el diseño de funcionalidades dentro de una plataforma LPS. El diseño de funcionalidades se basa principalmente en la creación de hojas de datos funcionales, las cuales se construyen en base a otros recursos semánticos ya definidos en trabajos previos de nuestra metodología de desarrollo de LPS [5]. A su vez, para dar soporte

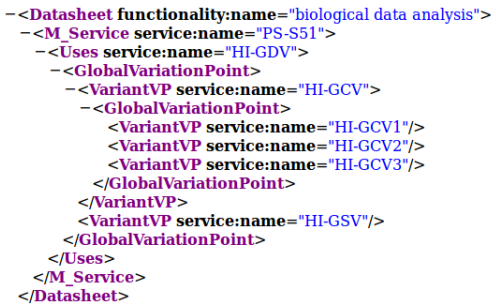


Fig. 6. Archivo XML de interacción de servicios para la funcionalidad análisis de datos biológicos

al diseño de las hojas de datos funcionales, hemos presentado el desarrollo de una herramienta, como un plug-in de Eclipse, llamada *Datasheet Modeler*, la cual provee todos los recursos necesarios (taxonomía de servicios y dependencias de variabilidad) para crear cada una de las funcionalidades incluidas en una plataforma LPS. Como trabajo futuro continuamos extendiendo las reglas y procedimientos necesarios para completar las actividades de la ingeniería de domino definidas en nuestra metodología, para luego extendernos a la fase de ingeniería de la aplicación basada en los artefactos creados.

References

1. Arias, M., Renzis, A.D., Buccella, A., Cechich, A., Flores, A.: Búsqueda de servicios para asistir en el desarrollo de una línea de productos de software. In: Proceedings of the ASSE'15. Rosario, Argentina (2015)
2. Bosch, J.: Design and use of software architectures: adopting and evolving a product-line approach. ACM Press/Addison-Wesley Publishing Co. (2000)
3. Buccella, A., Cechich, A., Arias, M., Pol'la, M., Doldan, S., Morsan, E.: Towards systematic software reuse of gis: Insights from a case study. Computers & Geosciences 54(0), 9 – 20 (2013)
4. Buccella, A., Cechich, A., Pol'la, M., Arias, M., Doldan, S., Morsan, E.: Marine ecology service reuse through taxonomy-oriented SPL development. Computers & Geosciences 73(0), 108 – 121 (2014)
5. Buccella, A., Pol'la, M., Cechich, A., Arias, M.: A variability representation approach based on domain service taxonomies and their dependencies. In: International Conference of the SCCC'14. Chile (2014)
6. Chen, L., Muhammad, B.A., Nour, A.: Variability management in software product lines: A systematic review. In: Proceedings of the 13th International SPL Conference. pp. 81–90. SPLC '09, Pittsburgh, PA, USA (2009)
7. Galster, M., Weyns, D., Tofan, D., Michalik, B., Avgeriou, P.: Variability in software systems 2014 - a systematic literature review. Software Engineering, IEEE Transactions on 40(3), 282–306 (March 2014)
8. Klaus, P., Böckle, G., van del Linden, F.: Software Product Line Engineering: Foundations, Principles and Techniques. Springer-Verlag New York, Inc. (2005)

Adaptability-based Service Behavioral Assessment

Diego Anabalón[1,3], Martin Garriga[1,3], Andres Flores[1,3], Alejandra Cechich[1], and Alejandro Zunino[2,3]

[1] GIISCo Research Group, Facultad de Informática,
Universidad Nacional del Comahue, Neuquen, Argentina.
[diego.anabalon, martin.garriga, andres.flores,
alejandra.cechich]@fi.uncoma.edu.ar
[2] ISISTAN Research Institute, UNICEN
Tandil, Argentina. azunino@isistan.unicen.edu.ar
[3] CONICET (National Scientific and Technical Research Council), Argentina.

Abstract. Building Service-oriented Applications implies the selection of adequate services to fulfill required functionality. Even a reduced set of candidate services involves an overwhelming assessment effort. In a previous work we have presented an approach to assist developers in the selection of Web Services. In this paper we detail its behavioral assessment procedure, which is based on testing and adaptation. This is done by using black-box testing criteria to explore services behavior. In addition, helpful information takes shape to build the needed adaptation logic to safely integrate the selected candidate into a Service-oriented Application. A concise case study shows the potential of this approach for both selection and integration of a candidate Web Service.

1 Introduction

Service-oriented Applications imply a business facing solution that consumes services from one or more providers and integrates them into the business process [13]. Although developers do not need to know the underlying model and rules of a third-party service, its proper reuse still implies quite a big effort. Yet searching for candidate services is mainly a manual exploration of Web catalogs usually showing poorly relevant information [12]. Even a favorable search result requires skillful developers to deduce the most appropriate service to be selected for subsequent integration tasks. The effort on assessing candidate services could be overwhelming. Not only services interfaces must be assessed, but also their operational behavior as key feature of a service contract. Besides, correct adaptations must be identified so client applications may safely consume services while enabling loose coupling for maintainability.

To ease the development of Service-oriented Applications we presented in previous work [3,6] a proposal to assist developers on service selection by means of testing and adaptation. This approach complements the conventional compatibility assessment by using black-box testing criteria to explore services behavior. The aim is to fulfill the *observability* testing metric [8,1] that observes a service operational behavior by analyzing its functional mapping of data transformations (input/output). In addition, a helpful information takes shape concerning the adaptation logic to integrate a service

into a client application. Hence, a wrapping algorithm was defined based on *mutation testing* [4,9], to identify the right adapter configuration. However, mutation testing carries a high effort (cost) both on generation and execution.

In this work, we improve the wrapping algorithm based on a set of adaptability factors recently defined [3]. In this way, we were able both to be more accurate on setting the best adapter and to highly reduce the involved costs on mutation testing. A concise case study shows the potential of improvements implemented into our approach.

The rest of the paper is organized as follows. Section 1 presents an overview of the *Selection Method*. Section 3 explains the steps to build a *Behavioral TS*. Section 4 briefly describes the *Interface Compatibility* procedure. Section 5 describes the *Behavioral Compatibility* procedure. Section 6 presents related work. Conclusions and future work are presented afterwards.

2 Service Selection Method

During development of Service-Oriented Applications, specific parts of the system may be implemented in the form of in-house components. Besides, some of the comprising software pieces could be fulfilled by the connection to Web Services. A set of candidate services could be obtained by making use of any service discovery registry. Even with a wieldy candidates' list, a developer must be skillful enough to determine the most appropriate service for the consumer application. Figure 1 shows our proposal to assist developers in the selection of Web Services, which is briefly described as follows:

As an initial step, a simple specification is needed, in the form of a required interface I_R, as input for the three comprising procedures.

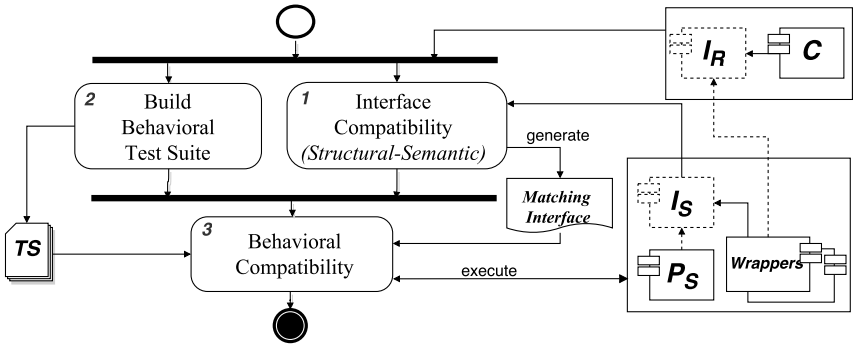


Fig. 1: Service Selection Method

The *Interface Compatibility* procedure (step 1) matches the required interface (I_R) and the interface (I_S) provided by a candidate service S. A structural-semantic analysis is performed to characterize operation signatures (return, name, parameters, exceptions) at four compatibility levels: *exact*, *near-exact*, *soft*, *near-soft*. This analysis also considers adaptability factors to reduce the integration effort. The outcome of this step is an *Interface Matching* list where each operation from I_R may have a matching with one or

more operations from I_S [3]. Particularly, operations from I_R with multiple matchings are considered as "*conflictive operations*" in this approach – i.e., they must be disambiguated yet.

When a functional requirement (I_R) from an application can be fulfilled by a potential candidate Web Service, a *Behavioral Test Suite* (TS) is built (step 2) [6]. This TS describes the required messages interchange from/to a third-party service, upon a selected testing coverage criteria [8,1], to fulfill the *observability* testing metric.

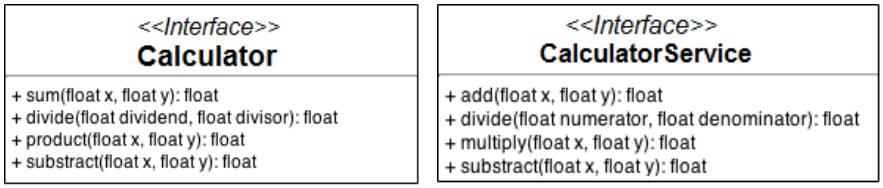
The *Behavioral Compatibility* procedure (step 3) evaluates the required behavior of candidate Web Services by executing the *Behavioral TS*. For this the *Interface Matching* list is processed to generate a set of wrappers W (adapters) – based on the identified *conflictive operations* – allowing to run the TS against the candidate service S.

After exercising the TS against each wrapper $w \in W$, at least one wrapper must successfully pass most of the tests to confirm both the proper matching of *conflictive operations* and the behavioral compatibility of the candidate service S. Besides, such successful wrapper allows an in-house component to safely call service S once integrated into the client application.

Next sections provide detailed information particularly related to the aforementioned procedures. A simple example will be used to illustrate the usefulness of the Selection Method.

2.1 Proof-of-Concept

To illustrate our proposal, we assume a simple example of a calculator application, with the four basic arithmetic operations. Figure 2a shows the required interface (I_R) called Calculator and Figure 2b shows the interface (I_S) of a candidate Web Service named CalculatorService.



(a) Required Interface Calculator (b) Service Interface CalculatorService

Fig. 2: Case Study of Calculator service

3 Behavioral Test Suite

In order to build a TS as a behavioral representation of services, specific coverage criteria for component testing has been selected [6]. The goal of this TS is to check that a candidate service S with interface I_S coincides on behavior with a given specification described by a required interface I_R. Therefore, each test case in TS will consist of a set

of calls to I_R's operations, from where the expected results were specified to determine acceptance or refusal when the TS is exercised against S (through I_S).

The *Behavioral TS* is based on the *all-context-dependence* criterion [8,1], where synchronous events (e.g., invocations to operations) and asynchronous (e.g., exceptions) may have sequential dependencies on each other, causing distinct behaviors according to the order in which operations and exceptions are called. The criterion requires traversing each operational sequence at least once. In our approach, this is called "interaction protocol" [2], formalized by using *regular expressions*, which allows to automatize test case generation. The alphabet for regular expressions comprise the signature of service operations.

In addition, an imperative specification must be built to describe the expected behavior of the interface I_R, with a set of representative test data. This is called *shadow class* and takes the same name as I_R. Hence, each test case uses these test data as input for parameters on each call to operations of the I_R's interface. This means a black box relationship or input/output functional mapping.

3.1 TS for Calculator

For the interface (I_R) Calculator, a shadow class was defined using the values 0 and 1 as *test data* to the four arithmetic operations. Then, the *interaction protocol* (in the form of a regular expression) is defined as follows:

Calculator (sum | subtract | product | divide)

This regular expression implies operational sequences limited to an only operation to be invoked, since Calculator is a *stateless* service without dependencies between operations. A set of *test templates* is generated from the regular expression, representing each operational sequence. In this case, 4 test templates are derived, each one composed of the constructor operation and one arithmetic operation.

Then, the selected test data is combined with the 4 test templates to generate a TS in a specific format: based on the MuJava framework [10]. From this combination, 8 test cases were generated in the form of methods into a test file called MujavaCalculator. Code Listing 1.1 shows the test case testTS_3_1, which invokes the sum operation.

Listing 1.1: MuJava test case for Calculator

```
public String testTS_3_1() {
        calc.calculator obtained=null;
        obtained = new calc.calculator();
        float arg1 = (float) 0;
        float arg2 = (float) 1;
        float result0 = obtained.sum(arg1,arg2);
        return result0.toString();
}
```

4 Interface Compatibility

In the *Interface Compatibility* procedure is determined the level of compatibility between the operations of the interface I_R and the operations of the interface I_S of a

candidate service S [3]. A structural-semantic analysis is performed to operation signatures. Structural aspects consider signatures and data types, while semantic aspects consider identifiers and terms in the names of operations and parameters. Information Retrieval (IR) techniques and the WordNet[4] dictionary are used for semantic aspects. A scheme of constraints allows to characterize pairs of operations ($op_R \in I_R$, $op_S \in I_S$) in four compatibility levels: *exact*, *near-exact*, *soft* and *near-soft*. Such constraints describe similarity cases based on adaptability (structural and/or semantic) conditions for each element of an operation signature (return, name, parameters, exceptions). As a result an *Interface Matching* list is generated, where each operation $op_R \in I_R$ may have a match to one or more operations $op_S \in I_S$, with likely one o more matchings in the parameters list.

In some cases, certain required operations ($op_R \in I_R$) could obtain multiple matchings (with the same compatibility) – at level of operations and/or parameters – to the candidate service interface (I_S). At *operation level*: an op_R has matching to several op_S. At *parameters level*: an op_R has several matchings in the parameters list – i.e., a set of all possible permutations of arguments. These operations need a disambiguation and they are called *"conflicting operations"* in this approach.

For non-conflictive operations it is possible to assume a high reliability in the operation matching – i.e., they may confirm their compatibility through the *Behavioral Compatibility* procedure.

4.1 Calculator-CalculatorService Interface Matching

Table 1 shows the interface matching result for Calculator and CalculatorService. Operations sum and product of Calculator are identified as *conflictive operations* at operation level. They obtained three matchings with operations add, subtract and multiply of CalculatorService, with the same level of compatibility *near-soft* (*n_soft_55*). Operations subtract and divide of Calculator are non-conflictive operations. They obtained a unique correspondence of higher compatibility level to their homonyms from CalculatorService – i.e., *exact* match for subtract operation and *near-exact* (*n_exact_3*) match for divide operation.

Moreover, all operations obtained a unique matching at parameters level. Parameters $(float\,x, float\,y)$ of operations sum, subtract and product of Calculator are identical (in name and type) to their counterparts of CalculatorService. For divide operation of Calculator, its parameters have identical types and equivalent (synonyms) names – $dividend$ with $numerator$ and $divisor$ with $denominator$ – with the operation of CalculatorService.

5 Behavior Compatibility

To carry out the *Behavior Compatibility* evaluation for a candidate service S, a set of *wrappers* (adapters) W needs to be built to allow executing the *Behavioral TS* and compare their results with those specified in the interface I_R. The wrappers set is generated

[4] https://wordnet.princeton.edu/

Table 1: Interface Compatibility for Calculator-CalculatorService

Calculator	**CalculatorService**		
float subtract (float x, float y)	[1, exact, float subtract (float x, float y)] {(x:float-x:float),(y:float-y:float)}	[109, n_soft_55, float add (float x, float y)]	[109, n_soft_55, float multiply (float x, float y)]
float sum (float x, float y)	[109, n_soft_55, float add (float x, float y)] {(x:float-x:float),(y:float-y:float)}	[109, n_soft_55, float subtract (float x, float y)] {(x:float-x:float), (y:float-y:float)}	[109, n_soft_55, float multiply (float x, float y)] {(x:float-x:float), (y:float-y:float)}
float divide (float dividend, float divisor)	[4, n_exact_3, float divide (float numerator, float denominator)] {(dividend:float-numerator:float), (divisor:float-denominator:float)}	[116, n_soft_62, float add (float x, float y)]	[116, n_soft_62, float subtract (float x, float y)]
float product (float x, float y)	[109, n_soft_55, float add (float x, float y)] {(x:float-x:float),(y:float-y:float)}	[109, n_soft_55, float subtract (float x, float y)] {(x:float-x:float), (y:float-y:float)}	[109, n_soft_55, float multiply (float x, float y)] {(x:float-x:float), (y:float-y:float)}

by processing the *Interface Matching* list, according to the multiple correspondences from the *conflictive operations* identified – both at operation and parameters levels. Hence, those multiples correspondences could be disambiguated so to identify proper univocal correspondences.

Wrappers generation can be seen as applying the Interface Mutation technique [4,9], by using a mutation operator to change invocations to operations and to change arguments in the parameters list. Thus, each wrapper is considered a faulty version (or mutant) regarding the wrapper that contains the proper matchings of operations and parameters.

Previously [6], our approach was only based on structural aspects (signatures and data types) to generate wrappers, producing a larger set of wrappers W. This is because usually a larger number of *conflictive operations* were identified – both at operation and parameters levels.

A major improvement in this work involves to consider the semantic aspects provided by the *Interface Matching* list, in which a less number of *conflictive operations* is identified, effectively reducing the W set.

5.1 Wrappers Generation

A tree structure is built to generate wrappers, where each path from the root to a leaf node represents a specific matching between operations of I_R and I_S (i.e., a wrapper to be generated). Thus, the number of leaf nodes determines the size of the wrappers set W. Each *conflictive operation* produces several branches on the tree. On the contrary, a non-conflictive operation (implying a univocal match) does not involve additional branches in the tree.

In the case of a *conflictive operation* at operation level, a new branch is added for each matching to a service operation. At parameters level, a new branch is added for each arguments matching from the set of permutations – even though there could be a univocal operation matching.

Particularly, in this work was updated the algorithm that implements the mutation operator to change arguments into the wrappers generation. Thereby, the new algorithm to treat parameters matchings considers the following cases:

1. *Without any matching*: if any matching was identified at all (structural and/or semantic), parameters will be permuted between each other, producing branches for each arguments combination.

2. *Only structural matching*: If a semantic matching was not identified, parameters are related only through the structural information (data types). If multiple matchings were identified, for each of them a branch is produced. For the remaining parameters the case 1 is applied.

3. *Structural-semantic matching*: Parameters are related through the structural and semantic information. If multiple matchings were identified, for each of them a branch is produced. For the remaining parameters the case 2 is applied.

4. *Service's extra parameters*: If a service operation contains more parameters than the required operation, then some parameters are left outside of the matchings. For them, a test value is required when invoking the service operation. Hence, in this approach a *default* value is assigned according to each parameter data type – "" (quotes) for strings, ' ' (space character) for characters, *true* value for booleans, and 0 (zero) for numerical types.

5.2 Wrappers for Calculator-CalculatorService

Figure 3 shows the wrapper generation tree for Calculator and CalculatorService. Branches were only produced at operation level according to the *conflictive operations* identified: sum y product of Calculator with respect to add, subtract and multiply of CalculatorService. Regarding to parameters matching, the case 3 was applied since a structural-semantic matching was identified for all parameters.

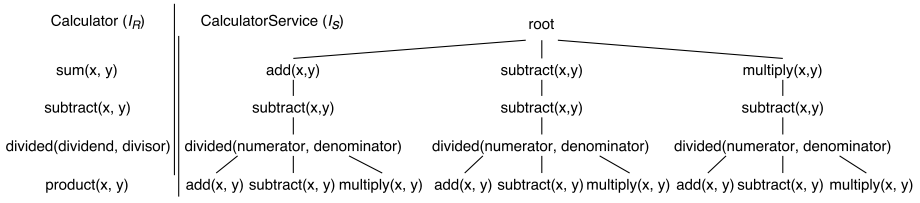


Fig. 3: Wrapper generation tree to Calculator Service

The total number of wrappers (size of W) to be generated is 9, which is the number of leaves on the tree. Notice that without considering semantic aspects, particularly for parameters, a major number of permutations there had been generated. Since all parameters are of the same type, multiple structural matchings there had been identified, making the size of W scaling to 144 wrappers.

Listing Code 1.2 and 1.3 show a fragment of the code from *wrapper2* and *wrapper3* respectively. Where *wrapper2* represents both the tree path down-to the third leaf node and the most appropriate matchings. Likewise, *wrapper3* represents the path down-to the fourth leaf node – being a faulty (mutant) version.

Listing 1.2: Wrapper2 for Calculate-CalculateService

```
public class Calculator{
        protected katze. ... .CalculatorService proxy = null;
        public Calculator(){
                this.proxy = new katze. .. .CalculatorService();
        }
        public float sum (float arg1, float arg2){
                float ret0;
                try{ ret0= candidate.add(arg1,arg2);
                }catch (exception ex){
                        ex.printStackTrace();
                        throw new RuntimeException(ex);
                }
                return ret0;
        }
//...
        public float product (float arg1, float arg2) {
                float ret0;
                try{ret0 = candidate.multiply(arg1, arg2);
                }catch (exception ex){
                        ex.printStackTrace();
                        throw new RuntimeException(ex);
                }
        }
                return ret0;
}
```

5.3 Wrappers Evaluation

Once generated the set of wrappers W, the *Behavior TS* is executed against each wrapper $w \in W$ to assess the behavior of the candidate service S. Using our tool based on the MuJava framework, the TS is exercised against the I_R and iterating over the list of wrappers. After that, results are compared to determine for each wrapper the number of test cases that failed – which produced a result different from the one expected. A wrapper may survive (as mutation case) when most of the test cases are successful. A successful wrapper allows to disambiguate the *conflictive operations,* confirming the right matchings both at operation and parameters levels. In addition, this wrapper may be used as integration artifact allowing a safe communication to the candidate service S.

5.4 Behavioral Evaluation for Calculator-CalculatorService

The TS called *MujavaCalculator* was executed against Calculator (I_R) and the 9 wrappers generated for CalculatorService. Table 2 shows the execution results, where *wrapper2* passed successfully 100% allowing to confirm the behavioral compatibility of CalculatorService. In addition, this wrapper contains the right matchings of operations (sum-add, subtract-subtract, divide-divide, product-multiply). Finally, *wrapper2* can be used as an adapter for the safe integration of CalculatorService in the client application.

6 Related work

Due to lack of space this section briefly presents related work without a detailed comparison with our approach.

In [7] we survey current approaches on selection, testing and adaptation of services with focus on composition. Service selection approaches are closely related to discovery, in which IR techniques and/or a semantic basis (e.g., ontologies) are generally used.

Listing 1.3: Wrapper3 for Calculate-CalculateService

```
public class Calculator{
//...
        public float sum (float arg1, float arg2){
                float ret0;
                try{ ret0= candidate.substract(arg1,arg2);
                }catch (exception ex){
                        ex.printStackTrace();
                        throw new RuntimeException(ex);
                }
                return ret0;
        }
//...
        public float product (float arg1, float arg2) {
                float ret0;
                try{ret0 = candidate.add(arg1, arg2);
                }catch (exception ex){
                        ex.printStackTrace();
                        throw new RuntimeException(ex);
                }
                return ret0;
        }
}
```

Table 2: Execution results of TS for Calculator-CalculatorService

Wrappers	**Test Cases**		
	successful	*failed*	*success rate*
wrapper3, wrapper4, wrapper6, wrapper7	0	4	0
wrapper0, wrapper1, wrapper5, wrapper8	2	2	50
wrapper2	4	0	100

Service evaluation mainly use WSDL documents and/or XML schemes of data types, or even WSDL-based ad-hoc enriched specifications. Service implementation may also affect its evaluation: *contract-first* services are designed prior to code, improving their WSDL descriptions; *code-first* services use automatic tools to derive WSDL documents from source code, reducing their description quality.

Regarding service testing, the work in [2] presents a survey of approaches that use strategies of verification and software testing. Some of them evaluate individual operations of atomic services, others also use a semantic basis such as OWL-S, and others evaluate a group of services that could interact in a composition.

The work in [5] presents an overview on service adaptation, at service interface and business protocol levels. This is required even though the Web Service standardization reduces the heterogeneity and simplifies interaction. At interface level adaptations deal with operation signatures, that implies perform message transformations or data mapping. At business protocol level, services behavior is affected on the order constraints of the message exchange sequences – such as deadlock and non-specified reception.

7 Conclusions and Future Work

In this paper we have presented an approach to assist developers in the selection of services, when developing a Service-oriented Application. Particularly, our approach addresses two main aspects. On the one side, confirming the suitability of a candidate service by a dynamic behavioral evaluation (execution behavior), in which the applied

testing criteria increase the reliability level. On the other side, effectively building the right adaptation logic for a selected Web Service, while reducing the adaptation and integration effort.

Currently, we are working on service compositions [7]. This is particularly useful when a single service cannot provide all the required functionality. In this context, it is necessary to generate software artifacts (e.g., tests and adapters) according to specifications in business process languages such as BPEL and BPML [14]. Finally, another interesting extension of this work is to automatically derive software artifacts from system models – for example from models described in SoaML [11], a UML profile for modeling Service-oriented Applications.

References

1. Xiaoying Bai, Wenli Dong, W-T Tsai, and Yinong Chen. Wsdl-based automatic test case generation for web services testing. In *Service-Oriented System Engineering, 2005. SOSE 2005. IEEE International Workshop*, pages 207–212. IEEE, 2005.
2. M. Bozkurt, M. Harman, and Y Hassoun. Testing and verification in service-oriented architecture: a survey. *Software Testing, Verification and Reliability*, 23(4):261–313, 2013.
3. A. De Renzis, M. Garriga, A. Flores, A. Zunino, and A. Cechich. Semantic-structural assessment scheme for integrability in service-oriented applications. In *Latin-american Symposium of Enterprise Computing, held during CLEI'2014*, September 2014.
4. M. Delamaro, J. Maidonado, and A. Mathur. Interface mutation: An approach for integration testing. *IEEE Transactions on Software Engineering*, 27(3):228–247, 2001.
5. Maryam Eslamichalandar, Kamel Barkaoui, and Hamid Reza Motahari-Nezhad. Service composition adaptation: An overview. *2nd IEEE IWAISE*, page 20À7, 2012.
6. M. Garriga, A. Flores, A. Cechich, and A Zunino. Behavior assessment based selection method for service oriented applications integrability. In *Proceedings of the 41st Argentine Symposium on Software Engineering*, ASSE '12, pages 339–353, La Plata, BA, Argentina, 2012. SADIO.
7. Martin Garriga, Andres Flores, Alejandra Cechich, and Alejandro Zunino. Web services composition mechanisms: A review. *IETE Technical Review*, In press, 2015.
8. M. Jaffar-Ur Rehman, F. Jabeen, A. Bertolino, and A. Polini. Testing Software Components for Integration: a Survey of Issues and Techniques. *Software Testing, Verification and Reliability*, 17(2):95–133, June 2007.
9. Jia, Y. y Harman, M. An Analysis and Survey of the Development of Mutation Testing. *IEEE Transactions on Software Engineering*, 37(5):649–678, 2011.
10. μJava Home Page. Mutation system for Java programs, 2008. http://www.cs.gmu.edu/ offutt/mujava/.
11. OMG. Service oriented architecture modeling language (soaml) specification. Technical report, Object Management Group, Inc., 2012. http://www.omg.org/spec/SoaML/1.0.1/PDF/.
12. M. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann. Service-oriented computing: A research roadmap. *International Journal of Cooperative Information Systems*, 17(02):223–255, 2008.
13. D. Sprott and L. Wilkes. Understanding Service-Oriented Architecture. *The Architecture Journal. MSDN Library. Microsoft Corporation*, 1:13, January 2004. http://msdn.microsoft.com/en-us/library/aa480021.aspx.
14. S. Weerawarana, F. Curbera, F. Leymann, T. Storey, and D. Ferguson. *Web Services Platform Architecture: SOAP, WSDL, WS-Policy, WS-Addressing, WS-BPEL, WS-Reliable Messaging, and More*. Prentice Hall PTR, 2005.

Un Análisis comparativo de rendimiento en Aplicaciones Móviles Multiplataforma

Lisandro Delía[1], Nicolás Galdamez[1], Leonardo Corbalan[1], Pablo Thomas[1]
Patricia Pesado[1,2]

[1] Instituto de Investigación en Informática LIDI. Facultad de Informática.
Universidad Nacional de La Plata. Argentina
[2] Comisión de Investigaciones Científicas de la Provincia de Buenos Aires. Argentina

{ldelia, ngaldamez, corbalan, pthomas, ppesado}@lidi.info.unlp.edu.ar

Resumen. Es razonable pensar que la comunidad de desarrolladores de aplicaciones móviles opte por plataformas nativas cuando pretende obtener mejor rendimiento en relación al tiempo de ejecución en sus productos. No obstante, existe la posibilidad de optar por un desarrollo multiplataforma, a priori con más restricciones, pero con mayores beneficios en cuanto a portabilidad y escalabilidad. En este trabajo se realiza un análisis comparativo de rendimiento de distintos modos de desarrollo multiplataforma, a través de un cálculo numérico simple.

Palabras claves: dispositivos móviles, aplicaciones móviles multiplataforma, aplicaciones móviles nativas, performance.

1 Introducción

El desarrollo de software para dispositivos móviles tiene características particulares. La necesidad de tratar con diversas plataformas, estándares, protocolos y tecnologías de red; las capacidades limitadas de los dispositivos y las exigencias de tiempo del mercado, son sólo algunos de los problemas a tratar. Por ello, el desarrollo de software para dispositivos móviles difiere considerablemente del tradicional [1].

Cuando se desarrolla una aplicación para dispositivos móviles, la expectativa es que sea ejecutable en la mayor cantidad de dispositivos posible. Para ello existen dos alternativas: desarrollar en forma "nativa" para cada plataforma del mercado, es decir, utilizar un entorno de desarrollo integrado, un lenguaje y herramientas propias de cada plataforma [2]; o construir aplicaciones que puedan ejecutarse en cualquier plataforma, es decir, que sean multiplataforma.

La elección del modo (nativo o multiplataforma) de desarrollo de aplicaciones para dispositivos móviles depende de varios factores. Uno de ellos, en ocasiones primordial, es el tiempo de ejecución.

Si el objetivo es priorizar el rendimiento de la aplicación sobre otros factores que inciden en la producción de software, el desarrollo nativo se presenta como la opción natural [5] [6] [7]. No obstante, las aplicaciones nativas no son portables y la escalabilidad se reduce al mismo sistema operativo para las cuales fueron creadas. Por

lo tanto, es razonable optar por desarrollos multiplataforma, aunque teniendo en cuenta el rendimiento en término de tiempo, entre otros factores, de las alternativas posibles.

En artículos anteriores [5] [6] [7] hemos analizado ventajas y desventajas de las alternativas mencionadas, desde el punto de vista del Ingeniero de Software. En el presente trabajo nos focalizamos sobre el rendimiento relacionado con la velocidad de procesamiento. Para ello hemos efectuado comparaciones de tiempos de ejecución de distintas aplicaciones multiplataforma, avanzando así en la línea de investigación iniciada en [5].

En la sección 2 se describe el experimento utilizado para comparar tiempos de ejecución entre diversas aplicaciones móviles multiplataforma, en la sección 3 se muestran y analizan los resultados obtenidos y en las secciones posteriores se presentan las conclusiones y trabajo futuro.

2 Experimento

Los Sistemas Operativos Móviles elegidos para la experimentación fueron Android e iOS, dado que a Junio del 2015 entre ambos abarcaban el 80% del mercado mundial, situación también reflejada en la Argentina, en los dos casos de acuerdo a lo indicado en [8].

Se seleccionaron las plataformas de desarrollo multiplataforma, considerando las alternativas presentadas en [6] [7]. Se optó por un desarrollo híbrido utilizando Apache Cordova [9], una aplicación interpretada en Appcelerator Titanium [10], y una generada por compilación cruzada utilizando Xamarin [11]. De este modo se logró una muestra razonablemente representativa de las diversas opciones existentes en la actualidad. Las aplicaciones web móviles, también consideradas multiplataforma, quedaron excluidas de este experimento.

El conjunto de dispositivos móviles utilizados durante la experimentación se describe en la tabla 1. Cada uno de ellos representa un escenario de prueba sobre el que se establecieron los distintos casos multiplataforma a evaluar.

Tabla 1. Dispositivos móviles seleccionados para el experimento.

Escenario de prueba	Dispositivo	Marca	Modelo	Sistema Operativo
1	Smartphone	Samsung	Advance, Nº GT - I9070	Android 4.1.2
2	Smartphone	Samsung	S3 NEO, Nº I9301I	Android 4.4.2
3	Smartphone	Motorola	Moto-G	Android 5.0.2
4	Smartphone	Motorola	Moto-G2	Android 4.4
5	Tablet	Samsung	Tab 2	Android 4.2.2
6	Smartphone	Apple	Iphone 5S	iOS 8.4

En resumen, se realizó el experimento en dos sistemas operativos (cinco versiones

distintas en el caso de Android), tres ambientes multiplataforma, y seis dispositivos móviles diferentes.

Con el fin de evaluar la velocidad de procesamiento, se planteó un cálculo simple que incluyó varias iteraciones, funciones matemáticas y aritmética de punto flotante y se resume en la siguiente serie:

$$serie = \sum_{j=1}^{5} \sum_{k=1}^{100000} (\log_2(k) + \frac{3k}{2j} + \sqrt{k} + k^{j-1} \quad (1)$$

A modo de ejemplo a continuación se muestra el código multiplataforma desarrollado en Apache Cordova.

```
var inicio = new Date().getTime();
var serie = 0;
for ( var j=1; j <= 5; j++ )
{
    for ( var k=1; k <= 100000; k++ )
   {
     serie = serie + (Math.log(k)/Math.LN2) + (3*k/2*j) +
             Math.sqrt(k) + Math.pow(k, j-1);
      }
   }
var fin = new Date().getTime();
var tiempo = fin - inicio;
document.getElementById('resultado').innerHTML = tiempo +
' -> ' + serie;
```

El experimento planteado es simple pero efectivo y permite medir con precisión la variable analizada, en este caso, el tiempo de ejecución requerido para realizar cálculo intensivo. En la siguiente sección se analizan los resultados obtenidos.

3 Resultados obtenidos

A fin de obtener resultados verosímiles, para cada caso de prueba definido, se realizaron 30 ejecuciones independientes del experimento diseñado obteniendo una muestra T = T_1, T_2, … T_{30}, con T_i = tiempo requerido para el cálculo de la serie en la *i*-ésima ejecución del experimento planteado. El tiempo T_i se expresa en milisegundos.

Para caracterizar cada una de las muestras obtenidas, se han calculado los estadísticos $\bar{T}$ y S que se corresponden con la media (o promedio muestral) y la desviación estándar muestral (ver Tabla 2).

En la tabla 3 se presenta una síntesis de los resultados obtenidos conformada por los valores $\bar{T}$ y S calculados para cada caso de prueba planteado. Estos valores deben ser contrastados entre los distintos modos de desarrollo multiplataforma dentro de un

mismo escenario de prueba. Aunque estos dos estadísticos conforman una buena caracterización de la muestra sobre la que se calculan, los diagramas de caja presentados en la figura 1 complementan esa caracterización ofreciendo mayor información sobre la distribución de los datos en cada muestra.

Tabla 2. Estadísticos empleados en el análisis de los datos

Dada la muestra T=T_1, T_2, …, Tn	
Media o promedio muestral	$\bar{T} = \left(\frac{1}{n}\right)\sum_{i=1}^{n} T_i$
Desviación estándar muestral	$S = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(T_i - \bar{T})^2}$

Tabla 3. Síntesis de resultados obtenidos

		Apache Cordova	**Titanium**	**Xamarín**
Samsung Advance	$\bar{T}$	228,07	237,40	468,20
	S	14,43	13,01	8,98
Samsung S3 Neo	$\bar{T}$	596,10	190,10	339,93
	S	7,35	7,33	10,45
MOTO-G	$\bar{T}$	189,43	286,23	379,20
	S	6,61	24,04	5,55
MOTO-G 2	$\bar{T}$	230,33	211,67	395,17
	S	14,22	24,95	8,95
Samsung Tab 2	$\bar{T}$	190,60	192,70	379,33
	S	9,36	16,80	8,31
Iphone 5S	$\bar{T}$	287,90	267,20	98,47
	S	7,10	2,78	2,97

Un diagrama de caja consiste en un rectángulo dispuesto en forma horizontal, también puede ser en sentido vertical, que representa el recorrido desde el primer cuartil hasta el tercero. Este rectángulo, sobre el que recaen la mitad de los datos muestreados, está dividido por un segmento vertical que indica donde se posiciona la mediana (segundo cuartil) haciendo visible su relación con los cuartiles primero y tercero. La caja se ubica a escala sobre un segmento que tiene como extremos los valores mínimo y máximo de la muestra.

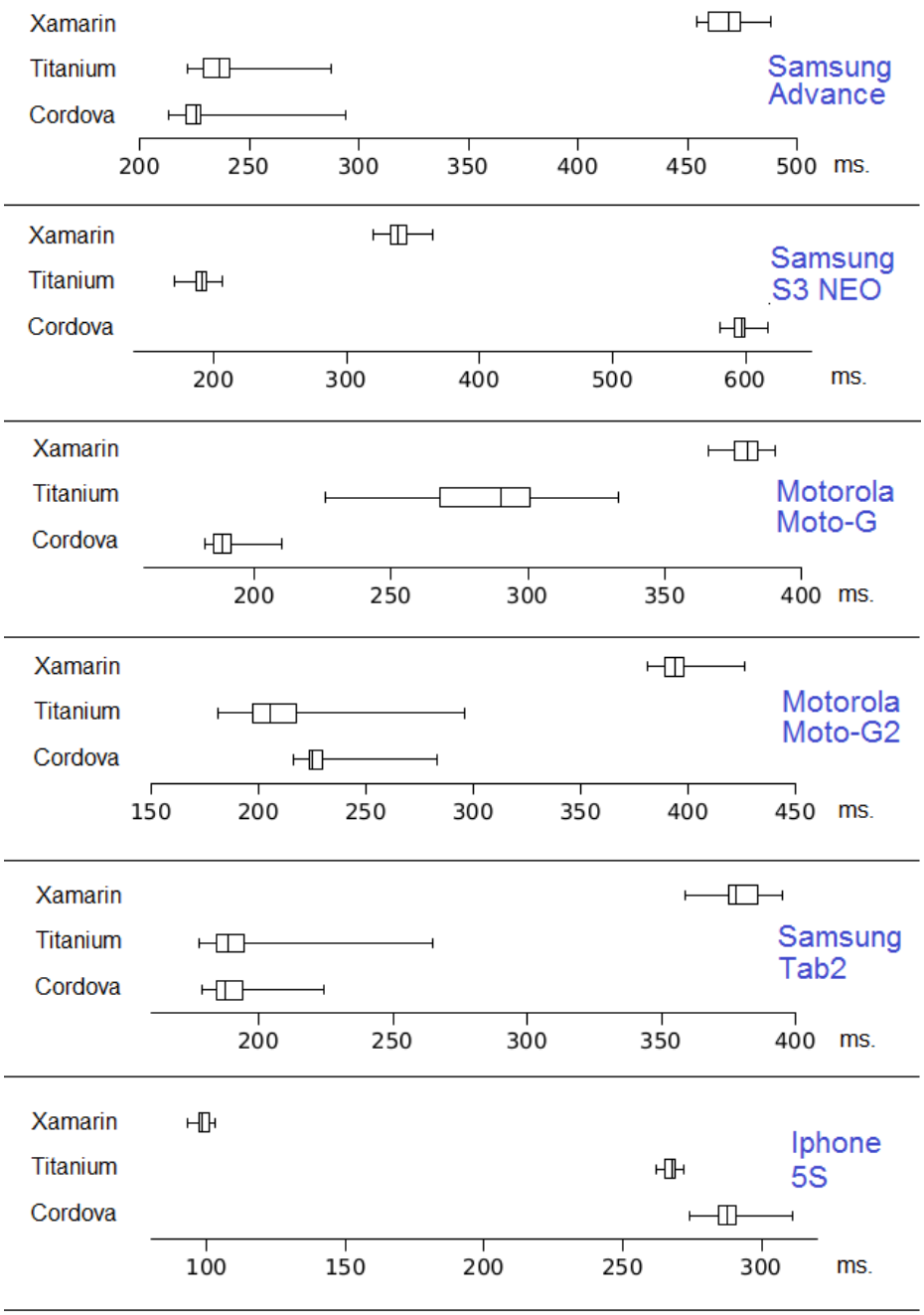


Figura 1. Diagrama de cajas con las muestras recolectadas en los seis escenarios de prueba planteados.

Los diagramas de caja son sumamente útiles pues permiten una rápida comparación visual sobre la dispersión de los datos en cada una de las muestras representadas.

Según se aprecia en la figura 1, en la mayoría de los casos las muestras obtenidas resultaron suficientemente separadas unas con otras. Ello es prueba de que las diferencias observadas sobre $\bar{T}$ en cada escenario de prueba (ver Tabla 3) son significativas.

Sin embargo, en los escenarios 1, 4 y 5, Samsung Advance, Motorola Moto-G2 y Samsung Tab2 respectivamente, las muestras obtenidas para los modos de desarrollo multiplataforma Apache Cordova y Titanium presentan un marcado solapamiento haciendo más difícil el análisis visual a partir de los diagramas de caja. Por lo tanto es necesario establecer si tales diferencias son estadísticamente significativas por medio de un test de hipótesis sobre diferencia de medias.

Debido a que las muestras obtenidas por la experimentación llevada a cabo presentan una distribución cercana a la distribución Normal (se aplicó la prueba para normalidad de Kolmogorov-Smirnov), es posible utilizar el test *t-Student* para diferencias de medias a partir de muestras independientes.

Básicamente el test *t-Student* para diferencia de medias postula como hipótesis nula que ambas medias son iguales y como hipótesis alternativa que son distintas. Como resultado de aplicar el test *t-Student* se obtiene el p-valor que representa la probabilidad de cometer error de tipo I, es decir de rechazar la hipótesis nula cuando ésta es verdadera. Consecuentemente, cuanto más pequeño sea el p-valor, mayor evidencia habrá a favor del postulado que las medias son distintas. En general, hay consenso en la comunidad científica en que si p-valor < 0,05 debe reconocerse que la diferencia de medias observada es estadísticamente significativa.

La tabla 4 resume el análisis estadístico llevado a cabo sobre los escenarios y los modos de desarrollo indicados

Tabla 4. Prueba de significancia estadística *t-Student*

	Apache Cordova $\bar{T}$ (*S*)	**Titanium** $\bar{T}$ (*S*)	**p-valor**	**Resultado**
Samsung Advance	228,07 (14,43)	237,40 (13,01)	0,01091	ES
Motorola Moto-G2	230,33 (14,22)	211,67 (24,95)	0,00087	EES
Samsung Tab 2	190,60 (9,36)	192,70 (16,80)	0,55271	ENS

ES = Estadísticamente Significativo. ENS = Estadísticamente No Significativo.
EES = Extremadamente Estadísticamente Significativo.

A partir de los datos presentados en la tabla precedente y demás pruebas *t-Student* realizadas, se puede afirmar que todas las diferencias en los promedios muestrales presentados en este trabajo son estadísticamente significativas a excepción de la correspondiente a Apache Cordova y Titanium en el escenario 5 (Samsung Tab2). En este último caso no se ha encontrado evidencia suficiente para sostener que las

aplicaciones realizadas con Apache Cordova mejoran el rendimiento de aquellas construidas con Titanium, o viceversa.

La figura 2 presenta en forma de diagrama de barras la información respecto del promedio muestral para cada escenario de prueba y modo de desarrollo multiplataforma evaluado. Estos valores se corresponden con los descriptos en la tabla 3.

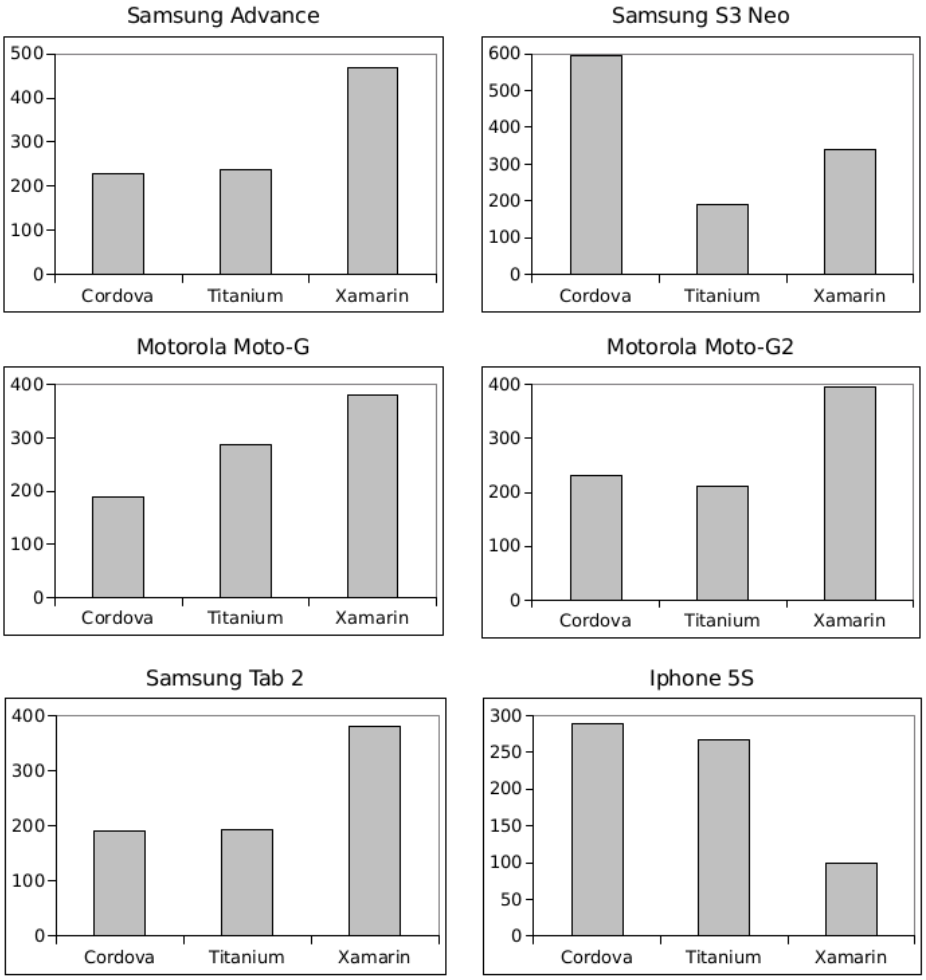


Figura 2. Diagrama de barras que visualiza la media del tiempo de ejecución empleado en el cálculo de la serie (1) observado en cada uno de los escenarios de prueba para los modos de desarrollo multiplataforma evaluados

Planteada la competencia entre Apache Cordova (desarrollo híbrido), Titanium (aplicación interpretada) y Xamarin (compilación cruzada), la siguiente tabla resume los resultados generales de esta contienda.

Tabla 5. Resultados generales

Escenario	Posiciones		
	Apacehe Cordova	Titanium	Xamarin
Samsung Advance	1º puesto	2º puesto	3º puesto
Samsung S3 Neo	3º. puesto	1º puesto	2º puesto
Motorola Moto-G	1º. puesto	2º puesto	3º puesto
Motorola Moto-G2	2º. puesto	1º puesto	3º puesto
Samsung Tab 2	1º. puesto	1º puesto	3º puesto
Iphone 5S	3º puesto	2º puesto	1º puesto

4 Conclusiones

Se ha presentado un estudio comparativo sobre el rendimiento de distintos modos de desarrollo multiplataforma en relación con el tiempo de ejecución de las aplicaciones construidas. Los escenarios de prueba diseñados incluyeron dos sistemas operativos (cinco versiones de Android y una de iOS) y diversos dispositivos (una *Tablet* y cinco *smartphones* distintos). Sobre dichos escenarios fueron puestos a prueba Apache Cordova, Titanium, y Xamarin (desarrollo híbrido, aplicación interpretada y generada por compilación cruzada respectivamente).

Planteada la competencia entre estos tres modos de desarrollo multiplataforma, Titanium ha mostrado el mejor rendimiento global. De los 6 escenarios configurados, obtuvo el primer puesto tres veces (uno compartido con Apache Cordova) y el segundo puesto las otras tres.

Apache Cordova representa el segundo mejor rendimiento general obteniendo el primer puesto tres veces (uno compartido con Titanium) una vez el segundo puesto y dos veces el tercero.

Finalmente Xamarin mostró el rendimiento general más bajo obteniendo cuatro veces el tercer puesto, una vez el segundo y una vez el primer puesto.

Es destacable que la compilación cruzada, por medio de su representante Xamarin, sólo haya obtenido el mejor rendimiento sobre el sistema operativo iOS mientras que en todos los escenarios de prueba basados en dispositivos Android obtuvo el rendimiento más bajo, salvo en una ocasión que, aún así, fue superado por Titanium.

Se concluye que, entre los tres modos de desarrollo multiplataforma puestos a prueba, cuando se prioriza el rendimiento en relación al tiempo de ejecución de las aplicaciones construidas, Titanium representa la mejor opción resultando siempre uno de los mejores tiempos de ejecución y comportándose de manera estable, sin grandes variaciones de rendimiento entre distintos dispositivos.

5 Trabajo futuro

Se plantea como línea de trabajo futuro profundizar el estudio y análisis sobre el desarrollo multiplataforma examinando vía experimentación el rendimiento de las

aplicaciones así construidas. Se plantea incorporar en este análisis a otros modos de desarrollo multiplataforma y establecer las pertinentes comparaciones con las aplicaciones desarrolladas en forma nativa. Se propone también la evaluación de otros aspectos de rendimiento, para cuantificarlo en relación al acceso a disco, al consumo de batería y en general a otras capacidades del dispositivo.

Referencias

1. Hayes, I. S. *Just Enough Wireless Computing*. Prentice Hall Professional Technical Reference. 2002. ISBN:0130994618
2. Spyros Xanthopoulos, Stelios Xinogalos, *A Comparative Analysis of Cross-platform Development Approaches for Mobile Applications*, BCI' 2013, Greece
3. Yonathan Aklilu Redda, *Cross platform Mobile Applications Development*, Norwegian University of Science and Technology, Norwegian University of Science and Technology, Norwegian University of Science and Technology, Master in Information Systems, June 2012.
4. Dalmasso I., Datta S.K., Bonnet C. Nikaein N., *Survey, comparison and evaluation of cross platform mobile application development tools,* Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International.
5. Delia L., Galdamez N. Thomas P., Pesado P., Un Análisis Experimental de Tipo de Aplicaciones para Dispositivos Móviles, XVIII Congreso Argentino de Ciencias de la Computación, CACIC 2013.
6. Delia, L.; Galdamez, N.; Thomas, P.; Corbalan, L.; Pesado, P.,, Análisis Experimental de desarrollo de Aplicaciones Móviles Multiplataforma, XX CACIC, Universidad Nacional de La Matanza, Octubre 2014.
7. Delia, L.; Galdamez, N.; Thomas, P.; Corbalan, L.; Pesado, P., *Multi-platform mobile application development analysis*, Research Challenges in Information Science (RCIS), 2015 IEEE 9th International Conference on, Atenas, Grecia, Año 2015.
8. http://gs.statcounter.com
9. http://cordova.apache.org
10. http://www.appcelerator.com
11. http://xamarin.com

Resolución más eficiente de dependencias Java

Martín Agüero[1], Luciana Ballejos[2]

[1] Tesista de Maestría en Ingeniería de Software – Facultad de Informática, UNLP
aguero.martin@gmail.com
[2] CIDISI – Centro de I+D en Ingeniería en Sistemas de Información – FRSF, UTN
lballejos@santafe-conicet.gov.ar

Resumen. En este trabajo se realiza una revisión general acerca de las características y modos de gestión de dependencias de código abierto para proyectos Java. Asimismo, también se desarrolla un estudio para establecer una tasa de utilización habitual respecto del total de recursos disponibles en cada dependencia. En base a los resultados obtenidos en las mediciones y el análisis de otros trabajos que también abordaron el tema, se propone cambiar la estrategia de sincronización completa de dependencias (repositorio local) por un middleware que interactúe entre el entorno de desarrollo y los repositorios públicos. Se plantea establecer un servicio que resuelva automáticamente los requerimientos de dependencias directas e indirectas y que atienda solicitudes puntuales de bytecode en tiempo de ejecución y de descriptores para la compilación. Para una siguiente etapa, se planea desarrollar el software propuesto a modo de prueba de concepto.

Palabras clave: java, open source, software library, apache maven, software engineering, software dependencies.

1. Introducción

La gestión de dependencias es una actividad central en el desarrollo de software. En la actualidad, el éxito de una plataforma está ligada, en gran parte, a la disponibilidad masiva de librerías[1] publicadas bajo licencia de código abierto [1]. Estos recursos suelen distribuirse desde el website de las comunidades que agrupan y promueven proyectos colaborativos, tales como las fundaciones Apache, Eclipse o Linux. Una de las fortalezas que ofrece el open source es que se trata de un modelo donde el esfuerzo de la comunidad genera un ecosistema que se nutre a sí misma, donde todos los participantes obtienen un beneficio común. A simple vista parece un modelo ideal, no obstante, los artefactos creados por las distintas comunidades padecen de una severa falta de compatibilidad e integración entre ellos [2]. En la mayoría de los casos, los autores ofrecen el código fuente desde un repositorio especializado y público como GitHub, SourceForge o Bitbucket y versiones compiladas desde el propio website de la comunidad. Esto también sucede en repositorios públicos como ibiblio, The Central Repository o MVN Repository.

Para el caso del software compilado con Java, el procedimiento estándar es distribuir los binarios (bytecode) comprimidos en archivos de formato ZIP, pero con

[1] Es una mala traducción de la palabra 'libraries', la correcta es 'bibliotecas'.

extensión JAR[2]. El empaquetado en archivos de extensión JAR no establece como obligatorio la creación de un manifiesto. Su presencia es optativa y puede contener meta-datos como una firma electrónica, control de versiones, declaración de dependencias o el punto de entrada (Main-Class). Para el caso de los bundles OSGi [3], que también son empaquetados en archivos JAR, el manifiesto es obligatorio. Allí se declara el nombre del bundle, identificador, versión, dependencias e interfaces. Esta tecnología es superadora de la especificación Java. No obstante, aún no consiguió posicionarse masivamente entre la comunidad y la industria, posiblemente por no cubrir aspectos como calidad, usabilidad y reusabilidad [4].

A fin de facilitar la gestión y acceso a dependencias para la compilación de proyectos Java, han surgido herramientas como Apache Maven que en la actualidad es el estándar de facto para todo desarrollo de software a gran escala [5]. Maven propone un modelo de descripción genérica del proyecto a través de un archivo POM (Project Object Model) donde se especifican las dependencias, jerarquías, ciclo de vida de la construcción, parámetros de configuración, casos de prueba y otros [6]. Para optimizar la gestión de dependencias, Maven crea un repositorio local donde se copian todos los archivos JAR requeridos por el proyecto. De este modo, todos los proyectos apuntan sus dependencias a la copia única, pudiendo establecer políticas a nivel particular o departamental, donde un grupo de desarrolladores comparte un único repositorio [7].

Esta estrategia de replicar las librerías –inclusive las transitivas– a un repositorio local ha probado ser exitosa, sin embargo, es tecnología que fue pensada en el año 2002, cuando recién se comenzaba a hablar de Web Services [8] y el cómputo en la nube todavía no era más que un símbolo en diagramas de diseño de redes [9].

La especificación de la Máquina Virtual de Java [10] define para el tiempo de ejecución al Class Loader como el responsable de cargar el bytecode en la máquina virtual (JVM) [11].

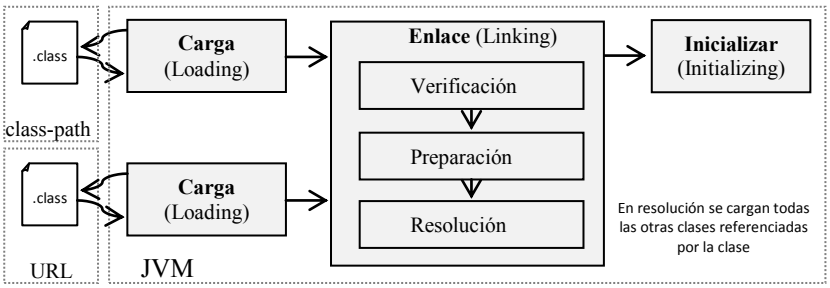


Fig. 1. Proceso de carga de clases en la JVM [12].

Durante este proceso, una instancia de la clase ClassLoader busca en el class-path la clase requerida y la copia a la JVM. Existen implementaciones de ClassLoader de lectura local y a través de la red, lo más habitual es que sea desde una fuente local (Ver Figura 1).

A simple vista este modelo parece ideal, no obstante tanto la industria como la academia han detectado problemas. Uno de ellos es el denominado "Jar Hell" que se da cuando en un mismo class-path conviven clases que comparten un mismo nombre o cuando distintas versiones de una clase deben coexistir en un mismo class-path [4].

[2] https://docs.oracle.com/javase/tutorial/deployment/jar/basicsindex.html

Otro problema que se da con frecuencia es la necesidad de contar también con copias locales de las dependencias transitivas (las dependencias de las dependencias) y con la versión apropiada [13]. Es cierto que Maven (y similares) ofrecen soluciones para los casos citados, mediante perfiles de compilación o automatizando la descarga de las dependencias transitivas [14]. Sin embargo, esta solución implica definir una serie de especificaciones no inherentes al proyecto en sí, además de tener que recuperar del repositorio central tanto la totalidad de las dependencias directas como así también las indirectas.

Este trabajo tiene como objetivo analizar el grado de utilización promedio de estas librerías por parte del software Java y proponer una alternativa al modelo de repositorio local. Por medio de un programa creado específicamente para medir la proporción entre recursos disponibles y empleados, se intentará demostrar que el nivel de referencia a recursos externos, por lo general, no supera la décima parte del total disponible en los archivos JAR. De forma análoga, se puede representar la situación como la de precisar disponer de un libro y por ello solicitar toda una fila de la estantería de la biblioteca. Asimismo en este paper, también se definirá un modelo conceptual que propondrá un servicio intermediario cuya función será la de atender en tiempo real solicitudes de bytecode por parte de la JVM y el compilador. Este sistema ubicará en los repositorios las librerías correspondientes y entregará al cliente sólo las clases solicitadas y sus transitivas. El modelo no está pensado para sustituir la disponibilidad local de dependencias en ambientes de producción, pero sí para ser una alternativa ágil y actual a emplear en instancias de evaluación de tecnología, desarrollo o pruebas.

A continuación, la sección 2 describe la situación actual y otras alternativas propuestas por la academia y la industria en el área. Luego, la sección 3 describe la herramienta desarrollada para el análisis de las dependencias y analiza las características de otras similares que fueron descartadas, además de explicar los motivos. En la sección 4 se presentan mediciones de software libre Java y la sección 5 describe un modelo conceptual de la propuesta. Finalmente, la sección 6 presenta las conclusiones y trabajos futuros propuestos.

2. Antecedentes

Desde la versión 1.2 de la plataforma estándar de Java (J2SE) está disponible la clase URLClassLoader que permite cargar clases Java desde fuentes remotas. Ya en la versión 1.1 de Java, el Class Loader podía obtener clases desde cualquier origen. No obstante, con esta implementación existían problemas de performance, seguridad y permisos de acceso [15].

Aprovechando la característica de carga desde fuentes remotas, Parker y Cleary [15] evaluaron la posibilidad de emplear el protocolo P2P para la carga remota de clases entre JVM. Dos implementaciones de referencia confirman que este paradigma es un enfoque válido para aplicar a sistemas distribuidos. En el trabajo de Ryan y Newmarch [16] estudian distintas técnicas de class loading distribuido y proponen una estructura de descubrimiento y publicación remoto de clases. Las pruebas posicionan a esta técnica como una alternativa viable incluso para entornos wireless.

En el mismo sentido, para ambientes de ejecución de recursos limitados, el trabajo de Petrea y Grigoras [17] presenta una implementación de máquina virtual con

capacidad de carga remota de clases. Mediciones de desempeño verifican que el impacto en la performance es mínimo al comparar el tiempo de respuesta con una máquina virtual CLDC (Connected Limited Device Configuration).

Ossher y otros [1] proponen a Sourcerer para resolver automáticamente las dependencias del software open source a través de un algoritmo de referencia cruzada. Los autores llegan a la conclusión que la resolución automática de dependencias es una opción viable, pero es necesario mejorar la indexación de los meta-datos disponibles en los repositorios.

Por otro lado, el trabajo de Frénot y otros [18] presenta un framework orientado a dispositivos de recursos restringidos, donde se ejecuta OSGi. La solución se basa en servidores de caché remotos que entregan el bytecode de una clase a demanda. En el ambiente local sólo se instalan representaciones de los componentes. Los resultados de las pruebas confirman que, en especial para bundles[3] de mayor tamaño, el tiempo de instalación se reduce considerablemente.

García y otros [2] al estudiar la fragmentación que existe entre los repositorios de open source, proponen un metamodelo para describir los componentes OSGi y un sistema de federación de repositorios. Este trabajo apunta a ofrecer resolución de dependencias por facetas. Wang y otros [19] identifican dos casos puntuales de malas dependencias: las subutilizadas y las inconsistentes. Mediante software ad hoc, los autores detectaron utilización por debajo del 20% de dependencias de terceras partes y dependencias externas en módulos base en un proyecto open source (C++).

El trabajo de Jezek y otros [13] realiza un detallado análisis de los problemas que se dan a consecuencia del alto nivel de reutilización que existe en el software open source. Se identifican los casos denominados como: mediación, reempaquetado, compatibilidad y redundancia de dependencias. Propone un análisis estático de dependencias para verificar todas las interfaces de los componentes a través de un algoritmo de prueba de compatibilidad de tipo. Un estudio empírico dimensiona el problema y advierte sobre la utilización de Maven cuando se renombran las librerías o existen clases duplicadas.

En base a los aportes de los trabajos citados y cuestiones vinculadas para las cuales está pendiente encontrar una mejor solución, a continuación se desarrollará un estudio que intentará establecer un grado de utilización medio de las dependencias en el software open source.

3. Análisis de dependencias

El objetivo de esta tarea consistió en determinar el nivel de acoplamiento entre las librerías JAR. Para ello, en una primera instancia se evaluó utilizar una herramienta de terceros. Se probó con Google Codepro Analytix[4] y también con JDepend[5]. Si bien ambas están muy difundidas tanto en la industria como la academia e implementan las métricas propuestas por Robert Martin [20], se llegó a la conclusión que las siguientes características no eran apropiadas para este estudio:

[3] Conjunto de clases Java empaquetadas como archivo JAR y que posee un archivo manifiesto donde se detallan características como nombre, versión, dependencias, interfaces y otros.

[4] https://developers.google.com/java-dev-tools/codepro/

[5] http://clarkware.com/software/JDepend.html

JDepend:

- El análisis que realiza no es relativo a una dependencia en particular, sino que evalúa a todas las dependencias del proyecto.
- Los resultados son por paquete, no hay un cálculo general.

Google Codepro Analytix:

- La cantidad de referencias a una dependencia no son únicas, es decir, si una clase tiene 2 referencias a una misma clase externa (librería), son contadas como 2 referencias.
- En el informe detallado, los resultados son globales, en relación a todas las dependencias y no una en particular.

Por estos motivos se decidió desarrollar una herramienta específica. El proyecto está publicado como open source, se denomina Deep y ejecuta un análisis similar al de Wang y otros [19] y una métrica basada en el trabajo de Martin [20]. En una misma sesión analiza dos archivos JAR y establece una tasa de dependencia entre sí.

Internamente ejecuta las siguientes tareas:

1. Identifica las clases públicas (incluyendo las abstractas y las interfaces), miembros (variables y métodos) del JAR objetivo (la librería).
2. Busca referencias a esas clases/miembros en el JAR origen y muestra por consola un resultado preliminar.
3. Genera una visualización jerárquica de las dependencias a través de un árbol de dependencias.
4. Por último, calcula la Tasa de Dependencia y muestra los resultados.

3.1. Tasa de dependencia

A fin de cuantificar el grado de dependencia de un JAR hacia otro, se definió una métrica cuyo resultado se obtiene de los siguientes resultados parciales. Siendo:

S: el JAR origen
T: el JAR objetivo
Rc: Clases concretas referenciadas en S
Tc: Total de clases concretas disponibles en T
Ra: Clases abstractas referenciadas en S
Ta: Total de clases abstractas disponibles en T
Ri: Interfaces referenciadas en S
Ti: Total de interfaces disponibles en T
Rm: Miembros referenciados en S
Tm: Total de miembros disponibles en T

$$\text{Tasa de dependencia} = \frac{\frac{Rc}{Tc} + \frac{Ra}{Ta} + \frac{Ri}{Ti} + \frac{Rm}{Tm}}{4} \qquad (1)$$

En resumen, es un promedio de las proporciones entre los recursos referenciados y los disponibles. En el repositorio del proyecto[6] se puede bajar la última versión compilada, se explica el modo de uso y se muestran pantallas con salidas por consola.

4. Mediciones

Empleando la herramienta presentada en la sección anterior, se seleccionó un conjunto de productos de software de importante magnitud, con características heterogéneas entre sí[7] y muy difundidos en la comunidad open source Java. También se sumó al análisis el mismo proyecto Deep (sin incluir las dependencias en el mismo JAR). En la Tabla 1 puede observarse el resultado de la medición individual de cada JAR de origen (S) con cuatro dependencias (T) elegidas aleatoriamente.

Tabla 1 – Medición de tasa de dependencia

JAR Origen (S)	JAR Objetivo (T)	Tasa de dependencia	Promedio
spring-2.0.7.jar	log4j-1.2.14.jar	0,0122	0,03172
	standard-1.1.2.jar	0,0375	
	cglib-2.1.jar	0,0714	
	jstl-1.5.0.jar	0,0375	
drools-core-6.2.0.jar	protobuf-2.5.0.jar	**0,3292**	0,12760
	xstream-1.4.7.jar	0,0688	
	slf4j-api-1.7.2.jar	0,1005	
	comm-codec1.4.jar	0,0119	
hibernate-core4.3.jar	javassist-3.18.jar	0,1196	0,18902
	dom4j-1.6.1.jar	0,1603	
	antlr-2.7.7.jar	0,1548	
	jpa-2.1-api.jar	**0,3214**	
symmetric-3.7.19.jar	comm-codec1.3.jar	0,0177	0,051175
	comm-coll-3.2.jar	0,0015	
	comm-io-2.4.jar	0,1671	
	log4j-1.2.17.jar	0,0184	
dbvisualizer-9.2.8.jar	synthetica.jar	0,0013	0,056275
	jdom-2.0.jar	0,1610	
	icepdf-core-4.3.jar	0,0240	
	dom4j-1.6.jar	0,0388	
drools-com-6.2.0.jar	antlr-runt-3.5.jar	**0,3160**	0,096975
	xstream-1.4.7.jar	0,0336	
	slf4j-api-1.7.2.jar	0,0842	
	mvel2-2.2.4.jar	0,0719	
deep-nodep-1.01.jar	antl-rt-4.5.1.jar	**0,2243**	0,067775
	bcel-5.2.jar	0,0222	
	procyon-dec0.5.jar	0,0111	
	ini4j-0.5.4.jar	0,0135	

[6] https://github.com/martinaguero/deep

[7] Se los considera heterogéneos dado que fueron desarrollados por diferentes empresas / comunidades y poseen objetivos diversos.

Si bien se observan casos particulares como Protocol Buffers (protobuf) para Drools y la API de JPA para Hibernate donde se da una tasa de dependencia significativamente por encima de los demás, el promedio general (0,08864) no llega a alcanzar la décima parte. A priori, se puede afirmar que en esta muestra, en promedio, se está utilizando menos del 10% de las capacidades disponibles en las librerías.

Otro caso que también se midió es la librería ANTLR Runtime 3.5 para Drools Compiler 6.2 donde el análisis entrega un resultado de 0.316 lo cual es esperable, dado que el Parsing es una función central del módulo compilador de Drools.

5. Propuesta

Como respuesta a la situación planteada y con intención de la simplificar y optimizar la gestión de dependencias, se propone el modelo conceptual representado en la Figura 2.

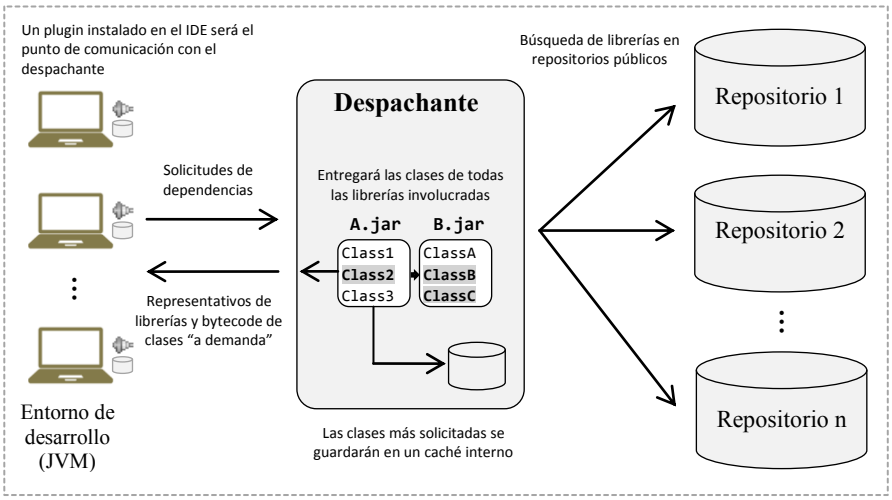


Fig. 2. Intermediario entre entornos de desarrollo (IDE) y repositorios de librerías.

En la Figura 2 las entidades vinculadas por el despachante son: el compilador, la máquina virtual de Java (entorno de desarrollo) y los repositorios de bytecode empaquetado en archivos JAR (librerías).

5.1. Tiempo de compilación

En esta instancia, el intermediario entregará representativos de los JAR requeridos por el cliente, de manera similar a la solución propuesta por Frénot y otros para el framework ROCS [18]. Un representativo es una "sombra" o "cáscara" de las librerías que contiene únicamente los descriptores y firmas de las clases. Todas las dependencias transitivas también serán resueltas automáticamente por el despachante y se enviarán al cliente en forma de representativos.

5.2. Tiempo de ejecución

Durante la ejecución del programa, la JVM obtendrá el bytecode mediante la carga remota de las clases. El despachante será el encargado de entregar al Class Loader el bytecode requerido. Un caché interno guardará en el cliente una copia del bytecode solicitado para que en futuras ejecuciones la carga se realice desde una fuente local.

5.3. Despachante

El despachante se encargará de resolver la clausura de dependencias en función de los datos provistos por el class-path definido en el IDE del cliente. En tiempo de compilación, entregará los representativos de las dependencias directas e indirectas. La resolución de todas las dependencias estará a su cargo y lo realizará de forma automática en base a los datos disponibles en los repositorios respecto a las dependencias de cada librería. La comunicación con el cliente será a través de un plugin específico y estará instalado en el IDE. En tiempo de ejecución responderá a las solicitudes de clases y enviará por red únicamente aquellas involucradas en la ejecución. Un caché interno en el despachante y el plugin permitirá optimizar el tiempo de respuesta para las solicitudes de carga más frecuentes. El contacto del cliente con el despachante será siempre a través de un único punto de entrada y una dirección permanente, más allá de la naturaleza del proyecto en desarrollo y los requisitos de dependencias, será responsabilidad del plugin establecer comunicación con el despachante y lo mismo para el Class Loader.

5.4. Dependencias transitivas

Como se explicó en las secciones anteriores, será responsabilidad del despachante resolver las dependencias de primer nivel y subsiguientes. En la Figura 3 se presenta un caso de resolución encadenada de dependencias.

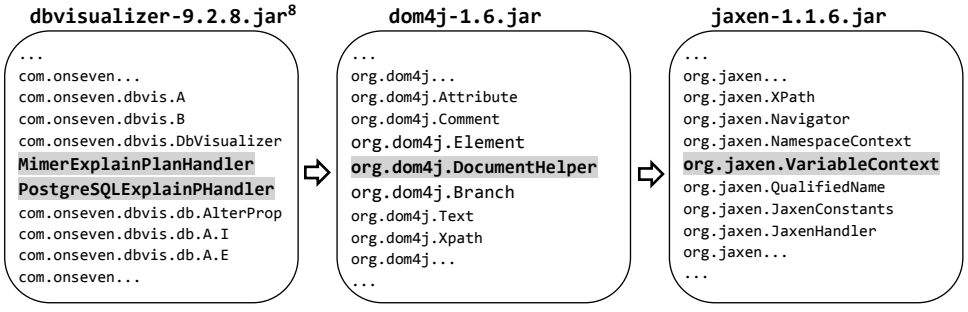


Fig. 3. Resolución de dependencia transitiva.

Como se ve en la Tabla 1, DBVisualizer 9.2 sólo requiere del 0,04 de los recursos públicos de Dom4J que a su vez éste sólo requiere el 0.10 de los recursos públicos de

[8] Los nombres completos de estas clases son `com.onseven.dbvis.db.mimer.MimerExplainPlanHandler` y `com.onseven.dbvis.db.postgresql.PostgreSQLExplainPlanHandler`

Jaxen[9]. El software visualizador de bases de datos referencia a sólo 5 clases de Dom4J de las cuales 3 son interfaces y 2 son concretas, entre ellas está `org.dom4j.DocumentHelper`. A su vez Dom4J hace referencia a 17 clases de Jaxen de las cuales 6 son interfaces, 1 es abstracta y 10 son concretas, una de ellas es `org.jaxen.VariableContext`.

El despachante entregará al cliente los representativos de cada librería para conseguir la compilación y en tiempo de ejecución, sólo las clases requeridas, no el JAR completo, como hubiese ocurrido si no se utilizara la propuesta de este trabajo.

6. Concusiones y trabajos futuros

Inicialmente este trabajo se enfocó en conocer las características de las librerías de programas compilados de dominio público. Se estudiaron las características técnicas y el modo en que son puestas a disposición de la comunidad de desarrolladores open source. También se presentaron las herramientas de software habitualmente empleadas para gestionar las dependencias, haciendo foco en las características que hoy, por diversos motivos, son obsoletas. Asimismo también se hizo un breve repaso por trabajos que han abordado el tema y las alternativas que proponen la academia y la industria.

A fin de obtener datos cuantitativos de la relación entre el software compilado Java, se desarrolló un estudio para establecer una tasa de dependencia habitual entre el software open source y sus dependencias. En una primera instancia se evaluó realizar el análisis utilizando software de terceros pero finalmente se decidió desarrollar una herramienta ad hoc. Los resultados de las mediciones arrojaron una proporción inferior al 0.10 en una muestra de 7 proyectos, llegando a la primera conclusión empírica de este trabajo: se utilizan menos del 10% de los recursos disponibles (clases y miembros públicos) en los archivos JAR.

En base a la situación presentada, se propone reemplazar el vuelco total de dependencias por un servicio intermediario ubicado entre los repositorios de binario open source y los entornos de desarrollo. Se plantea establecer un middleware despachante de representativos de librerías que también atienda solicitudes puntuales de bytecode de clases Java en tiempo de ejecución.

Por último, se presentó un caso de resolución completa de dependencias de primer y segundo nivel, explicando cómo la propuesta obtendría y enviaría al cliente únicamente las clases referenciadas.

La próxima etapa del proyecto tiene planeado desarrollar la especificación completa y detallada de la solución y una implementación de referencia. El objetivo será desarrollar y publicar un servicio en la nube que, en conjunto con un plugin para Eclipse, permita resolver dependencias, compilar y ejecutar software Java. También se planea medir la diferencia de tiempo entre ejecución con dependencias locales y remotas con despachante.

[9] 0,0958 medido con Deep 1.01

7. Referencias

1. Ossher, J., Bajracharya, S., Lopes, C.: Automated Dependency Resolution for Open Source Software. Working Conference on Mining Software Repositories. IEEE (2010)
2. García-Carmona, R., Cuadrado, F., Dueñas, J., Navas, A.: A repository for integration of software artifacts with dependency resolution and federation support. Software and Data Technologies. Springer (2013)
3. OSGi Core Release 6, The OSGi Alliance (2014)
4. Zhou, J., Zhao, D., Ji, Y., Liu, J.: Examining OSGi from an Ideal Enterprise Software Component Model. Software Engineering and Service Sciences. IEEE (2010)
5. InfoQ, http://www.infoq.com/news/2015/03/maven-polyglot
6. McIntosh, S., Adams, B., Hassan, A.: The evolution of Java build systems. Springer Science+Business Media (2011)
7. Massol, V., Van Zyl, J.: Better Builds with Maven. Mergere Library Press (2006)
8. XML, http://www.xml.com/pub/a/ws/2001/04/04/soap.html
9. Schmidt, E., Rosenberg, J.: How Google Works. Grand Central Publishing (2014)
10. The Java Virtual Machine Specification, http://docs.oracle.com/javase/specs/jvms/se8/html/ (2015)
11. Liang, S., Bracha, G.: Dynamic Class Loading in the Java Virtual Machine. ACM SIGPLAN Conference (1998)
12. Shankar, L., Burns, S.: Demystifying class loading problems, Part 1: An introduction to class loading and debugging tools. DeveloperWorks. IBM (2005)
13. Jezek, K., Dietrich, J.: On the Use of Static Analysis to Safeward Recursive Dependency Resolution. 40th Euromicro Conference on Software Engineering and Advanced Applications (2014)
14. Maven Transitive Dependencies, https://maven.apache.org/guides/introduction/introduction-to-dependency-mechanism.html
15. Parker, D., Cleary, D.: A P2P Approach to ClassLoading in Java. Agents an Peer-to-Peer Computing. Lecture Notes in Computer Science. Springer-Verlag (2003)
16. Ryan, A., Newmarch, J.: A Dynamic, Discovery Based, Remote Class Loading Structure. Software Engineering and Applications (2003)
17. Petrea, L., Grigoras, D.: Remote Class Loading for Mobile Devices. International Symposium on Parallel and Distributed Computing. IEEE (2007)
18. Frénot, S., Ibrahim, N., Le Mouël, F., Ben Hamida, A.: ROCS: a remotely provisioned OSGi framework for ambient systems. Network Operations and Management Symposium. IEEE (2010)
19. Wang, P., Yang, J., Tan, L., Kroeger, R., Morgenthaler, J.: Generating Precise Dependencies for Large Software. 35th. International Conference on Software Engineering. IEEE (2013)
20. Martin, R.: OO Design Quality Metrics An Analisis of Dependencies. Object Mentor (1994)

Características del desarrollo de software en la ciudad de Corrientes

Gladys Dapozo; Cristina Greiner, Emanuel Irrazábal, Yanina Medina, María de los Ángeles Ferraro, Berenice Lencina
Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura
Universidad Nacional del Nordeste, Av.Libertad 5450, 3400, Corrientes, Corrientes, Argentina
{gndapozo,cgreiner, yanina}@exa.unne.edu.ar, emanuelirrazabal@gmail.com;
mafferraro@hotmail.com

Resumen. En este trabajo se presentan los resultados de una encuesta orientada a determinar el estado de situación del desarrollo de software en la ciudad de Corrientes, Argentina, considerando las empresas específicas de software y las áreas de sistemas de las principales organizaciones, de dependencia estatal o privada, que desarrollan software para contribuir al logro de los objetivos de las mismas. En general, se advierte que las áreas/empresas no aprovechan los métodos y herramientas más actuales para lograr gestionar los proyectos de desarrollo con mayor eficiencia y lograr productos de mayor calidad. Estos resultados permitirán direccionar actividades de divulgación o capacitación sobre estos temas, a través de las universidades o polos tecnológicos de la región.

Palabras clave: Calidad de software. Métodos y herramientas. Gestión de proyectos.

1. Introducción

A partir del año 2002, en la Argentina, las políticas públicas se orientaron a la promoción de la industria del software a lo largo y ancho del territorio.

En particular, en la provincia de Corrientes, en el año 2007 se lanza el Plan Estratégico para la Industria del Software y Servicios Informáticos (PEISSI), cuyo objetivo es fortalecer a este sector de la economía por considerarlo clave para el desarrollo tecnológico del sistema productivo en su conjunto. Se apunta específicamente a proveer asistencia a las empresas del rubro de la provincia a través de las herramientas que dispone el estado para potenciarlas. Con este impulso, nace el Polo IT Corrientes. Las empresas nucleadas en el Polo IT Corrientes, mediante la asociatividad, buscan lograr el crecimiento individual y conjunto, la transferencia de conocimiento, investigación y desarrollo, la búsqueda de alianzas estratégicas con organismos y universidades y la mejora continua. Entre sus logros sobresalientes se encuentra la adhesión a la CESSI, la certificación de calidad bajo la norma ISO 9001:2000 de las empresas integrantes del Polo, mediante el financiamiento obtenido a través del estado provincial [1].

Un primer estudio, realizado en el año 2009, caracterizó las pymes de software de la región NEA con el propósito de definir un marco de referencia para el desarrollo de acciones enfocadas en la mejora de la calidad del proceso y/o de la calidad de los productos de software que las empresas producen. El relevamiento realizado en las provincias de Chaco, Corrientes y Misiones, reflejó que el 96% de las empresas tenían

menos de 10 empleados. De la totalidad considerada ninguna había obtenido una certificación de calidad aunque manifestaban la importancia de contar con ella [2].

Dado el avance de la Ingeniería de Software, en general, así como los métodos y herramientas orientados a mejorar la calidad del software, resulta interesante revisar y analizar cómo trabajan las empresas de software en la ciudad de Corrientes, incluyendo en esta observación a los departamentos o áreas de sistemas cuya función es desarrollar el software que se utiliza en las principales organizaciones gubernamentales y empresas de servicios, que en la zona son las principales demandantes de los recursos humanos formados en las carreras de Informática.

Para este estudio se definieron las siguientes áreas de interés: Conformación de los equipos de trabajo en cuanto a cantidad, roles que desempeñan y formación académica. En lo que respecta a la Ingeniería de software y los aspectos vinculados con la calidad, se decidió observar lo referido a las metodologías de desarrollo, métodos de estimación, gestión cuantitativa, medición de software y testing.

A continuación se brindan los principales conceptos que enmarcan los distintos aspectos analizados.

1.1. Metodologías de desarrollo

A lo largo del tiempo en la ingeniería del software se han sucedido una gran cantidad de metodologías y técnicas de desarrollo software [3]. Las metodologías han pasado de estar directamente relacionadas con la predictibilidad y rigidez, a hacerse cada vez menos prescriptivas. Actualmente las metodologías o técnicas ágiles, como SCRUM, XP, KANBAN, son ampliamente utilizadas en las empresas de desarrollo software y las áreas de sistema. Esto se relaciona con la posibilidad de gestionar los cambios a lo largo del proyecto, con mantenimientos perfectivos en lugar de realizar aplicaciones "desechables" [4]. En este sentido diferentes estudios coinciden en que el uso de las metodologías ágiles está en torno a un 50%, tanto en el hemisferio norte [5] como en Argentina [6].

Por otra parte, el conocimiento de las diferentes metodologías y herramientas no asegura su utilización. Son bien conocidos los estudios donde se relaciona lo que el profesional ha conocido en su vida académica y profesional, con lo que utiliza en el entorno laboral [7], por tanto es necesario preguntar por ambos aspectos: lo que conoce y lo que utiliza.

1.2. Estimación

Según [8] "una estimación es una predicción de cuánto tiempo durará o costará un proyecto", constituye la base para la planificación de los proyectos. El desarrollo del software requiere de la estimación para controlar y administrar los recursos que se necesitan utilizar antes y durante el proyecto. Son numerosas las variables, relacionadas con los recursos humanos, el contexto y las políticas que intervienen en el proceso de desarrollo, que pueden afectar los resultados finales.

Para las empresas de software una estimación deficiente del esfuerzo y duración que conlleva un proyecto puede ocasionar incumplimiento de plazos, entrega de productos incompletos y pérdida de competitividad. En contextos de desarrollo de proyectos de pequeña o mediana envergadura resulta que a mayor especificidad en cuanto al tipo de proyecto y una mayor disponibilidad y pertinencia de datos

históricos, los valores de esfuerzo y duración estimados se aproximan más a los valores reales [9].

1.3. Gestión cuantitativa

La gestión cuantitativa de procesos [10] proporciona una visión del grado de cumplimiento de metas así como de las causas que explican desviaciones significativas en procesos o productos. El propósito de esta gestión es dirigir un proyecto u organización basado en un conocimiento cuantitativo, es decir medible, determinable, de los aspectos de mayor relevancia, que generalmente son procesos cuyo rendimiento afecta en forma significativa al logro de los objetivos del proyecto y la satisfacción de los clientes [11].

La mejora de procesos basada en medición promueve la gestión cuantitativa de proyectos de software, mediante el seguimiento continuo de procesos y productos, con el fin de predecir su comportamiento y detectar desviaciones durante su ejecución. Las mediciones, cuando son analizadas, constituyen una base importante para una gestión efectiva por parte del equipo de desarrollo [11], [12].

1.4. Medición de atributos de calidad del software

La medición de los atributos de calidad del software es una necesidad para las empresas de software y servicios informáticos (SSI), dado que permite mejorar las tareas menos eficientes y obtener productos de calidad. Esto se traduce en una ventaja competitiva para las organizaciones, ya que el conocimiento adquirido a través de la medición permite modificar aquellos factores que aportan una mayor eficacia en el proceso productivo y contribuyen a la calidad del producto [13].

La calidad del software es una compleja combinación de factores. Diversos autores [14] [15], y estándares tales como el ISO 9126 [16] han tratado de determinar y categorizar los factores que afectan a la calidad del software. Para garantizar la calidad del software, se requiere medir los atributos que la definen. El análisis de estas mediciones durante el proceso de desarrollo proporciona al ingeniero de software una aproximación en tiempo real de la eficacia de los modelos de análisis, diseño y código, y de la calidad general del software que se construye.

La medición del producto se realiza mediante las métricas, para caracterizar numéricamente los distintos aspectos del desarrollo del software. Michael Mah [17] define las métricas de software como “La aplicación continua de mediciones basadas en técnicas para el proceso de desarrollo del software y sus productos para suministrar información relevante a tiempo, así el administrador junto con el empleo de estas técnicas mejorará el proceso y sus productos”. Los datos recogidos durante el proceso de medición se deben conservar en registros históricos, y mantener como un recurso organizacional [18].

1.5. Testing

Los fallos en el software, tanto de funcionalidad como de seguridad, continúan presentándose, y se considera que en parte el problema se debe a que la complejidad del software se incrementa en forma permanente y no se consigue controlarla, a pesar de las nuevas técnicas y metodologías que van surgiendo. La complejidad del software juega un importante papel en la presencia de errores: a mayor complejidad, mayor número de errores. Los desarrolladores dedican un 80 por ciento de los costos

de desarrollo a la identificación y corrección de errores. La planificación y diseño de pruebas en las primeras fases de desarrollo permiten encontrar errores, omisiones, inconsistencias e, incluso, especificaciones redundantes en los requisitos funcionales cuando aún es fácil y económico corregirlas ya que el costo de eliminar defectos aumenta a medida que aumenta el tiempo que transcurre entre la aparición del defecto y su detección [19]. Para que el proceso de prueba del sistema sea eficaz debe estar integrado dentro del propio proceso de desarrollo y, como cualquier otra fase del desarrollo, debe realizarse de manera sistemática [20].

2. Metodología

Para relevar la situación respecto de la actividad de desarrollo de software de las empresas de software y áreas de Sistemas de las organizaciones y empresas de la ciudad de Corrientes, se diseñó un cuestionario para recabar información, enfocado en los puntos de interés de la investigación: Características generales, del desarrollo, estimación, gestión cuantitativa, medición de producto software y testing.

El cuestionario se envió por correo electrónico a los gerentes y responsables de desarrollo de software de las distintas organizaciones preseleccionadas. Una vez recibidas las encuestas, se realizaron entrevistas (presenciales o telefónicas) a fin de verificar las respuestas, aclarar o ampliar los puntos tratados.

A continuación, los datos fueron procesados mediante una planilla de cálculo.

3. Resultados

A continuación se describen los resultados de la encuesta en cuanto a las distintas variables seleccionadas para su análisis:

Características generales:

Se relevaron datos de 21 empresas/áreas de sistemas que desarrollan software en la ciudad de Corrientes, de las cuales 14 (67%) corresponden a áreas de sistemas y 7 (33%) son empresas de desarrollo de software, exclusivamente. Las áreas de sistemas son en su mayoría (79%) de gestión estatal nacional, provincial o municipal. Entre estas se cuentan: Banco de Corrientes, Dirección General de Rentas, Instituto de Viviendas, Dirección Provincial de Energía, Poder Judicial de Corrientes, Ministerio de Salud, Municipalidad de Corrientes, Rectorado UNNE, Obra Social UNNE, Centro de Cómputos de Corrientes y Hospital Escuela de Corrientes.

Son 186 personas que desarrollan software o realizan tareas vinculadas. El 76% corresponde a personas que se desempeñan en áreas de sistemas (ver Tabla 1).

Tabla 1: Cantidad de personas

Áreas/Empresas	Privada	Pública	Emp Soft	Total	%
Área Sistemas	28	113		141	76%
Empresa software			45	45	24%
				186	100%

En cuanto a los roles declarados en la actividad, en la Figura 1 se puede observar una preponderancia del rol de Analista-Programador (37%), seguida por la de

Programador (14%) y Tester (13%). Se destaca que los equipos comienzan a incorporar el rol de Diseñador Gráfico. Cabe aclarar que en la mayoría de los casos los roles no son exclusivos, es decir, habrá más roles que personas.

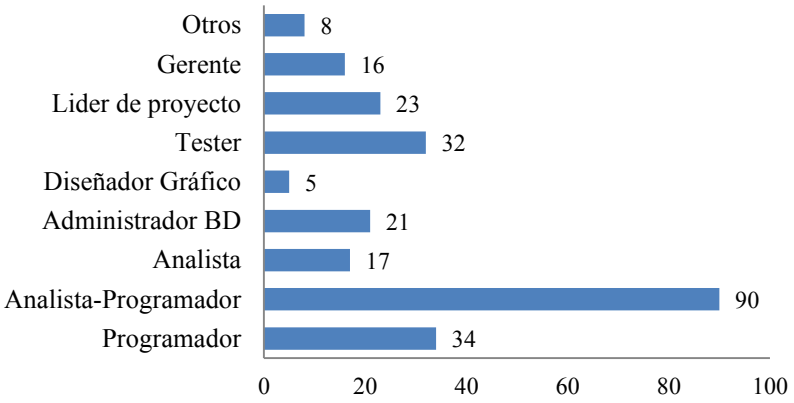


Fig.1: Roles desempeñados en los equipos de trabajo

En cuanto a la formación académica puede observarse en la Tabla 2 que mayoritariamente (63%) los trabajadores poseen título de grado. Se observa también que los estudiantes activos (se encuentran cursando una carrera de informática) constituyen un 18%. Los estudiantes inactivos son aquellos que en la actualidad no se encuentran cursando regularmente una carrera, constituyen el 7%. El más bajo porcentaje corresponde a los Idóneos (5%), justificado en la región porque existen variadas ofertas de carreras de Informática en universidades públicas y privadas.
Respecto de la cuestión de género, se puede observar una fuerte presencia masculina en los equipos de trabajo (74%).

Tabla 2: Formación académica

Formación	Varones	Mujeres	Total	%
Título de grado	73	32	105	63%
Título intermedio o superior	8	4	12	7%
Estudiante activo	25	5	30	18%
Estudiante inactivo	10	2	12	7%
Idóneos	9	0	9	5%
Totales	125	43	168	100%
	74%	26%	100%	

Dentro de los títulos académicos habilitantes para el desempeño profesional se observa una mayor presencia de Licenciados en Sistemas de Información (50%), dado que esta carrera se dicta en la ciudad de Corrientes. Continúan los Ingenieros en Sistemas de Información (27%) de UTN Regional Resistencia, oferta distante a 25 km de la ciudad de Corrientes. Ver Figura 2.

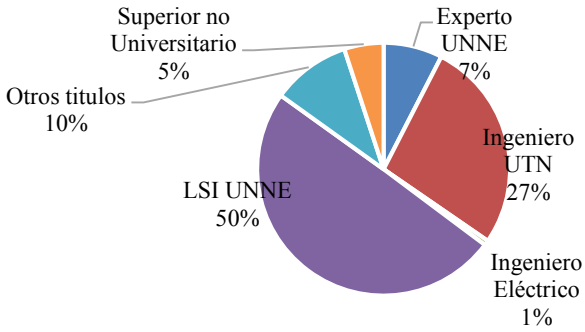


Fig.2: Titulaciones de grado y pregrado

Características de los proyectos de desarrollo software de la región.

En la Fig.3 se puede ver un resumen de los tipos de proyecto llevados a cabo en la región. Por un lado, la mayor cantidad de proyectos es "web" seguido por el tipo de "escritorio". Los proyectos móviles son los de menor porcentaje con un 26%. Todavía se percibe un sesgo en cuanto a proyectos de tipología exclusiva. Finalmente, como un dato a destacar, es que otro tipo de proyectos, por ejemplo relacionados con firmware, ocupan en promedio un 30% de los desarrollos en las empresas.

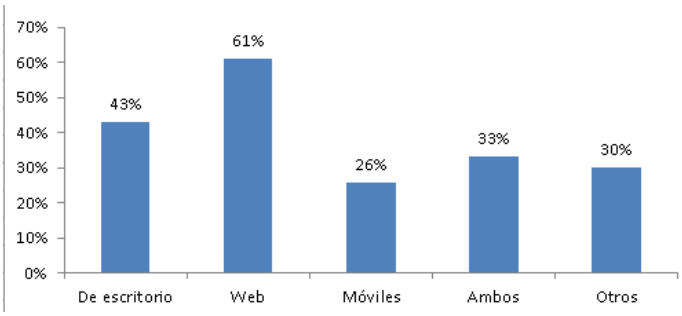
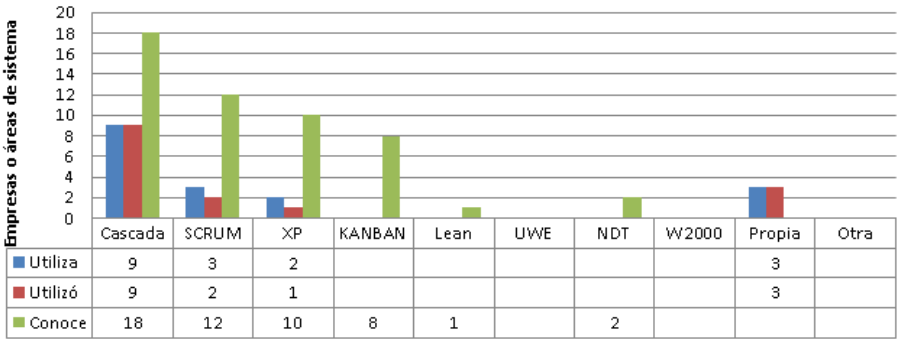


Fig.3. Tipos de desarrollo

Los proyectos que desarrollan un nuevo sistema respecto de los proyectos de mantenimiento, representan un (43%). Esto se da especialmente en la administración pública.

En la Fig. se detallan las metodologías y técnicas de desarrollo, utilizadas por las empresas y áreas de sistema encuestadas. Para cada metodología se preguntó si la conocía (barra de la derecha), si trabajó con ella en los últimos 3 años (barra central) y si trabaja actualmente (barra de la izquierda). Como puede verse en el gráfico, continúa siendo la metodología en cascada la más conocida y utilizada. Las metodologías ágiles están siendo cada vez más conocidas, pero utilizadas en un porcentaje mucho menor. Cerca de un 20% de las empresas que conocen las metodologías ágiles la utilizan. Asimismo, no se utilizan metodologías propias de proyectos web, sino solamente métodos y técnicas genéricas.



	Cascada	SCRUM	XP	KANBAN	Lean	UWE	NDT	W2000	Propia	Otra
Utiliza	9	3	2						3	
Utilizó	9	2	1						3	
Conoce	18	12	10	8	1		2			

Fig.4. Metodologías de desarrollo

3.1. Estimación

Los resultados indican que el 52% de las empresas/áreas realizan una estimación para el 100% de los proyectos desarrollados. A su vez, cuando se realiza estimación, esta tarea está a cargo de grupo de expertos (40%) o de una sola persona (40%), como se observa en la Figura 5.

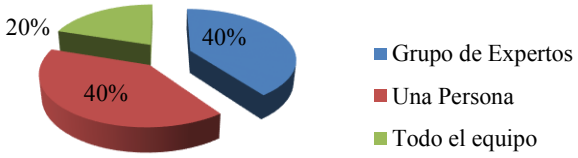


Fig. 5. Responsables de la estimación

El interés que tienen las empresas en la estimación recae, principalmente, en la Duración y Costo, y en el Tamaño-Esfuerzo y Duración, como se advierte en la Figura 6.

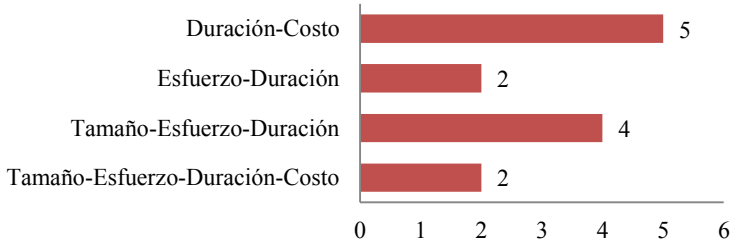
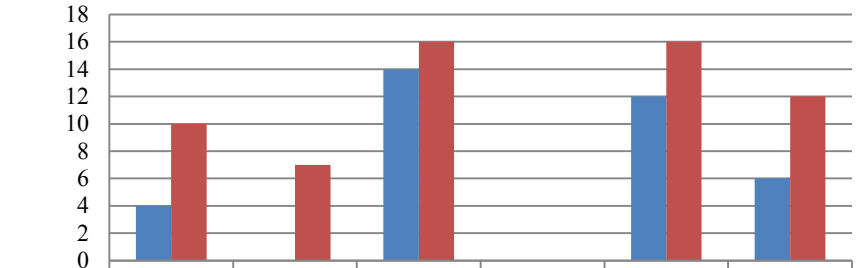


Fig. 6. Atributos que estiman las empresas de los proyectos

Respecto de las técnicas utilizadas para la estimación se puede observar, en la Figura 7, que la mayoría utiliza las técnicas convencionales de juicio de experto, por analogía y por consenso. Se observa que las técnicas paramétricas, como Puntos de Función o COCOMO, son conocidas pero poco utilizadas.



	COCOMO	Puntos de función	Juicio experto	Otras	Por analogía	Por consenso
Utiliza	4		14		12	6
Conoce	10	7	16		16	12

Fig, 7. Técnicas de estimación

La técnica que se utiliza depende de los datos disponibles y de la naturaleza del problema bajo estimación. Respecto al origen de los datos disponibles para el momento de estimar se pudo observar que la gran mayoría (61%) no utiliza datos históricos al momento de realizar la estimación de un proyecto, aspecto que se torna importante para obtener resultados más precisos.

3.2. Gestión cuantitativa

Se sabe que la medición se emplea para establecer una línea base del proceso, a partir de la cuáles evalúan las mejoras. Esta línea base se constituye con datos recopilados en proyectos previos que contienen medidas de proyectos y métricas derivadas de estos. De las empresas encuestadas, un 48 % guardan datos históricos de los proyectos, tales como fecha de inicio, fin, recursos, costos. Sin embargo esta información, como se ha visto en el apartado anterior, no se utiliza para la estimación.

De las herramientas que se utilizan para la gestión de los proyectos puede observarse en la Figura 8 que las más usadas son Project, Excel y Jira, en versión propietaria. En tanto que en versiones libres, se indicó el uso de RedMine y GitHub. Se indicaron otras soluciones, por ejemplo: Planillas de Google Drive, SharePoint, Rational Clear Quest, Team Fundation Server (TFS) y herramientas propias.

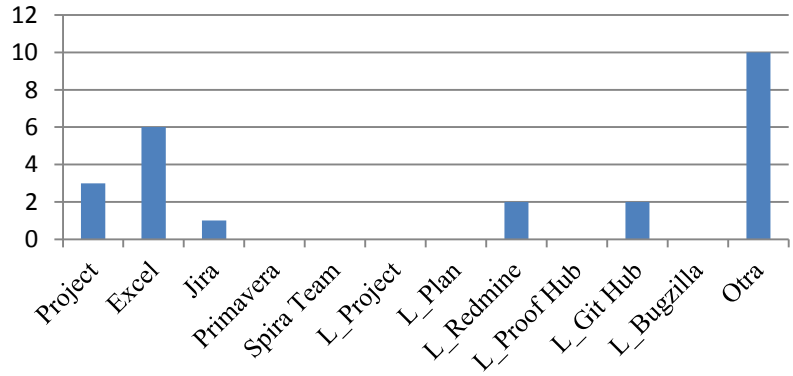


Fig. 8. Herramientas de gestión cuantitativa

Entre los aspectos en los que contribuye la herramienta utilizadas, se señalaron (ver Figura 9): Planificación, Colaboración (comunicación con el equipo) y Estimación. En menor medida contribuyen a la gestión de incidencias, versionado y configuración.

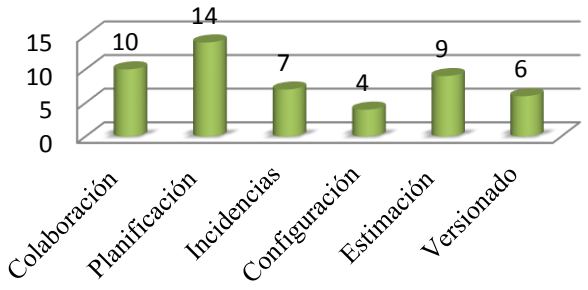


Fig. 9. Aspectos en los que contribuyen las herramientas de gestión

3.3. Medición de atributos de calidad del software

Del total de empresas y áreas de sistemas encuestadas, ninguna de ellas realiza mediciones de atributos de calidad del software.

Entre las razones expuestas por los entrevistados para esta situación, como puede observarse en la Figura 10, destacan el desconocimiento sobre metodologías, herramientas y métricas (38%), seguido por la falta de tiempo y RRHH (28%), y el considerar muy costosa esta actividad (19%). En menor medida, señalaron que la información referida a mediciones no es usada, pudiendo considerarse que a nivel gerencial se le resta valor, y en otro caso señalaron que no resulta útil.

Fig. 10. Razones para no medir atributos de calidad del software

3.4. Testing

Tanto en las empresas de desarrollo como en las áreas de sistemas los recursos humanos en general no tienen un rol exclusivo, sino que desarrollan distintas tareas. Esta característica se repite en lo relacionado a la tarea de testing o prueba de software, ya que sólo en el 10% de los casos cuentan con un equipo exclusivo para testing, siendo 1 caso para empresa y 1 caso para área de sistemas.

De manera similar, solo el 10% de empresas realiza testing en forma automática, siendo en este caso en 2 empresas dedicadas al desarrollo de software.

En cuanto al tipo de pruebas que se realizan, tal como se observa en la Figura 11, la mayoría de los encuestados (58%) señala que realizan prueba de aceptación, siendo la prueba de integración menos utilizada.

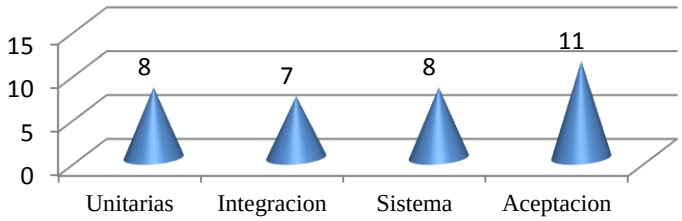


Fig. 11: Tipos de pruebas

Finalmente, sólo el 19% manifiesta utilizar herramientas para realizar test, siendo las nombradas Selenium, JUnit, utPLSQL, Rspec y Capybara.

4. Conclusiones

De los resultados obtenidos, en cuanto a los tipos de proyectos se nota un fuerte avance de los proyectos web pero falta una mayor apuesta por los desarrollos móviles.

Las metodologías ágiles están siendo cada vez más conocidas, pero utilizadas en un porcentaje mucho menor. Sólo un 20% de las empresas que conocen las metodologías ágiles la utilizan.

En la mayoría de las áreas/empresas, las prácticas de estimación más utilizadas son la de juicio de experto y por analogía. Se conocen pero no se utilizan, o se utilizan poco, las técnicas paramétricas que pueden aportar mayor precisión en la estimación. Así también es bajo el porcentaje de las áreas/empresas que utilizan datos históricos para la estimación.

Se destaca también que las organizaciones encuestadas no miden atributos de calidad, y esto se debe en gran medida a que no conocen los métodos y herramientas específicas, y también manifiestan que no pueden dedicarle tiempo ni recursos humanos a esta práctica, desconociendo los beneficios que estas prácticas pueden aportar al desarrollo.

Respecto del testing no existen en general recursos humanos dedicados en forma exclusiva, y en un alto porcentaje se realiza en forma manual, siendo el test de aceptación el más realizado.

Se concluye que las áreas o empresas, en gran medida, desconocen métodos y herramientas que pueden contribuir a mejorar la calidad y la gestión del software que desarrollan, siendo necesario realizar desde las universidades o los polos tecnológicos actividades de divulgación o capacitación sobre estos temas.

5. Referencias

[1] Cuenca Pletsch, L.;Dapozo, G.;Greiner, C.; Estayno, M. “Vinculación Universidad-Empresa orientada a la promoción de la industria del software. Una experiencia de colaboración en la región NEA”. Revista del Núcleo de Estudios e Investigaciones en Educación Superior del MERCOSUR. ISSN 2313–9080. Vol. 1 (2012). Pp. 36-41. http://nemercosur.siu.edu.ar/publicaciones/Numero_1.pdf

[2] Estayno, M.; Dapozo G., Cuenca Plestch, L.; Greiner C., Pelozo S., “Caracterización de las pymes de software de la región NEA orientada hacia un marco de mejora de la calidad”. Anales del XV Congreso Argentino de Ciencias de la Computación. (CACIC2009). Facultad de Ingeniería. Universidad Nacional de Jujuy. 5 al 9 octubre de 2009. ISBN 978-897-24068-4-1.

[3] Bohem, Barry. A view of 20th and 21st century software engineering. EnProceedings of the 28th internationalconferenceon Software engineering. ACM, 2006. p. 12-29.

[4] Fowler, Martin; Highsmith, Jim. The agile manifesto. Software Development, 2001, vol. 9, no 8, p. 28-35.

[5] VersionOne Inc., State of Agile 9th Survey, USA, 2014.

[6] Alende, Andrea N., et al. La Utilización de los Métodos Ágiles en las Empresas de Desarrollo de Software de Argentina. 2010.

[7] Lethbridge, Timothy C. What knowledge is important to a software professional? Computer, 2000, no 5, p. 44-50.

[8] McConnell, S. (2006) "Software Estimation: Demystifying the Black Art" (Developer Best Practices). Microsoft.

[9] Gladys N. Dapozo, Yanina Medina, Andrea B. Lencina, Gabriel O. Pedrozo Petrazzini. "Métodos de estimación de esfuerzo y duración en proyectos web pequeños". Revista Electrónica Argentina-Brasil Tecnologías de Información y Comunicación. Editorial Faculdade Três de Maio (SETREM) Brasil. ISSN 2422-6424. 2015 Volumen 1 Tomo 1 Nº 2

[10]Gou, L., Wang, Q., Yuan, J., Yang, Y., Li, M., & Jiang, N. (2009). Quantitative defects management in iterative development with Bi Defect. Software Process Improvement and Practice, 14(4), 227-241

[11] Baldassarre, T., Boffoli, N., Caivano, D., &Visaggio, G. (2004). Managing Software Process Improvement (SPI) through Statistical Process Control (SPC).Lecture Notes in Computer Science, 3009, 30-46

[12] Ardila, C. & Pino, F. (2013). Panorama de gestión cuantitativa de procesos de desarrollo de software en pequeñas organizaciones. Revista S&T, 11(26), 29-46.

[13] Hernández Ballesteros, J.F., Minguet Melián, J. M. "La Medida de la Calidad del Software como Necesidad y Exigencia en Modelos Internacionales (CMMI, ISO 15504, ISO 9001). www.issi.uned.es/CalidadSoftware/Noticias/PonIng2005.rtf

[14] Pressman, R. S. Ingeniería de Software. Un enfoque práctico. Editorial MCGRAW-HILL.

[15] McCall, J.A.; Richards, P.K.; Walters, G.F. - Factors in Software Quality. Vols I, II, III. NTIS AD-AO49-014, 015, 055, Nov. 1977

[16] International Organisation for Standardisation ISO 9126: Software Engineering – Product quality, Geneva, Switzerland. 2001

[17] Mah, Michael - High-definition software measurement - Software Development. Vol. 7 Nº 5, Pages: 9-14 - ISSN: 1070-8588 - Mayo 1999.

[18] Greiner, C.; Demchum, D.; Dapozo, G.; Estayno, M. "Una propuesta de solución para automatizar la medición de aplicaciones orientadas a objeto". CACIC 2010.

[19] Jalote, P., Software Project Management in Practice, Addison Wesley, 2002.

[20] Ponce, J., Escalona, M.J., Gómez, A., Luque, M., Molina, A. Definición de una política de pruebas en la gestión cultural: aplicación al desarrollo del proyecto Mosaico. REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software. 2010. Vol.6, No. 2. ISSN: 1885-4486.

XII Workshop Bases de Datos y Minería de Datos (WBDDM)

7399 | Capturing relational NEXPTIME with a Fragment of Existential Third Order Logic
José María Turull Torres(UNLaM, Massey University)

7410 | Recuperación de noticias públicas ordenadas temporalmente y por zonas utilizando funciones combinadas de Apache Solr
Luis Ignacio Aita (UNPSJB), Damián Pablo Barry (UNPSJB), Carlos Buckle (UNPSJB), Claudio Delrieux (UNS)

7436 | Una herramienta para la evaluación y comparación de metodologías de minería de datos
Juan Miguel Moine (UTN-FRRo), Ana Silvia Haedo (UBA)

7437 | Inverted Index Entry Invalidation Strategy for Real Time Search
Esteban A Rissola (UNLu), Gabriel Tolosa (UNLu)

7516 | Comportamiento de Integración de Algoritmos para Descubrimiento de Reglas de Pertenencia a Grupos
José David Panchuk (UNaM), Sebastian Martins (UNLA), Horacio Kuna (UNaM), Ramon Garcia Martinez (UNLA)

7537 | Keyword Identification in Spanish Documents using Neural Networks
Germán Aquino (UNLP), Laura Lanzarini (UNLP)

7607 | Iris Information Management in Object-Relational Databases
Carlos Alvez (UNER), Graciela Etchart (UNER), Silvia Ruiz (UNER), Ernesto Miranda (UNER), Juan Aguirre (UNER), Marcelo Benedetto (UNER), Santiago Delfín (UNER)

7640 | An experimental study for the Cross Domain Author Profiling classification
María José Garciarena Ucelay (UNSL), María Paula Villegas (UNSL), Leticia Cagnina (UNSL), Marcelo Errecalde (UNSL)

7662 | List of Clustered Permutations in Secondary Memory
Patricia Roggero (UNSL), Nora Reyes (PUC Rio), Karina Figueroa (UMSNH), Rodrigo Paredes (UTALCA)

Capturing relational NEXPTIME with a Fragment of Existential Third Order Logic

Turull-Torres José Maria[1,2]

[1] Depto. de Ingeniería e Investigaciones Tecnológicas
Universidad Nacional de La Matanza

[2] and Massey University, New Zealand
J.M.Turull@massey.ac.nz

Abstract. We prove that the existential fragment $\Sigma_1^{2,\omega}$ of the third order logic TO^{ω} captures the *relational* complexity class *non deterministic exponential time*. As a Corollary we have that relational machines can simulate third order relational machines.

1 Introduction

Relational machines (RM) were introduced in [AV,91] (there called *loosely coupled generic machines*) as abstract machines that compute queries to (finite) relational structures, or relational database instances (dbi's) as functions from such structures to relations, that are *generic* (i.e., that preserve isomorphisms), and hence are more appropriate than Turing machines (TM) for query computation. RM's are TM's endowed with a relational store that hold the input structure, as well as *work* relations, and that can be accessed through first order logic (FO) queries (sentences) and updates (formulas with free variables). As the set of those FO formulas for a given machine is fixed, an RM can only distinguish between tuples (i.e., sequences of elements in the domain of the dbi) when the differences between them can be expressed with FO formulas with k variables, where k is the maximum number of variables in any formula in the finite control of the given RM. Note that the same is true for FO queries (i.e., relational calculus), or equivalently relational algebra queries.

On the other hand, it has been proved that RM's have the same computation, or expressive power, as the (effective fragment of the) well known infinitary logic with finitely many variables $\mathcal{L}^{\omega}_{\infty\omega}$ ([AVV,95]), (in the context of Finite Model Theory, i.e., with sentences interpreted by finite relational structures or database instances - dbi's). This logic extends FO with conjunctions and disjunctions of sets of formulas of arbitrary (infinite) cardinality, while restricting the number of variables in each (infinitary) formula to be finite. This is a very important logic in *descriptive complexity theory*, in which among other properties, equivalence is characterized by pebble (Ehrenfeucht-Fraissee) games, and on ordered dbi's it can express all computable queries (see [Lib,04], among others). Hence, a nice characterization of the *discerning* power of RM's is also given by those games.

Consequently, *k-ary* RM's are incapable of computing the size of the input structure though, however, they can compute its $size_k$. A k-ary RM, for a positive integer k, is an RM in which the FO formulas in its finite control have at most k different variables, and the $size_k$ of a structure (or dbi) is the number of equivalence classes in the relation $\equiv_k$ of equality of FO^k types in the set of k-tuples of the structure, for $1 \leq k$.

Then, it was a natural consequence to define a new notion of complexity suitable for RM's. *Relational complexity* was introduced in [AV,91] as a complexity theory where the (finite relational) input structure $\mathcal{A}$ to an algorithm is measured as its $size_k$, for some $k \geq 1$, instead of the size of its encoding, as in computational complexity. Roughly, two k-tuples in $\mathcal{A}$ have the same FO^k types if they both satisfy in $\mathcal{A}$ exactly the same FO formulas with up to k variables, r of them being free, for all $0 \leq r \leq k$. That is, if the two tuples have the same properties in the structure $\mathcal{A}$, considering only the properties that can be expressed in FO^k. In that way, relational complexity classes mirroring computational complexity classes like P, NP, $PSPACE$, $EXPTIME$ and $NEXPTIME$, etc., have been defined ([AV,91],[AVV,97]), and denoted as P_r, NP_r, $PSPACE_r$, $EXPTIME_r$ and $NEXPTIME_r$, respectively (the class $NEXPTIME_r$ is actually defined later in this article).

Beyond the study of RM's as a model of computation for queries to relational databases, relational complexity turned out to be a theoretical framework in which we can characterize *exactly* the expressive power of the well known *fixed point quantifiers* (FP) of a wide range of types. Those quantifiers are typically added to first order logic, thus forming the so called *fixed point logics*, where the different types of fixed point quantifiers add to FO different kinds of iterations of first-order operators ([Lib,04], [AVV,97]).

In [AVV,97], S. Abiteboul, M. Vardi and V. Vianu introduced new fixed point quantifiers, and organized a wide range of them as either deterministic (det), non deterministic (ndet), or alternating (alt), and either inflationary (inf) or non inflationary (ninf), according to the type of iteration implied by the semantics of each such quantifier. In the same article they proved the following equivalences: det-inf-FP = P_r, ndet-inf-FP = NP_r, alt-inf-FP = det-ninf-FP = ndet-ninf-FP = $PSPACE_r$, and alt-ninf-FP = $EXPTIME_r$ (in the case of ndet FP no negation affecting an FP quantifier is allowed).

Those characterizations of *relational* complexity classes are actually very interesting and meaningful, given that it was already known that if we restrict the input to only *ordered* structures, the following equivalences with *computational* complexity classes hold: det-inf-FP = P, ndet-inf-FP = NP, det-ninf-FP = ndet-ninf-FP = alt-inf-FP = $PSPACE$, and alt-ninf-FP = $EXPTIME$ ([Lib,04], [AVV,97]).

Regarding the characterization of relational complexity classes with other logics, A. Dawar introduced in [Daw,98] the logic SO^ω, defining it as a semantic restriction of second order logic (SO) where the valuating relations for the quantified second order variables are "unions" of complete FO^k types for r-tuples for

some constants $k \geq r \geq 1$, that depend on the quantifiers[3]. That is, the relations are *closed* under the relation $\equiv_k$ of equality of FO^k types in the set of r-tuples of the structure.

In [Daw,98] it was also proved that the existential fragment of SO^ω, $\Sigma_1^{1,\omega}$, characterizes exactly the non deterministic fixed point logic (FO + NFP), and hence, by the equivalences mentioned above, it turned out that $\Sigma_1^{1,\omega}$ captured NP_r, analogously to the well known relationssip $\Sigma_1^1 = NP$ ([Fag,74]). Continuing the analogy, the characterization of the relational polynomial time hierarchy PH_r with *full* SO^ω was stated without proof in [Daw,98], and later proved by the second author jointly with F. Ferrarotti in [FT,08].

In [AT,14], aiming to characterize higher relational complexity classes, and as a natural continuation of the study of the logic SO^ω, we defined a variation of third order logic (TO) denoted as TO^ω, under finite interpretations. We defined it as a semantic restriction of TO where the (second order) relations which form the tuples in the third order relations that valuate the quantified third order variables are *closed* under the relation $\equiv_k$ as above. In [AT,14] we also introduced a variation of the non deterministic relational machine, which we denoted 3-NRM (for third order NRM), where we allow TO relations in the relational store of the machine. We defined the class $\text{NEXPTIME}_{3,r}$ as the class of 3-NRM's that work in time exponential in the $size_k$ (see above) of the input dbi. We then proved that the existential fragment of TO^ω, denoted $\Sigma_1^{2,\omega}$, captures $NEXPTIME_{3,r}$.

In the present article, we prove a stronger result: we show that the existential fragment of TO^ω *also captures* the relational complexity class $NEXPTIME_r$. Then, adding the result proved in this article, we have the following picture regarding the known characterizations of relational complexity classes up to now: $P_r = (\text{FO} +$ det-inf-FP $)$, $NP_r = (\text{FO} +$ ndet-inf-FP $) = \Sigma_1^{1,\omega}$, $PH_r = \text{SO}^\omega$, $PSPACE_r = (\text{FO} +$ alt-inf-FP $) = (\text{FO} +$ det-ninf-FP $) = (\text{FO} +$ ndet-ninf-FP $)$, $EXPTIME_r = (\text{FO} +$ alt-ninf-FP $)$, and $NEXPTIME_r = \Sigma_1^{2,\omega}$.

Then, as it turned out that $NEXPTIME_r = NEXPTIME_{3,r}$, an interesting consequence of our result is that RM's in their original formulation are strong enough as to *simulate* the existence of TO relations in their relational store and, hence, to also *simulate* the existence of TO^ω formulas in their finite control (without TO^ω or SO^ω quantifiers, as in 3-NRM's in [AT,14], see below).

That is, for every 3-NRM that works in time $NEXPTIME_{3,r}$, i.e., relational *third order* exponential time, in the $size_k$ of their input, there is an NRM that computes the same query, and that works in time $NEXPTIME_r$, i.e., relational exponential time in the $size_k$ of their input.

2 Preliminaries

We assume a basic knowledge of Logic and Model Theory (refer to [Lib,04]). We only consider vocabularies of the form $\sigma = \langle R_1, \ldots, R_s \rangle$ (i.e., *purely relational*), where the arities of the relation symbols are $r_1, \ldots, r_s \geq 1$, respectively. We

[3] in the sense of [FPT,10] these relations are *redundant* relations

assume that they also contain equality. And we consider only *finite* σ-structures, denoted as $\mathbf{A} = \langle A, R_1^{\mathbf{A}}, \ldots, R_s^{\mathbf{A}} \rangle$, where A is the domain, also denoted $dom(\mathbf{A})$, and $R_1^{\mathbf{A}}, \ldots, R_s^{\mathbf{A}}$ are (second order) relations in $A^{r_1}, \ldots, A^{r_s}$, respectively. If $\gamma(x_1, \ldots, x_l)$ is a formula of some logic with free FO variables $\{x_1, \ldots, x_l\}$, for some $l \geq 1$, with $\gamma^{\mathbf{A}}$ we denote the l-ary relation defined by γ in $\mathbf{A}$, i.e., the set $\{(a_1, \ldots, a_l) : a_1, \ldots, a_l \in A \wedge \mathbf{A} \models \gamma(x_1, \ldots, x_l)[a_1, \ldots, a_l]\}$. For any l-tuple $\bar{a} = (a_1, \ldots, a_l)$ of elements in A, with $1 \leq l \leq k$, we define the FO^k type of $\bar{a}$, denoted $Type_k(\mathbf{A}, \bar{a})$, to be the set of FO^k formulas $\varphi \in FO^k$ with free variables among $x_1, \ldots, x_l$, such that $\mathbf{A} \models \varphi[a_1, \ldots, a_l]$. If τ is an FO^k type, we say that the tuple $\bar{a}$ *realizes* τ in $\mathbf{A}$, if and only if, $\tau = Type_k(\mathbf{A}, \bar{a})$. Let $\mathbf{A}$ and $\mathbf{B}$ be σ-structures and let $\bar{a}$ and $\bar{b}$ be two l-tuples on $\mathbf{A}$ and $\mathbf{B}$ respectively, we write $(\mathbf{A}, \bar{a}) \equiv_k (\mathbf{B}, \bar{b})$, to denote that $Type_k(\mathbf{A}, \bar{a}) = Type_k(\mathbf{B}, \bar{b})$. If $\mathbf{A} = \mathbf{B}$, we also write $\bar{a} \equiv_k \bar{b}$. We denote as $size_k(\mathbf{A})$ the number of equivalence classes in $\equiv_k$ in $\mathbf{A}$. An l-ary relation R in $\mathbf{A}$ is *closed under* $\equiv_k$ if for any two l-tuples $\bar{a}$, $\bar{b}$ in A^l, $\bar{a} \in R \wedge \bar{a} \equiv_k \bar{b} \Rightarrow \bar{b} \in R$. Let S be a set, a binary relation R is a *pre-order* on S if it satisfies: 1) $\forall a \in S \;\; (a, a) \in R$ (reflexive). 2) $\forall a, b, c \in S \;\; (a, b) \in R \wedge (b, c) \in R \Rightarrow (a, c) \in R$ (transitive). 3) $\forall a, b \in S \;\; (a, b) \in R \vee (b, a) \in R$ (conex). A *pre-order* $\preceq$ on S induces an equivalence relation $\equiv$ on S (i.e., $a \equiv b \Leftrightarrow a \preceq b \wedge b \preceq a$), and also induces a total order over the set of equivalence classes of $\equiv$. When the equivalence classes induced by a pre-order on k-tuples from some structure $\mathbf{A}$ agree with the equivalence classes of $\equiv_k$, then the pre-order establishes a total order over the FO^k types for k-tuples which are realized on $\mathbf{A}$. We denote by $\Sigma_m^{1,\omega}[\sigma]$ the class of formulas of the form $\exists^{k_{11}} Y_{11}^{r_{11},k_{11}} \ldots \exists^{k_{1l_1}} Y_{1l_1}^{r_{1l_1},k_{1l_1}} \forall^{k_{21}} Y_{21}^{r_{21},k_{21}} \ldots \forall^{k_{2l_2}} Y_{2l_2}^{r_{2l_2},k_{2l_2}} \ldots Q^{k_{t1}} Y_{t1}^{r_{t1},k_{t1}} \ldots Q^{k_{tl_t}} Y_{tl_t}^{r_{tl_t},k_{tl_t}}(\phi)$, where the quantifiers $Q^{k_{t1}}, \ldots, Q^{k_{tl_t}}$ are $\forall^{k_{t1}}, \ldots, \forall^{k_{tl_t}}$, if t is even, or $\exists^{k_{t1}}, \ldots, \exists^{k_{tl_t}}$, if t is odd, ϕ is an FO formula in the vocabulary $\sigma \cup \{Y_{11}^{r_{11},k_{11}}, \ldots, Y_{tl_t}^{r_{tl_t},k_{tl_t}}\}$, with $r_{11} \leq k_{11}$, ..., $r_{tl_t} \leq k_{tl_t}$, respectively. We define $\mathrm{SO}^\omega = \bigcup_{m \geq 1} \Sigma_m^{1,\omega}$. The second order quantifier $\exists^k$ has the following semantics: let $\mathbf{I}$ be a σ-structure; then $\mathbf{I} \models \exists^k Y^{r,k} \varphi$ if there is an r-ary (second order) relation $R^{r,k}$ on I that is closed under the relation $\equiv_k$ in $\mathbf{I}$, and $(\mathbf{I}, R) \models \varphi$.

3 The Restricted Third-Order Logic TO^w and 3-NRM's

A *third order relation type* is a w-tuple $\tau = (r_1, \ldots, r_w)$ where $w, r_1, \ldots, r_w \geq 1$. In addition to the symbols of SO$^\omega$, the alphabet of TO$^\omega$ ([AT,14]) contains for every $k \geq 1$, a *third-order quantifier* $\exists^k$, and for every relation type τ such that $r_1, \ldots, r_w \leq k$ a countably infinite set of third order variables, denoted as $\mathcal{X}_1^{\tau,k}, \mathcal{X}_2^{\tau,k}, \ldots$, and called TO$^\omega$ *variables*. We use upper case Roman letters $X_i^{r,k}$ for *SO$^\omega$* variables (in this article we will often drop the superindex k, when it is clear from the context), where r is their arity, and lower case Roman letters for individual (i.e., FO) variables. Let σ be a relational vocabulary. A TO$^\omega$ *atomic formula* of vocabulary σ, on the TO$^\omega$ variable $\mathcal{X}^{\tau,k}$ is a formula of the form $\mathcal{X}^{\tau,k}(V_1, \ldots, V_w)$, where $V_1, \ldots, V_w$ are either second order variables of the form $X_i^{r_i,k}$, or relation symbols in σ, and whose arities are re-

spectively $r_1, \ldots, r_w \leq k$. Note that all the relations that form a σ-structure are closed under $\equiv_k$, since k is $\geq$ than all the arities in σ (see above, and Fact 9 in [FT,08]). Let $m \geq 1$. We denote by $\Sigma_m^{2,\omega}[\sigma]$ the class of formulas of the form $\exists^{k_{3,11}} \mathcal{X}_{11}^{\tau_{11},k_{3,11}} \ldots \exists^{k_{3,1s_1}} \mathcal{X}_{1s_1}^{\tau_{1s_1},k_{3,1s_1}} \forall^{k_{3,21}} \mathcal{X}_{21}^{\tau_{21},k_{3,21}} \ldots \forall^{k_{3,2s_2}} \mathcal{X}_{2s_2}^{\tau_{2s_2},k_{3,2s_2}} \ldots Q^{k_{3,m1}} \mathcal{X}_{m1}^{\tau_{m1},k_{3,m1}} \ldots Q^{k_{3,ms_m}} \mathcal{X}_{ms_m}^{\tau_{ms_m},k_{3,ms_m}}(\psi)$, where for $i, j \geq 1$, with $\tau_{ij} = (r_{ij,1}, \ldots, r_{ij,w_{ij}})$, it is $r_{ij,1}, \ldots, r_{ij,w_{ij}} \leq k_{3,ij}$, Q is either $\exists^k$ or $\forall^k$, for some k, depending on whether m is odd or even, respectively, and ψ is an SO^ω formula with the addition of TO^ω atomic formulas. As usual, $\forall^k \mathcal{X}^{\tau,k}(\psi)$ abbreviates $\neg\exists^k \mathcal{X}^{\tau,k}(\neg\psi)$. We define $\mathrm{TO}^\omega = \bigcup_{m\geq 1} \Sigma_m^{2,\omega}$.

A TO^ω *relation* $\mathcal{R}^{\tau,k}$ *of type* τ *and closed under* $\equiv_k$ on a σ structure $\mathbf{I}$ is a set of w tuples $(R_1^{r_1,k}, \ldots, R_w^{r_w,k})$ of (second order) relations on $\mathbf{I}$ with respective arities $r_1, \ldots, r_w \leq k$, closed under $\equiv_k$. The third order quantifier $\exists^k$ has the following semantics: let $\mathbf{I}$ be a σ-structure; then $\mathbf{I} \models \exists^k \mathcal{X}^{\tau,k} \varphi$ if there is a TO^ω relation $\mathcal{R}^{\tau,k}$ of type τ on I closed under the relation $\equiv_k$ in $\mathbf{I}$, such that $(\mathbf{I}, \mathcal{R}) \models \varphi$. Here $(\mathbf{I}, \mathcal{R})$ is the *third order* $(\sigma \cup \{\mathcal{X}^{\tau,k}\})$ structure expanding $\mathbf{I}$, in which $\mathcal{X}$ is interpreted as $\mathcal{R}$. Note that a *valuation* in this setting also assigns to each second order variable $X^{r,k}$ a (second order) relation on I of arity r that is closed under $\equiv_k$ in $\mathbf{I}$, and to each third order variable $\mathcal{X}^{\tau,k}$ a third order relation $\mathcal{R}^{\tau,k}$ on I of type τ, closed under $\equiv_k$ in $\mathbf{I}$. We don't allow free second or third order variables in the logics SO^ω and TO^ω. Note that allowing elements (from the domain of the structure) in a third order relation type would change the semantics of TO^ω, since we could use a third order relation of such type to simulate a second order relation *not closed* under $\equiv_k$. See [AT,14] for an example of a non trivial query in $\Sigma_1^{2,\omega}$.

A *third order non-deterministic relational machine* ([AT,14]), denoted as 3-NRM, of *arity* k, for $k \geq 1$, is a 11-tuple $\langle Q, \Sigma, \delta, q_0, \mathfrak{b}, F, \sigma, \tau, T, \Omega, \Phi \rangle$ where: Q is the finite set of internal states; $q_0 \in Q$ is the initial state; Σ is the finite tape alphabet; $\mathfrak{b} \in \Sigma$ is the symbol denoting blank; $F \subseteq Q$ is the set of accepting states; τ is the finite vocabulary of the *rs* (its *relational store*), with finitely many TO^ω relation symbols $\mathcal{R}_i^{\tau_i,k'}$ of any arbitrary type $\tau_i = (r_{i1}, \ldots, r_{iw})$, with $1 \leq r_{i1}, \ldots, r_{iw} \leq k' = k$, and finitely many SO^ω relation symbols $R_i^{r_i,k''}$ of arities $r_i \leq k'' = k$; $T \in \tau$ is the output relation; σ is the vocabulary of the input structure; Ω is a finite set of TO^ω formulas with up to k FO variables, with *no* SO^ω *or* TO^ω *quantifiers*, and with no free variables of any order (i.e., all the SO^ω and TO^ω relation symbols are in τ); Φ is a finite set of TO^ω formulas with up to k FO variables, that are not sentences, with *no* SO^ω *or* TO^ω *quantifiers*, and where the free variables are either *all* FO variables, or *all* SO^ω variables; $\delta : Q \times \Sigma \times \Omega \to \mathcal{P}(\Sigma \times Q \times \{R, L\} \times \Phi \times \tau)$ is the transition function. In any pair in δ, if φ, S occur in the 5-tuple of its second component, for Φ and τ, then either S is a TO^ω relation symbol $\mathcal{R}_i^{\tau_i,k'}$ in rs and φ has $|\tau_i|$ SO^ω free variables $X_1^{r_1,k''}, \ldots, X_{|\tau_i|}^{r_{|\tau_i|},k''}$ with arities according to τ_i, and $1 \leq r_1, \ldots, r_{|\tau_i|} \leq k'' = k' = k$, or S is an SO^ω relation symbol $R_i^{r_i,k''}$ in rs and φ has $1 \leq r_i \leq k'' = k$ FO free variables. At any stage of the computation of a 3-NRM on an input σ-structure $\mathbf{I}$, there is one relation in its *rs* of the corresponding relation type

(or arity) in $\mathbf{I}$ for each relation symbol in τ, so that in each transition there is a (finite) τ-structure $\mathbf{A}$ in the *rs*, which we can *query* and/or *update* through the formulas in Ω and Φ, respectively, and a finite Σ string in its tape, which we can access as in Turing machines. The concept of *computation* is analogous to that in the Turing machine. We define the complexity class $NEXPTIME_{3,r}$ as the class of the *relational languages* or *Boolean queries* (i.e., sets of finite structures of a given relational vocabulary, closed under isomorphisms) that are decidable by 3-NRM machines of *some* arity k', that work in non deterministic exponential time in the number of equivalence classes in $\equiv_{k'}$ of the input structure. In symbols: $NEXPTIME_{3,r} = \bigcup_{c\in N} NTIME_{3,r}(2^{c\cdot(size_k)})$ (as usual, this notation *does not mean* that the arity of the 3-NRM must be k).

A *non-deterministic relational machine*, i.e., an NRM in its classical formulation, denoted as NRM, of *arity* k, for $k \geq 1$, is a 11-tuple as above, where the formulas in Ω and Φ are FO formulas with up to k FO variables, in the vocabulary τ, and where all the relations in the *relational store* are second order relations of arity at most k. The relational complexity class $NEXPTIME_r$ is the class of the *relational languages* or *Boolean queries* that are decidable by NRM machines of *some* arity k', that work in non deterministic exponential time in the number of equivalence classes in $\equiv_{k'}$ of the input structure. In symbols: $NEXPTIME_r = \bigcup_{c\in N} NTIME_r(2^{c\cdot(size_k)})$.

In [AT,14] we proved the following results:

Theorem 1. *([AT,14]) $NEXPTIME_{3,r} \subseteq \Sigma_1^{2,\omega}$. That is, given a 3-NRM M in $NTIME_{3,r}(2^{c\cdot(size_k)})$, for some positive integer c, and with input vocabulary σ that computes a Boolean query q we can build a formula $\varphi_M \in \Sigma_1^{2,\omega}$ such that, for every σ-structure $\boldsymbol{I}$, M accepts $\boldsymbol{I}$ iff $\boldsymbol{I} \models \varphi_M$.*

Theorem 2. *([AT,14]) $\Sigma_1^{2,\omega} \subseteq NEXPTIME_{3,r}$. That is, every class of relational structures definable in $\Sigma_1^{2,\omega}$ is in $\bigcup_{c\in N} NTIME_{3,r}(2^{c\cdot(size_k)})$.*

4 Existential TO$^\omega$ captures NEXPTIME$_r$

Corollary 3 *$NEXPTIME_r \subseteq \Sigma_1^{2,\omega}$. That is, given an NRM M that works in $NTIME_r(2^{c\cdot(size_k)})$, for some positive integer c, and with input vocabulary σ that computes a Boolean query q we can build a formula $\varphi_M \in \Sigma_1^{2,\omega}$ such that, for every σ-structure $\boldsymbol{I}$, M accepts $\boldsymbol{I}$ iff $\boldsymbol{I} \models \varphi_M$.*

Proof. This is a consequence of Theorem 1 by the following two immediate facts: 1) an NRM is a special case of a 3-NRM, with no third order relations in its rs, and 2) an NRM M is in $NEXPTIME_r$ iff M, as a 3-NRM, it is in $NEXPTIME_{3,r}$. □

Theorem 4. *$\Sigma_1^{2,\omega} \subseteq NEXPTIME_r$. That is, every class of relational structures definable in $\Sigma_1^{2,\omega}$ is in $\bigcup_{c\in N} NTIME_r(2^{c\cdot(size_k)})$.*

Proof. Let σ be a relational vocabulary, let φ be a $\Sigma_1^{2,\omega}[\sigma]$ sentence of the form $\exists^{k_{3,1}}\mathcal{X}_1^{\tau_1,k_{3,1}}\ldots\exists^{k_{3,s}}\mathcal{X}_s^{\tau_s,k_{3,s}}(\psi)$, where ψ is a $\Sigma_t^{1,\omega}$ formula, for some $t \geq 1$, with atomic TO^ω formulas formed with the TO^ω variables $\mathcal{X}_1,\ldots,\mathcal{X}_s$. For the sake of a simpler presentation we assume w.l.o.g. that for $1 \leq i \leq s$ the type of the relation $\mathcal{X}_i$ is $\tau_i = (r_{3,i},...,r_{3,i})$ of cardinality $r_{3,i}$, with $r_{3,i} \leq k_{3,i}$.
Suppose the $\Sigma_t^{1,\omega}$ formula ψ is of the form $\exists^{k_{2,11}}Y_{11}^{r_{2,11},k_{2,11}}\ldots\exists^{k_{2,1l_1}}Y_{1l_1}^{r_{2,1l_1},k_{2,1l_1}}$ $\forall^{k_{2,21}}Y_{21}^{r_{2,21},k_{2,21}}\ldots\forall^{k_{2,2l_2}}Y_{2l_2}^{r_{2,2l_2},k_{2,2l_2}}\ldots Q^{k_{2,t1}}Y_{t1}^{r_{2,t1},k_{2,t1}}\ldots Q^{k_{2,tl_t}}Y_{tl_t}^{r_{2,tl_t},k_{2,tl_t}}(\phi)$, where the quantifiers $Q^{k_{2,t1}},\ldots,Q^{k_{2,tl_t}}$ are $\forall^{k_{2,t1}},\ldots,\forall^{k_{2,tl_t}}$, if t is even, or $\exists^{k_{2,t1}},\ldots,\exists^{k_{2,tl_t}}$, if t is odd, ϕ is an FO formula in the vocabulary $\sigma\cup\{Y_{11}^{r_{2,11},k_{2,11}},$ $\ldots,Y_{tl_t}^{r_{2,tl_t},k_{2,tl_t}}\}$, with atomic TO^ω formulas, and $r_{2,11} \leq k_{2,11}$, ..., $r_{2,tl_t} \leq k_{2,tl_t}$, respectively. We now build an NRM M_φ which accepts a given σ structure $\mathbf{I}$ iff $\mathbf{I} \models \varphi$. It is known that for every σ, and every $k \geq 1$, a formula $\gamma(\bar{x},\bar{y})$ with $k'' \geq 2k$ variables of the fixed point logic $(FO + LFP)$ can be built s. t. on any σ structure $\mathbf{J}$, γ defines a pre-order $\preceq_k$ in the set of k-tuples of $\mathbf{J}$, whose induced equivalence relation is $\equiv_k$ (see T.11.20 in [Lib,04]). On the other hand, it is known that $(FO+LFP)$ captures *relational polynomial time* P_r ([AVV,97]). Hence, an RM $M_{\preceq_k}$ of some arity $k' \geq 2k$ can be built, that constructs, on input $\mathbf{J}$, the pre-order $\preceq_k$ in $\mathbf{J}$, in time polynomial in $size_{k'}(\mathbf{J})$. We define the arity of M_φ as $k = max(\{k'_{3,1},\ldots,k'_{3,s},k'_{2,11},\ldots,k'_{2,tl_t}\})$, where the k'_{ij}'s are the arities of the RM's $M_{\preceq_{k_{3,1}}}$, ..., $M_{\preceq_{k_{2,tl_t}}}$, respectively.

Let $\mathbf{I}$ be the input structure. M_φ works as follows: **1):** M_φ simulates the RM's $M_{\preceq_{k_{3,1}}}$, ..., $M_{\preceq_{k_{2,tl_t}}}$, to build the pre-orders $\preceq_{k_{3,1}}$, ..., $\preceq_{k_{2,tl_t}}$, respectively. M_φ builds those pre-orders in time polynomial in $size_{k'_{3,1}}(\mathbf{I}),\ldots,size_{k'_{2,tl_t}}(\mathbf{I})$, respectively. As all these arities are $\leq k$ (see above), that time is also polynomial in $size_k(\mathbf{I})$ (see [FT,08]). **2):** By stepping through the equivalence classes of the relation $\equiv_{k_{3,1}}$ in the order given by $\preceq_{k_{3,1}}$, M_φ computes $size_{k_{3,1}}(\mathbf{I})$, and the same process is followed to compute $size_{k_{3,2}}(\mathbf{I})$, ..., $size_{k_{2,tl_t}}(\mathbf{I})$ by using the equivalence relations $\equiv_{k_{3,2}}$, ..., $\equiv_{k_{2,tl_t}}$, and the pre-orders $\preceq_{k_{3,2}}$, ..., $\preceq_{k_{2,tl_t}}$, respectively (recall that the pre-orders $\preceq_{k_{3,1}}$, ..., $\preceq_{k_{2,tl_t}}$, induce total orders in the equivalence classes of the corresponding equivalence relations $\equiv_{k_{3,1}}$, ..., $\equiv_{k_{2,tl_t}}$). Note that by the choice of k, all these computations are done by M_φ in time polynomial in $size_k(\mathbf{I})$. **3):** M_φ needs to *guess* the TO^ω relations $\mathcal{S}_1^{\tau_1},\ldots,\mathcal{S}_s^{\tau_s}$, as interpretations of the TO^ω variables $\mathcal{X}_1^{\tau_1,k_{3,1}},\ldots,\mathcal{X}_s^{\tau_s,k_{3,s}}$ respectively. Each $\mathcal{S}_i^{\tau_i}$ is a set of $r_{3,i}$-tuples of $r_{3,i}$-ary (SO) relations closed under $\equiv_{k_{3,i}}$. To represent $\mathcal{S}_i^{\tau_i}$ we use three sorts of bit strings as follows: **a)** each bit string of sort $b^3_{R^{r_{3,i},k_{3,i}}}$ of size $size_{k_{3,i}}(\mathbf{I})$ represents one of the possible $r_{3,i}$-ary (SO) relations on $\mathbf{I}$, closed under $\equiv_{k_{3,i}}$; note that each bit represents one equivalence class in $\equiv_{k_{3,i}}$, following from left to right the total order induced by $\preceq_{k_{3,i}}$; **b)** each bit string of sort $b^2_{\bar{R}^{r_{3,i},k_{3,i}}}$ of size $r_{3,i}\cdot size_{k_{3,i}}(\mathbf{I})$ represents one of the possible $r_{3,i}$-tuples of $r_{3,i}$-ary (SO) relations on $\mathbf{I}$, closed under $\equiv_{k_{3,i}}$; **c)** each bit string of sort $b^1_{\mathcal{S}_i^{\tau_i,k_{3,i}}}$ of size $2^{r_{3,i}\cdot size_{k_{3,i}}(\mathbf{I})}$ represents one of the possible sets of $r_{3,i}$-tuples of $r_{3,i}$-ary (SO) relations on $\mathbf{I}$, closed under $\equiv_{k_{3,i}}$, i.e., one of the possible TO^ω relations on $\mathbf{I}$ of type τ_i, closed under $\equiv_{k_{3,i}}$. Let b be a bit string of sort b^1.

Each bit in b represents one of the possible bit strings of sort $b^2_{\bar{R}^{r_{3,i},k_{3,i}}}$ of size $r_{3,i} \cdot size_{k_{3,i}}(\mathbf{I})$. The *leftmost* bit in b represents a bit string of type $b^2_{\bar{R}^{r_{3,i},k_{3,i}}}$ that has all its bits 0, i.e., it is the bit string that corresponds to the $r_{3,i}$-tuple formed by $r_{3,i}$ empty $r_{3,i}$-ary relations. The following bits in b represent the bit strings of sort $b^2_{\bar{R}^{r_{3,i},k_{3,i}}}$ that correspond to the order in all the possible bit strings of sort $b^2_{\bar{R}^{r_{3,i},k_{3,i}}}$ according to their binary value. And so on, up to the *rightmost* bit in b, which represents a bit string of sort $b^2_{\bar{R}^{r_{3,i},k_{3,i}}}$ that has all its bits 1 (i.e., it is the bit string that corresponds to the $r_{3,i}$-tuple formed by $r_{3,i}$ copies of the $r_{3,i}$-ary relation that has the $r_{3,i}$-tuples in all the equivalence classes in the relation $\equiv_{k_{3,i}}$). Then, M_φ guesses s bit strings of sort $b^1_{\mathcal{S}_i^{\tau_i}}$ of size $2^{r_{3,i} \cdot size_{k_{3,i}}(\mathbf{I})}$, one for each one of the relations $\mathcal{S}_i^{\tau_i}$. Note that this is done in time $2^{c \cdot size_{k_{3,i}}(\mathbf{I})}$, and hence also in time $2^{d \cdot size_k(\mathbf{I})}$, since $k_{3,i} \leq k$ (see above), for some constants c, d. **4):** Regarding the SO$^\omega$ variables quantified in the $\Sigma_t^{1,\omega}$ formula ψ, to interpret each of them we *build all* the possible SO$^\omega$ relations of the corresponding arity and closed under the corresponding equivalence class in the the rs of M_φ. We build those relations by stepping in the equivalence classes of tuples $\equiv_{k_{2,ij}}$ according to the total orders induced by the corresponding pre-orders $\preceq_{k_{2,ij}}$. The details on how to do that are equal to the algorithm used in [FT,08] to prove $\Sigma_1^{1,\omega} \subseteq NTIME_r((size_k)^c)$. Note that we can afford to do that because for each variable $Y_{ij}^{r_{2,ij},k_{2,ij}}$ the number of such relations is bounded by $2^{d \cdot size_{k_{2,ij}}(\mathbf{I})}$, and hence also by $2^{d \cdot size_k(\mathbf{I})}$, since $k_{2,ij} \leq k$ (see above), for some constant d that depends on the arity. Then, for each SO$^\omega$ variable $Y_{ij}^{r_{2,ij},k_{2,ij}}$ we will require that either *for all* the generated relations, or for *at least one* of them, depending on the corresponding quantifier being $\forall$ or $\exists$, respectively, the formula ϕ is true. **5): Evaluation of ϕ:** Recall that ϕ is an FO formula with atomic TO$^\omega$ formulas. To evaluate ϕ we consider the syntax tree of ϕ, T_ϕ, and evaluate one node of it at a time in the finite control of M_φ, in a bottom up direction. To that end, for every node α in T_ϕ, that represents a sub-formula with $r \geq 1$ free FO variables, we define in the rs an r-ary relation variable R_α. And for every node α in T_ϕ, that represents a sub-formula with *no* free FO variables, we define in the rs a 1-ary relation variable B_α that represents a Boolean variable, which we interpret as True if $B_\alpha = dom(\mathbf{I})$, and as False if $B_\alpha = \emptyset$. Note that all the SO$^\omega$ relations that appear in the nodes in T_ϕ are in the rs of M_φ. Every node in T_ϕ is of one of the following kinds: i) an atomic FO formula with a relation symbol either in σ or quantified by an SO$^\omega$ quantifier in ψ, ii) a $\vee$ connective, iii) a $\wedge$ connective, iv) a $\neg$ connective, v) an existential FO quantifier, or vi) an atomic TO$^\omega$ formula with a relation symbol quantified by a TO$^\omega$ quantifier in φ. We omit the details on how to evaluate the nodes of the first 5 kinds, since they are straightforward, and focus on the nodes that correspond to atomic TO$^\omega$ formulas. Suppose a given node α in T_ϕ corresponds to the sub-formula $\mathcal{X}_i^{\tau_i,k_{3,i}}(V_1^{r_{3,i},k_{3,i}}, \ldots, V_{r_{3,i}}^{r_{3,i},k_{3,i}})$

with $\tau_i = (r_{3,i}, ..., r_{3,i})$ of cardinality $r_{3,i}$, with $r_{3,i} \leq k_{3,i}$ as stated in the beginning of the proof, and where $V_1^{r_{3,i},k_{3,i}}, \ldots, V_{r_{3,i}}^{r_{3,i},k_{3,i}}$ are either relation symbols in σ or quantified by an SO$^\omega$ quantifier in ψ. We check whether or not the $r_{3,i}$-tuple

of relations $(V_1^{r_{3,i},k_{3,i}},\ldots,V_{r_{3,i}}^{r_{3,i},k_{3,i}})$ is in the TO$^\omega$ relation $\mathcal{S}_i^{\tau_i,k_{3,i}}$ guessed above for the variable $\mathcal{X}_i^{\tau_i,k_{3,i}}$, using the (guessed) bit string $b^1_{\mathcal{S}_i^{\tau_i,k_{3,i}}}$ that represents $\mathcal{S}_i^{\tau_i,k_{3,i}}$, with the following algorithm, that clearly runs in time $2^{c\cdot size_{k_{3,i}}(\mathbf{I})}$, and hence also in time $2^{d\cdot size_k(\mathbf{I})}$, since $k_{3,i}\leq k$ (see above), for some constants c,d:

- $B_\alpha \leftarrow \emptyset$ (*i.e.*, $B_\alpha \leftarrow FALSE$);
- **for all** bit strings of sort $b^2_{\bar{R}^{r_{3,i},k_{3,i}}}$, $(a_1,\ldots,a_{2^{(r_{3,i}\cdot size_{k_{3,i}}(\mathbf{I}))}})$, counting in binary, varying n from 1 through $2^{r_{3,i}\cdot size_{k_{3,i}}(\mathbf{I})}$ (*i.e., for all $r_{3,i}$-tuples of $r_{3,i}$-ary (SO) relations closed under $\equiv_{k_{3,i}}$*);
  - **for** $j=1$ through $r_{3,i}$ (*i.e., the j-th component in the tuple of (SO) rel.*);
    - $S_{i,j}^{r_{3,i},k_{3,i}} \leftarrow \emptyset$;
    - **for** $l=1$ through $size_{k_{3,i}}(\mathbf{I})$ (*i.e., bit l in bit substring of sort $b^3_{R_j^{r_{3,i},k_{3,i}}}$*);
      - **if** bit m of bit string a_n is 1, where $m=(j-1)\cdot size_{k_{3,i}}(\mathbf{I})+l$, (*i.e., bit m in a bit string of sort $b^2_{\bar{R}^{r_{3,i},k_{3,i}}}$*)
        - add to $S_{i,j}^{r_{3,i},k_{3,i}}$ the l-th equivalence class in $\equiv_{k_{3,i}}$, according to pre-order $\preceq_{k_{3,i}}$ (*i.e., all the $r_{3,i}$-tuples of elements in that class*);
    - **end** l;
  - **end** j;
  - **if** bit n in bit string $b^1_{\mathcal{S}_i^{\tau_i,k_{3,i}}}=1$
    - **if** $(V_1^{r_{3,i},k_{3,i}}=S_{i,1}^{r_{3,i},k_{3,i}}\wedge\ldots\wedge V_{r_{3,i}}^{r_{3,i},k_{3,i}}=S_{i,r_{3,i}}^{r_{3,i},k_{3,i}})$
      - $B_\alpha \leftarrow dom(\mathbf{I})$ (*i.e.*, $B_\alpha \leftarrow TRUE$);
- **end all**; □

5 Conclusions

From Theorems 1, 2, 4, and Corollary 3, we have the following result:

Corollary 5 *Let M_3 be a 3-NRM that works in $NTIME_{3,r}(2^{c\cdot(size_k)})$, for some positive integer c, that computes a Boolean query q. Then, there is a NRM M_2 that works in $NTIME_r(2^{d\cdot(size_k)})$, for some positive integer d, that also computes q.* □

This is very interesting, since in the general case it is *much easier* to define an NRM using TO relations in its rs, and TO formulas to access it, than restricting the machine to SO relations in its rs, and SO formulas. Then, to prove that a given query is computable by an NRM it is enough with showing that it can be computed by a 3-NRM. Note however, that we think that we still *need* 3-NRM's as well as the third order relational complexity class $NEXPTIME_{3,r}$, if we need to work with *oracle* NRM's with third order relations, since as the oracle cannot access the tape of the base machine (see [FT,08]), there seems to be no way to pass the bit strings that represent TO relations from the base to the oracle.

Recall that it has been proved that RM's have the same computation, or expressive power, as the (effective fragment of the) well known infinitary logic

with finitely many variables $\mathcal{L}^{\omega}_{\infty\omega}$ ([AVV,95]). On the other hand, analogously to the well known result that states that the computation power of deterministic and non deterministic Turing machines is the same, it is straightforward to see that any NRM M_n can be simulated by a (deterministic) RM M_d working in relational time exponentially higher, just by checking in M_d all possible transitions instead of guessing one in each non deterministic step of the transition relation of M_n. Then, the following is immediate:

Corollary 6 $\Sigma^{2,\omega}_1 \subseteq$ *(effective fragment of)* $\mathcal{L}^{\omega}_{\infty\omega}$. □

Finally, in [GT,10], the logic SO^F was introduced and defined as a semantic restriction of SO where the valuating r-ary relations for the quantified SO variables are *closed* under the relation $\equiv_F$ of equality of FO types in the set of r-tuples of the structure. It was shown there that its existential fragment $\Sigma^{1,F}_1$ is *not* included in $\mathcal{L}^{\omega}_{\infty\omega}$, as opposite to $\Sigma^{1,\omega}_1$ which is. Then, we have the following result:

Corollary 7 $\Sigma^{1,F}_1 \subsetneq \Sigma^{2,\omega}_1$. □

References

1. [AT,14] J. Arroyuelo, J. M. Turull-Torres, "The Existential Fragment of Third Order Logic and Third Order Relational Machines", in Proceedings of the XIX Argentine Conference on Computer Science CACIC 2014", ISBN 978-987-3806-05-6, Buenos Aires, October 20-24, p. 324-333, 2014.
2. [AV,91] Abiteboul, S., Vianu, V., "Generic Computation and its Complexity", STOC 1991.
3. [AVV,95] Abiteboul, S., Vardi, M., Vianu, V., "Computing with Infinitary Logic", Theoretical Computer Science 149, 1, pp. 101-128, 1995.
4. [AVV,97] Abiteboul, S., Vardi, M. Y., Vianu, V., "Fixpoint logics, relational machines, and computational complexity", JACM 44 (1997) 30-56.
5. [Daw,98] Dawar, A., "A restricted second order logic for finite structures". Information and Computation 143 (1998) 154-174.
6. [Fag,74] Fagin, R., "Generalized First-Order Spectra and Polynomial-Time Recognizable Sets", in "Complexity of Computations", edited by R. Karp, SIAM-AMS Proc., American Mathematical Society, Providence, RI, pp. 27–41, 1974.
7. [FPT,10] F. A. Ferrarotti, A. L. Paoletti, J. M. Turull-Torres, "Redundant Relations in Relational Databases: A Model Theoretic Perspective", Journal of Universal Computer Science, Vol. 16, No. 20, pp. 2934-2955, 2010. http://www.jucs.org/jucs_16_20/redundant_relations_in_relational
8. [FT,08] Ferrarotti, F. A., Turull-Torres, J. M., "The Relational Polynomial-Time Hierarchy and Second-Order Logic", invited for "Semantics in Databases", edited by K-D. Schewe and B. Thalheim, Springer LNCS 4925, 29 pages, 2008.
9. [GT,10] Grosso, A. L., Turul-Torres J. M., "A Second-Order Logic in which Variables Range over Relations with Complete First-Order Types", 2010 XXIX International Conference of the Chilean Computer Science Society (SCCC) IEEE, p. 270-279, 2010.
10. [Lib,04] Libkin, L., "Elements of Finite Model Theory", Springer, 2004.
11. [Tur,06] J. M. Turull-Torres, "Relational Databases and Homogeneity in Logics with Counting", Acta Cybernetica, Vol 17, number 3, pp. 485-511, 2006.

Recuperación de noticias públicas ordenadas temporalmente y por zonas utilizando funciones combinadas de Apache Solr

Luis Ignacio Aita[1], Damián Barry[1], Carlos Buckle[1], Claudio Delrieux[1 2]

LINVI, Departamento de Informática, Facultad de Ingeniería, UNPSJB, Puerto Madryn 9120, Argentina,
IMAGLAB, Departamento de Ingeniería Eléctrica y Computadoras, UNS, 8000 Bahía Blanca, Argentina,
I.ignacioaita@sursoftware.com.ar, II.damian_barry@unpata.edu.ar,
WWW home page:http://madryn.unp.edu.ar

Resumen El presente trabajo se enfoca en investigar y mejorar las técnicas de recuperación de información sobre una base de datos existente de noticias obtenidas en redes sociales y diarios en línea. Dicha información ya se encuentra clasificada e indexada por diversos atributos, entre los cuales vale destacar la pertenencia geográfica de la noticia, tanto en forma semántica (teniendo en cuenta localidad, partido, provincia, etc. en forma jerárquica) como geográfica (latitud, longitud). Este desarrollo propone comprobar que es posible mejorar la combinación de funciones de búsqueda y recuperación de la información almacenada en grandes volúmenes de información mediante la utilización índices existentes, principalmente en relación con la pertenencia geográfica y la antigüedad de la información buscada.

Keywords: Big Data, Boosting, Information Retrieval, Ecuaciones de Búsqueda, Apache Solr

1. Introducción

Este trabajo describe el desarrollo y resultados obtenidos en el proyecto *"Zonales"*[1], el cual consiste en un gestor de contenidos web (www.zonales.com) que brinda un conjunto de servicios de información reunidos en un mismo lugar que poseen un factor en común: la georreferenciación de la información y de servicios desde una perspectiva local para cada comunidad. Dentro del portal, cada usuario puede consultar y publicar contenidos relacionados con su localidad y zonas

[1] El proyecto se realizó en el marco de un convenio de colaboración entre la empresa Mediabit S.A., especializada en comunicación, asesoramiento e integración tecnológica orientada a los negocios en Internet (Internet Bussines Provider), y el equipo de vinculación y transferencia tecnológica de la Sede Puerto Madryn de la Universidad Nacional de la Patagonia San Juan Bosco, conformado con la finalidad de mejorar la calidad tecnológica de las organizaciones destinatarias, brindando cursos y realizando convenios con empresas y proyectos de extensión con empresas.

de interés, generando información desde la periferia hacia el centro. El desarrollo permitió que la información clasificada por zona y por fuente se recupere utilizando una combinación de funciones espec cuyos resultados se presentan en el presente trabajo.

La motivación principal de este trabajo consiste en la necesidad de diseñar, investigar y probar diversas técnicas para combinar funciones de recuperación y filtros de información, para que los datos recuperados sean de interés para el usuario en el marco geográfico-temporal, teniendo en cuenta la gran cantidad de información mayormente irrelevante que se puede generar en la base de datos. Dichas técnicas permitirán establecer las capacidades necesarias con las que debería contar una base de datos de información masiva, tanto desde la perspectiva de almacenamiento y técnicas de indexación, como de distribución de las consultas, escalabilidad y rendimiento en ambientes heterogéneos.

2. Componentes del proyecto Zonales

2.1. Motor de extracción de información

Para la extracción de contenido público de Internet se desarrollo un motor de extracción de contenido público de Internet denominado *"ZCrawler"*[4]. El mismo se compone de un lenguaje de extracción que le permite a los usuarios mediante una especificación formal comenzar a extraer información siguiendo múltiples criterios y permitiendo la clasificación de esta información según a quien se está evaluando. Para ello se desarrollaron servicios de extracción para las redes sociales Facebook y Twitter y para medios digitales que contaran con RSS estándar. Además se han adaptado algunos parsers específicos para blogs y diarios en línea que no contaban con una implementación estándar de RSS. En particular se implementaron conectores para los diarios Jornada y Chubut, ambos de la provincia de Chubut. Adicionalmente para los grandes medios nacionales se adaptaron extractores ya que sus definiciones de RSS no accedían ni a los comentarios ni a las referencias (links) externas ni a las imágenes.

A partir de poder extraer imágenes de las noticias y comentarios se mejoró la herramienta permitiendo realizar un tratamiento común a todo el material multimedia asociado a una publicación realizando una gestión uniforme tanto para Facebook, Twitter y las páginas sindicadas. La última versión generada de Zonales estuvo en linea aproximadamente seis meses en forma totalmente funcional. El volumen de datos que se generó durante ese tiempo fue de aproximadamente un millón y medio de documentos. El importante volumen de noticias extraídas (aproximadamente 8.500 noticias diarias y clasificadas de acuerdo a su pertenecía geográfica, ya sea por su emisor o por auto-clasificación según criterios de contenido) provocó el desafío de recuperar la información almacenada de forma efectiva y eficaz. Esto es especialmente importante, debido a que se estaban extrayendo noticias e información de solamente 15 municipios de los 2112 que existen en la República Argentina, lo que demandaba la construcción de una cuidadosa ecuación que garantizara los requerimientos de pertenencia geográfica

y ordenamiento temporal, además de las búsquedas tradicionales por términos que se desearan realizar.

Proceso de Extracción El proceso de extracción de la información para almacenar en el índice se divide en dos etapas claramente diferenciadas: configuración y extracción.

Como se puede apreciar el proceso consta de 3 fases bien definidas:

1. Definición de ecuaciones de extracción.
2. Ejecución de las extracciones de acuerdo a sus parámetros de planificación.
3. Clasificación e Indexación de las noticias extraídas. A su vez esta fase se divide en las siguientes tareas:
   a) Extraer de la noticia por fuente.
   b) Expandir la información no provista por la fuente.
   c) Clasificar la información.
   d) Indexar y almacenar la información en Apache Solr.

2.2. Recuperación de la Información

El desafío planteado de extracción de múltiples fuentes de información impone otro desafío que el proyecto debía resolver: la escalabilidad, debido al gran procesamiento de información extraída. En resultados previos (ver[6]) se pudo arribar a un conjunto de conclusiones relevantes en este trabajo, entre las cuales mencionamos las siguientes:

- La búsqueda secuencial de cualquier tipo de información presenta varios problemas, siendo el principal la falta de escalabilidad. Una solución a este inconveniente es el uso de estructuras de datos que permitan ser rápidamente consultadas.
- El indexado transforma los datos desde su forma original en una estructura que facilita la búsqueda y recuperación de los mismos en forma rápida y precisa, por ejemplo un índice invertido[5], un índice de citas, una matriz o un árbol.
- El proceso de indexado generalmente requiere un análisis y procesamiento de los documentos a incluir en el índice: lematización, tokenización, análisis fonético, etc. Estos pasos introducen problemas y desafíos importantes al momento del procesamiento[5], que no son alcance del presente trabajo.
- Dentro del estudio del presente trabajo, debido a la facilidad de implementación en un ambiente heterogéneo, se llegó a la conclusión que una solución óptima requeriría el uso del concepto de base de datos de partición horizontal o sharding [2].
- Apache Solr[7], que utiliza una base de datos no convencional para almacenar el índice de búsqueda (listas invertidas), provee múltiples capacidades para el escalado horizontal, permitiendo dividir la carga de trabajo entre múltiples instancias. Esto permite una fragmentación horizontal de los mismos entre múltiples servidores Apache Solr, a los cuales se les denomina shards[5]. Las

búsquedas son luego redirigidas a cada shard, y finalmente una respuesta única es construida en base a los resultados obtenidos de cada uno. Esta técnica es utilizada especialmente cuando se cuenta con un gran volumen de datos sobre el cual realizar consultas."

Adicionalmente al desafío de escalabilidad y rendimiento de la recuperación de información en grandes volúmenes de datos sobre Apache Solr, se platea la necesidad de filtrado por criterios y parámetros de búsqueda como de boosting (ordenamiento) de la misma.

Para resolver este último desafío el mismo se encara en 2 etapas bien definidas:

1. Diseño de filtros y boosting (ordenamiento) en un entorno de pruebas.
2. Implementación en un entorno real sobre la información extraída (gran volumen de datos)

3. Diseño e implementación de las Ecuaciones de Búsqueda

El alcance del entorno de pruebas para el presente trabajo está limitado al índice Solr construido durante el proyecto Zonales.

3.1. Diseño

Al decidir la utilización de la herramienta Apache Solr[7] para el indexado y la búsqueda de información se tomaron dos decisiones de arquitectura clave que afectaban al resto de los componentes de la solución:

- El diseño del esquema de documentos que se utilizaría para indexar la información.
- La configuración de la instancia Solr y el diseño de las consultas para recuperar la información buscada.

En el proyecto Zonales, el diseño del índice y la configuración de la instancia Solr para almacenar la información se realizó en dos etapas claramente diferenciadas por la arquitectura pensada para la solución en cada una de ellas.

En ambas etapas la unidad de información o documento para Zonales era el post. Un post es un artículo re-publicado en Zonales, proveniente de una red social como Facebook o Twitter, de un medio digital o de cualquier otra fuente que se indexaba y clasificaba para luego mostrarlo "zonificado" y ecualizado[3] según los intereses del usuario en la web. En la segunda etapa se comenzó a considerar los comentarios de los usuarios sobre las publicaciones extraídas (crawleadas) y también se indexaron como posts vinculados con otro post preexistente en el índice.

Indexación de documentos (posts): La extracción de información heterogénea de diversas fuentes y el proceso de normalización y homogenización de la información para ajustarla a una estructura única fue el mayor desafío. El caso más emblemático fue el de la extracción de información a partir de fuentes RSS, donde gran parte de la misma no respetaba los estándares para comentarios, enlaces (links) y contenido multimedia. Para poder lograrlo se desarrolló una herramienta de configuración de selectos CSS que, en conjunto con una serie de funciones de parsing incorporadas al extractor correspondiente, permitieron normalizar esta información. Este esfuerzo de homogenización de la información derivó en la creación de dos formatos normalizados para estructurar las noticias que permitieron contener tanto los posts extraídos de las diversas fuentes como los datos de clasificación asociados. Se decidió utilizar los formatos estándares XML y JSON para contener la estructura de datos diseñada y se la bautizó como XZone y JZone respectivamente.

3.2. Recuperación de posts

Partiendo de la premisa de mostrar al usuario la información ordenada en base a su contexto geográfico y sus preferencias, se diseñó el portal web considerando cuatro aspectos principales:

- **Ecualización**[3]: Se le permitía a los usuarios del portal definir un set de preferencias para el filtrado de la información.
- **Relevancia**: se le permitía a los lectores de zonales elevar o disminuir la relevancia de una noticia desde la propia web. Se utilizó además como parámetros la cantidad de "me gusta", "retweets" y "comentarios" de cada post para calcular la relevancia en base a un fórmula de peso para cada caso.
- **Tipo de fuente**: los lectores del portal pueden seleccionar el tipo de fuente de la cual recuperar las noticias. En principio se dividió la recuperación en dos conceptos: "En la red" para englobar las noticias de las redes sociales Facebook y Twitter, y "Noticias" para la información recuperada desde medios digitales a través de RSS.
- **Temporalidad de la noticia**: considerando que para el común de los los lectores las noticias pierden relevancia con el paso del tiempo este es un aspecto de peso en la recuperación de la información.
- **Zona**: el usuario debía ver principalmente noticias de su zona y aledaños.

Las pruebas narradas en la siguiente sección se realizaron haciendo foco en la temporalidad y la localización de la información, aspectos por los cuales se trabajó con las funciones de boosting de Solr y que son objeto del presente trabajo.

4. Pruebas

La búsqueda de la consulta Solr que permitiera recuperar la información teniendo en cuenta los aspectos de temporalidad y zona se realizó a través de

métodos de prueba sucesivas, refinando las consultas en cada iteración, analizando en cada uno de ellas los resultados obtenidos en comparación con los esperados y volviendo a ajustar los parámetros. Se utilizaron para tal fin boosting querys (bq) y boosting functions (bf), parámetros que permitieron especificar un factor que aumentara o "impulsara" la importancia de determinados campos, valores o combinación de ambos en la consulta.

Debido a la problemática planteada se decidió la utilización de técnicas de TDD (Test Driven Development o Desarrollo Guiado por Pruebas)[1] como opción válida para las pruebas de recuperación y boosting realizadas durante el proceso, brindando una opción repetible que permitiera garantizar el cumplimiento del objetivo de las pruebas a alcanzar para los distintos escenarios y sus variantes. Se diseñaron distintos escenarios de prueba, cada uno de los cuales suponía la ejecución de cambios en el contexto por parte del usuario, planteando los resultados esperados en relación al orden de las noticias, abarcando las combinaciones que se consideraron de mayor importancia.

Es importante considerar el razonamiento seguido para el planteo de los resultados esperados cuando la consulta involucraba estos aspectos. Es decir, cuando un usuario deseaba ver las noticias de su zona ordenadas desde la más reciente a la más antigua.

Partiendo de la suposición de la existencia de un conjunto de datos que recibe noticias extraídas de diversas fuentes en forma constante, tanto para la zona del usuario, por ejemplo su localidad, como para el resto de los elementos de la jerarquía geográfica (barrios, provincia, país, etc.), tanto ascendente como descendente, se consideró que, para un usuario de una determinada localidad, serían de mayor relevancia las noticias de su provincia dentro de las ultimas cuarenta y ocho horas que las de su localidad con mayor antigüedad, por ejemplo de cinco días.

Siguiendo este razonamiento se dividió la temporalidad en cuatro franjas que, combinadas con las jerarquías geográficas, definían el siguiente orden esperado en los resultados:

1. Noticias de la zona seleccionada dentro de las 48 horas
2. Noticias de los hijos de la zona seleccionada dentro de las 48 horas
3. Noticias de los padres de la zona seleccionada dentro de las 48 horas
4. Noticias hermanas de la zona seleccionada dentro de las 48 horas
5. Noticias de la zona seleccionada desde 48 horas hasta 1 semana
6. Noticias de los hijos de la zona seleccionada desde 48 horas hasta 1 semana
7. Noticias de los padres de la zona seleccionada desde 48 horas hasta 1 semana
8. Noticias hermanas de la zona seleccionada desde 48 horas hasta 1 semana
9. Noticias de la zona seleccionada desde 1 semana hasta 1 mes
10. Noticias de los hijos de la zona seleccionada desde 1 semana hasta 1 mes
11. Noticias de los padres de la zona seleccionada desde 1 semana hasta 1 mes
12. Noticias hermanas de la zona seleccionada desde 1 semana hasta 1 mes
13. Resto de las noticias respetando la temporalidad.

4.1. Escenarios

Los distintos escenarios de prueba se construyeron combinando conjuntos de datos y criterios de búsqueda donde para cada combinación se especificaron los resultados esperados.

Para facilitar la creación de los conjuntos de datos, la sistematización de las pruebas y el análisis de lo resultados, se utilizaron valores descriptivos de la noticia dentro de la prueba en el campo *verbatim*. Por ejemplo, para una noticia extraída de Facebook con una antigüedad de tres días, una relevancia con valor 20 para la zona Puerto Madryn en el campo verbatim se grabó la siguiente información: "Noticia 3 días - Puerto Madryn, Chubut, Argentina – Facebook – R20".

El primer conjunto de datos D1 se compone de treinta y cinco noticias, organizadas en sub-conjuntos de cinco (5) noticias de la siguiente forma:

- Sub-conjunto de noticias de barrio Centro, localidad Puerto Madryn, Chubut
- Sub-conjunto de noticias de localidad Puerto Madryn, Chubut
- Sub-conjunto de noticias de la provincia de Chubut
- Sub-conjunto de noticias de otra localidad de la provincia (hermana): Trelew, Chubut
- Sub-conjunto de noticias de una localidad de otra provincia: Rosario, Santa Fe
- Sub-conjunto de noticias de otra provincia: Mendoza
- Sub-conjunto de noticias a nivel país: Argentina

Cada uno de estos sub-conjuntos se compone con noticias de distinto valor temporal:

- Una noticia de hoy
- Una noticia de ayer
- Una noticia con 3 días de antigüedad
- Una noticia con 5 días de antigüedad
- Una noticia con 20 días de antigüedad

Para definir los criterios de búsqueda se abarcaron distintos niveles de jerarquía geográfica, considerando como "Nivel 1" a la zona de mayor escala, es decir Argentina. Se utilizaron los siguientes criterios:

- C1-Recuperar las noticias de Puerto Madryn, Chubut, Argentina – Nivel 3
- C2-Recuperar las noticias del barrio Centro, Puerto Madryn, Chubut, Argentina – Nivel 4
- C3-Recuperar las noticias de Trelew, Chubut, Argentina – Nivel 3
- C4-Recuperar las noticias de la provincia de Chubut, Argentina – Nivel 2

El segundo conjunto de datos D2 contenía cinco noticias adicionales, que tienen por objeto principalmente simular la dinámica de ingreso de nueva información. Es decir, la extracción de noticias en Zonales se realiza en forma constante de acuerdo a la programación del scheduler.

Los datos adicionales agregados son:

- Una noticia de Puerto Madryn, Chubut de ahora (más reciente)
- Una noticia de Puerto Madryn, Chubut de 3 días de antigüedad (más de 48 horas)
- Una noticia de Sarmiento, un barrio de Trelew, Chubut de ahora (nivel 4 en localidad hermana, temporalidad más reciente)
- Una noticia de Comodoro Rivadavia de ahora (localidad hermana, temporalidad más reciente)
- Una noticia de Formosa, Formosa de ahora (localidad de otra provincia, temporalidad más reciente)

La combinación de ambos conjuntos de datos con los criterios definidos dieron origen a un total de ocho escenarios donde reflejamos alguno de ellos en la siguiente tabla:

Escenario	**Resultado**	**Orden esperado**
E1	RE1	Se espera recuperar en primer lugar las noticias en el rango de 48 hs. para Puerto Madryn, luego las del barrio Centro de Puerto Madryn, Chubut, Trelew y finalmente las de Argentina y las de localidades de otras provincias. A continuación las noticias del rango 48hs. a 1 semana y 1 semana a 1 mes, ambos en el mismo orden de zona descripto anteriormente.
E2	RE2	Se espera recuperar en primer lugar las noticias en el rango de 48 hs. para barrio Centro de Puerto Madryn, luego las de Puerto Madryn, Chubut, Trelew y finalmente las de Argentina y las de localidades de otras provincias. A continuación las noticias del rango 48hs. a 1 semana y 1 semana a 1 mes, ambos en el mismo orden de zona descripto anteriormente.
...	...	...
E8	RE8	El orden esperado es el mismo del escenario E4 con las siguientes alteraciones: Dentro del rango de 48 horas se espera recuperar las noticias más recientes de Puerto Madryn, barrio Sarmiento de Trelew y Comodoro Rivadavia a continuación de las noticias de Chubut y la de Formosa al final del rango. La noticia insertada para Puerto Madryn de más de 48 horas debe aparecer a continuación de las noticias de Chubut dentro del rango correspondiente.

Cuadro 1: Resultados esperados de cada escenario

4.2. Sistematización de las pruebas

Las pruebas se realizaron ejecutando las consultas en forma directa sobre la herramienta Solr a través de request HTTP sobre su API pública y solicitando los resultados en formato XML.

Se estructuraron los resultados esperados respetando este formato y en cada ciclo de prueba y refinamiento se automatizó la comparación de resultados utilizando herramientas de testing mediante asertos[1].

5. Resultados

Luego de reiteradas modificaciones y adaptaciones sobre las funciones de boosting (ordenamiento) para lograr obtener los resultados esperados en cada escenario se logró una función estable, probada mediante la regeneración de los datos y la repetición de las consultas. Una vez estabilizada la función objetivo se repitieron las consultas veinte veces, logrando obtener siempre los mismos resultados para cada escenario. La efectividad lograda en función de los resultados obtenidos versus resultados esperados es cercana un 90 %, aunque la misma aumenta hasta un valor cercano al 100 % si se consideran solo la primer franja temporal, es decir las noticias dentro de las 48 horas.

Para la obtención de la función de boosting se creó un servicio javascript que, en función del contexto, construye la URL Solr. En particular la porción de la URL relacionada con las pruebas y las funciones de boosting se construyó utilizando la función getSolrBoosting(zCtx) que se explica a continuación. Todas las dimensiones definidas fueron divididas en escala en función de la cantidad de niveles jerárquicos de la zona geográfica. Es decir, Nivel 1: Argentina, Nivel 2: Provincias, Nivel 3: localidades, Nivel 4: barrios, y así sucesivamente. El campo selZone del contexto que contiene la zona en formato extendido permite calcular este nivel. Para el peso de las zonas en las consultas se utilizó boosting querys (bq) construidas alternando los campos zonePartialExtendedString y zoneExtendedString con el siguiente criterio de boosting:

- zonePartialExtendedString = (zona de nivel 1)$\hat{1}0^3$
- zoneExtendedString = (zona de nivel 1)$\hat{1}0^4$
- zonePartialExtendedString = (zona de nivel 2)$\hat{1}0^5$
- zoneExtendedString = (zona de nivel 2)$\hat{1}0^6$
- ...
- zonePartialExtendedString = (zona de nivel n)$\hat{1}0^{n+3}$
- zoneExtendedString=(zona de nivel n)$\hat{1}0^{n+4}$

Esta distribución de peso permite dar mayor importancia a las zonas más cercanas a las del nivel buscado en una relación 10^n. Las franjas temporales cobran una relevancia mayor que la zona en relación directa al nivel. Por ejemplo, en una búsqueda de zona nivel 4 el peso de la franja temporal debe ser mucho mayor que para un búsqueda de una zona nivel 2. Para el peso de las franjas temporales también se utilizó boosting querys (bq) construidas en base a campo modified de la noticia y la cantidad de niveles utilizando el siguiente criterio:

- modified:[NOW-48HOURS TO *$\hat{1}0^{\#nivel*4}$
- modified:[NOW-7DAYS TO NOW-48HOURS]$\hat{1}0^{\#nivel*(4-5)}$
- modified:[NOW-30DAYS TO NOW-7DAYS]$\hat{1}0^{\#nivel*(4-10)}$

El factor de corrimiento utilizado en el multiplicador del exponente, definido en una escala 1, 5, 10, es el que permitió aumentar o disminuir el peso de la franja temporal en función de los niveles de la zona. Otro requisito utilizado en las consulta fue recuperar en orden de fecha las noticias dentro de una misma

franja temporal. Para lograrlo se utilizó una boosting function aplicada sobre el campo modified y utilizando el nivel para el factor de boosting de la siguiente manera: bf=ord(modified)$\hat{1}0^{\#nivel*2}$. La consultas obtenidas se construyen en función de los criterios. A continuación se detalla la query resultante para cada uno de ellos.

5.1. Ejemplos de algunos criterios

Criterio 2, Escenarios 2 y 6 *Centro, Puerto Madryn, Chubut, Argentina*

```
/solr/select?indent=on&version=2.2&start=0&fl=*%2Cscore&rows=40&qt=zonalesConten
&sort=&wt=xml&explainOther=&hl.fl=&q=docType:post&bf=ord(modified)^100000000
&bq=+
        zoneExtendedString:"Centro,+ Puerto+Madryn,+Chubut,+Argentina"^10000000000
        zonePartialExtendedString:"Centro,+ Puerto+Madryn,+ Chubut,+Argentina"^100
        zoneExtendedString:"Puerto+Madryn,+ Chubut,+Argentina"^100000000+
         zonePartialExtendedString:"Puerto+Madryn,+ Chubut,+Argentina"^10000000+
         zoneExtendedString:"Chubut,+ Argentina"^1000000+
        zonePartialExtendedString:"Chubut,+Argentina"^100000+
        zoneExtendedString:"Argentina"^10000+
        zonePartialExtendedString:"Argentina"^1000+
        modified:[NOW-48HOURS+TO+*]^10000000000000000+
        modified:[NOW-7DAYS+TO+NOW-48HOURS]^100000000000+
        modified:[NOW-30DAYS+TO+NOW-7DAYS]^1000000
```

Criterio 4, Escenarios 4 y 8 *Chubut, Argentina*

```
/solr/select?indent=on&version=2.2&start=0&fl=*%2Cscore&rows=40&qt=zonalesConten
    zoneExtendedString:"Chubut,+Argentina"^1000000+
    zonePartialExtendedString:"Chubut,+Argentina"^100000+
    zoneExtendedString:"Argentina"^10000+
    zonePartialExtendedString:"Argentina"^1000+
    modified:[NOW-48HOURS+TO+*]^100000000+
    modified:[NOW-7DAYS+TO+NOW-48HOURS]^1000+
    modified:[NOW-30DAYS+TO+NOW-7DAYS]^0.01
```

No se hicieron pruebas de estrés sobre este conjunto de datos pues las pruebas de escalabilidad se realizaron en el trabajo ya mencionado "Distributed Search on Large NoSQL Databases" [6].

6. Conclusiones y Trabajo Futuro

Las funciones y consultas de boosting utilizadas en el presente trabajo son solo una pequeña parte de un gran conjunto de herramientas que ofrece Apache Solr para la recuperación ordenada de información. El campo de Big Data es tan amplio que incluso muchos expertos consideran a esta época como "la era del

Big Data" y en este contexto dominar una herramienta como Solr que permita indexar, clasificar y recuperar en forma ordenada gran cantidad de información valiosa es un punto fuerte de la herramienta. El resultado principal de este trabajo consistió en poder implementar los requerimientos propuestos en un entorno de producción real, con datos reales de la empresa Mediabit, para su proyecto Zonales. Si bien la aproximación ha sido empírica a las funciones de boosting utilizadas, las mismas han cumplido satisfactoriamente las premisas requeridas.

Como trabajo futuro podemos mencionar que las nuevas versiones de Apache Solr ofrecen la posibilidad de indexar datos geográficos asociados a la información e incorporar nuevas herramientas de búsquedas para recuperar dicha información por conceptos tales como distancias, pertenencia a una polígono geográfico, etc. Esto permitiría mejorar la solución Zonales a tal punto que un usuario podría recuperar las noticias ordenadas por cercanía geográfica a partir del punto donde se encuentre con niveles de precisión de metros.

En [6] se indexaron artículos de wikipedia utilizando como criterio de particionamiento horizontal de los datos la paridad del id numérico de los documentos. Repetir la experiencia utilizando la información de Zonales y tomando como método de partición de los datos el campo de zona geográfica en su formato semántico es otro desafío de interés y permitiría incluso probar la escalabilidad utilizando las funciones de boosting del presente trabajo. Otro desafío es mejorar el modelo matemático para refinar los resultados del presente trabajo. Se considera que las relaciones de pesos encontradas tanto para la zona como para la temporalidad pueden mejorarse a través de este tipo de modelos.

Referencias

1. Kent Beck. *Test-driven development: by example.* Addison-Wesley Professional, 2003.
2. Henderson Cal. *Building Scalable Web Sites.* 2006.
3. Barry Damián, Juan Manuel Cortez, and Francisco Páez. Construcción de un ecualizador de interés mediante el uso de lucene-solr. *IEEE Intercon*, 2010.
4. Barry Damián, Aita Luis Ignacio, and Cortez Juan Manuel. Zcrawler: Extracción, clasificación y publicación de información pública desde su perspectiva geográfica. *JAIIO 2014.*
5. Hatcher Erik and Otis Gospodnetić. *Lucene in Action.* Manning Publications Co., 2nd. ed edition, 2004.
6. Tinetti Fernando, Damián Barry, Ignacio Aita, and Franciasco Páez. Distributed search on large nosql databases. *PDPTA2011*, 2011.
7. Apache Fundation. Apache solr reference guide.

Una herramienta para la evaluación y comparación de metodologías de minería de datos

Juan Miguel Moine[1], Ana Silvia Haedo[2]

1. Universidad Tecnológica Nacional, Facultad Regional Rosario
2. Facultad de Ciencias Exactas, Universidad Nacional de Buenos Aires

[1]juanmiguelmoine@gmail.com, [2]ahaedo@dc.uba.ar

Resumen. Para llevar a cabo en forma sistemática el proceso de descubrimiento de conocimiento en bases de datos, conocido como minería de datos, es necesaria la implementación de una metodología. Actualmente las metodologías en éste área se encuentran en etapas tempranas de madurez, aunque algunas como CRISP-DM ya están siendo utilizadas exitosamente por los equipos de trabajo para la gestión de sus proyectos. En el presente trabajo se propone un marco comparativo como herramienta para la evaluación y confrontación de metodologías de minería de datos. A partir del mismo se han comparado las metodologías más difundidas en la actualidad.

Palabras clave: Minería de Datos, Gestión de Proyectos, Knowledge Discovery in Databases, Explotación de Información, Metodologías de Minería de Datos, CRISP-DM, Catalyst, P3TQ, KDD.

1 Introducción

Los esfuerzos en el área de la minería de datos o explotación de información se han centrado en su gran mayoría en la investigación de técnicas para la explotación de información y extracción de patrones (tales como árboles de decisión, análisis de conglomerados y reglas de asociación). Sin embargo, se ha profundizado en menor medida el hecho de cómo ejecutar este proceso hasta obtener el "nuevo conocimiento", es decir, en las metodologías.

Una metodología permite llevar a cabo el proceso de minería de datos en forma sistemática y no trivial. Ayuda a las organizaciones a entender el proceso de descubrimiento de conocimiento y provee una guía para la planificación y ejecución de los proyectos. Define además de las fases del proceso, las tareas específicas que deben realizarse y cómo llevarlas a cabo.

En los inicios del año 1996, KDD (Knowledge Discovery in Databases) [1] constituyó el primer modelo aceptado en la comunidad científica que estableció las etapas principales de un proyecto de explotación de información.

A partir del año 2000, con el gran crecimiento en el área de la minería de datos, surgen tres nuevos modelos que plantean un enfoque sistemático para llevar a cabo el proceso: SEMMA [2], CRISP-DM [3] y Catalyst (conocida como P3TQ) [4].

SEMMA fue desarrollada por el SAS Institute y su nombre es el acrónimo correspondiente a las cinco fases básicas del proceso: Sample (Muestreo), Explore (Exploración), Modify (Modificación), Model (Modelado), Assess (Evaluación).

CRISP–DM, creada por el grupo de empresas SPSS, NCR y Daimer Chrysler en el año 2000, estructura el proceso en seis fases: Comprensión del negocio, Comprensión de los datos, Preparación de los datos, Modelado, Evaluación e Implementación. Cada fase se descompone en varias tareas generales de segundo nivel.

La metodología Catalyst, conocida como P3TQ (Product, Place, Price, Time, Quantity), fue propuesta por Dorian Pyle en el año 2003. Esta metodología plantea la formulación de dos modelos: el Modelo de Negocio y el Modelo de Explotación de Información. El Modelo de Negocio, proporciona una guía de pasos para identificar un problema (o la oportunidad del mismo) y los requerimientos reales de la organización. El Modelo de Explotación de Información, proporciona una guía de pasos para la construcción y ejecución de modelos de minería de datos a partir del Modelo de Negocio.

Según [5] KDD y SEMMA no llegan a ser una metodología propiamente dicha, ya que dejan a criterio del equipo de trabajo la definición de las actividades específicas a realizar en cada etapa del proyecto. En cambio, CRISP-DM y Catalyst podrían ser considerados una metodología, porque además de especificar detalladamente las tareas en cada fase proporcionan guías sobre cómo ejecutarlas.

En la actualidad son escasos los estudios que comparan las metodologías mencionadas, los cuales están enfocados en establecer un paralelismo entre las fases del proceso [6, 7, 8] y no un análisis comparativo confrontando las características de cada una.

A medida que la minería de datos madure con el tiempo, irán surgiendo nuevas metodologías y cada vez serán más las alternativas que estarán a disposición del equipo de trabajo. Surge entonces la necesidad de contar con una herramienta que permita evaluar y confrontar diferentes metodologías para esta emergente disciplina, es decir, un marco comparativo.

Objetivos. El objetivo principal de este trabajo es la construcción de un marco comparativo que sirva como herramienta para poder evaluar y confrontar diferentes metodologías de minería de datos. Como objetivo secundario se pretende utilizar esta herramienta para evaluar y comprar CRISP-DM y Catalyst, las cuales constituyen las dos metodologías más difundidas en la actualidad para la gestión de proyectos de minería de datos.

2 Metodología

El marco comparativo que se propone en este trabajo está formado por cuatro aspectos (Fig. 1):

- Nivel de detalle en la descripción de las actividades de cada fase
- Escenarios de aplicación
- Actividades específicas de cada fase
- Actividades destinadas a la dirección del proyecto.

Fig. 1. Aspectos del marco comparativo

Para cada uno de estos aspectos se propone la evaluación de una serie de características, las cuales deberían estar presentes en una metodología de minería de datos bien definida.

Entre los cuatro aspectos se evalúan un total de 52 características, que se describen y explican en forma completa en [9].

Aspecto 1: Nivel de detalle en la descripción de las actividades de cada fase

Una metodología está formada por un conjunto de actividades, las cuales por lo general son agrupadas en un mayor nivel de abstracción denominado fase. La cantidad de niveles de abstracción puede variar entre una metodología y otra. Lo importante es que para cada fase, se definan un conjunto de actividades específicas que secuencien el trabajo a realizar. En una metodología completa, se espera que no sólo se describan las actividades, sino también se especifique la forma en la que deben llevarse a cabo. El resultado tangible de una actividad se denomina entregable.

Este aspecto del marco comparativo está compuesto por 5 características [9] enumeradas en la Tabla 1, y su objetivo es evaluar el nivel de detalle con el que una metodología define las actividades que conforman al proceso.

Tabla 1. Características del primer aspecto del marco comparativo

Características a evaluar
• ¿Se definen actividades específicas para cada fase del proceso? • ¿Se explicitan los pasos a seguir para llevar a cabo cada actividad? • ¿Se definen las entradas de cada actividad? • ¿Se definen las salidas o entregables de cada actividad? • ¿Se provee una guía de buenas prácticas para cada una de las actividades específicas?

Aspecto 2: Escenarios de aplicación.

Los proyectos de explotación de información pueden ser llevados a cabo en distintos escenarios. En algunos casos el usuario desea obtener nuevo conocimiento para abordar algún problema/situación, y en otros se encuentra interesado en explorar sus datos transaccionales en busca de relaciones o patrones que puedan serle útiles. Sin

embargo, en este último caso también existe una situación de trasfondo que motiva el proyecto, ya que difícilmente una organización lo financie si no se establece los beneficios que producirá.

Este aspecto se evalúa en función de 4 características [9] enumerados en la Tabla 2.

Tabla 2. Características del segundo aspecto del marco comparativo

Características a evaluar
• ¿Se especifican actividades para la definición y el análisis del problema u oportunidad con el cual colaborará la minería de datos? • ¿Se consideran puntos de partida alternativos donde el usuario no refiere un problema sino que sólo desea explorar sus datos? • ¿La metodología es independiente del dominio de aplicación? • ¿La metodología es aplicable a proyectos de diferente tamaño?

Aspecto 3: Actividades específicas que componen cada fase.

En este aspecto se pretende analizar la incorporación de ciertas actividades relevantes que deberían estar presentes a lo largo del proceso de minería de datos. Para ello, se propone la evaluación de una serie de características en función de las distintas fases generales que componen al proceso: análisis del problema, selección y preparación de los datos, modelado, implementación y evaluación.

Este aspecto se evalúa en función de 26 características [9], las cuales se agrupan según las diferentes fases del proceso de extracción de conocimiento. Algunas de ellas se mencionan en la Tabla 3.

Tabla 3. Características del tercer aspecto del marco comparativo

Fase	Características a evaluar
Fase de análisis del problema	• ¿Se propone una evaluación general de la organización? • ¿Se identifica al personal involucrado en el proyecto (stackeholders)? • ¿Se especifica de qué forma el usuario utilizará el nuevo conocimiento?
Fase Selección y Preparación de los datos	• ¿Se propone un análisis exploratorio inicial de los datos? • ¿Se contemplan actividades para la transformación de variables y la creación de atributos derivados? • ¿Se verifica con el usuario la completitud del conjunto de datos final?
Fase de Modelado	• ¿Se planifica de qué forma se evaluarán los resultados? • ¿Se proveen directivas para el caso donde se dificulta el descubrimiento de los patrones?
Fase de Evaluación	• ¿Se interpretan los modelos en función de los objetivos organizacionales? • ¿Se comparan y ponderan los modelos obtenidos? • ¿Se propone una revisión general del proceso?
Fase de Implementación	• ¿Se planifica la implementación del nuevo conocimiento? • ¿Se propone la creación de un programa de mantenimiento? • ¿Se documenta la experiencia adquirida por el equipo de trabajo?

Aspecto 4: Actividades destinadas a la dirección del proyecto.

Las actividades de dirección del proyecto consisten en la planificación, ejecución y control de ciertos aspectos importantes para que el mismo se desarrolle exitosamente. Entre estos aspectos se encuentran, por ejemplo, la administración del costo (presupuesto) y la del tiempo del proyecto (cronograma).

Este tipo de actividades se clasifica en dos grupos: actividades de planificación y actividades de control. Las actividades de planificación incluyen la identificación de las tareas a realizar en el proyecto, estimación de la duración de las mismas, estimación de los recursos afectados y la definición del curso de acción. Las actividades de control tienen por objetivo el monitoreo del estado actual del proyecto para su comparación con lo planificado.

Existen ciertos estándares reconocidos internacionalmente, como el PMBOK [10], que establecen cuáles son las áreas que deben administrarse para lograr una adecuada gestión del proyecto, independientemente de su tipo.

Tomando como referencia el PMBOK, en este aspecto del marco comparativo se definen cinco áreas de dirección del proyecto, para cada una de las cuales se evaluarán características relacionadas a la planificación y control de las mismas. Las áreas definidas son: gestión del alcance, gestión del tiempo, gestión del costo, gestión del equipo de trabajo y gestión del riesgo.

Este aspecto se evalúa en función de 17 características [9], las cuales se agrupan según el área de dirección del proyecto. Algunas de ellas se enumeran en la Tabla 4.

Tabla 4. Características del cuarto aspecto del marco comparativo

Área	Características a evaluar
Gestión del alcance	• ¿Se propone la selección de los entregables que se generarán durante el proyecto? • ¿Se especifican actividades de control del alcance?
Gestión del tiempo	• ¿Se construye un cronograma para el proyecto? • ¿Existen actividades de control del cronograma?
Gestión del costo	• ¿Se construye un presupuesto de costos? • ¿Existen actividades de control del presupuesto a medida que avanza el proyecto?
Gestión del equipo de trabajo	• ¿Se efectúa una planificación de los recursos humanos? • ¿Se proponen actividades para motivar la interacción entre los miembros del equipo? • ¿Se efectúa un seguimiento del rendimiento de los recursos humanos?
Gestión del riesgo	• ¿Se efectúa una identificación de los riesgos del proyecto? • ¿Se realiza una cuantificación y priorización de los riesgos? • ¿Existen actividades de supervisión y control de los riesgos?

Utilización del marco comparativo

Como se ejemplifica en la Tabla 5, las características podrán ser evaluadas positiva o negativamente dependiendo si las metodologías en estudio cumplen o no con las mismas, obteniendo como resultado final el porcentaje de valoraciones positivas de cada enfoque.

Tabla 5. Ejemplo de comparación de 3 metodologías.

Característica	Metodología 1	Metodología 2	Metodología 3
Nivel de detalle en actividades de cada fase			
Característica 1.1	NO	NO	SI
Característica …	SI	NO	SI
Escenarios y puntos de partida del proyecto	..	..	..
…	..	..	..
Total de características cumplidas	40 de 52 (77%)	35 de 52 (67%)	51 de 52 (98%)

No se ha realizado una ponderación de cada ítem debido a la subjetividad que podría implicar dicha tarea, quedando esta decisión a criterio del usuario de este marco comparativo. Si se utilizara un sistema de puntajes los resultados probablemente sean diferentes (ya que la presencia de cada característica tendrá su propio peso).

3 Resultados

El marco comparativo se utilizó para evaluar y confrontar a CRISP-DM y a Catalyst [9]. Previamente, para conocer en profundidad ambas metodologías, se aplicaron a un caso de estudio de un centro médico [9] donde se abordó el problema de ausentismo en atenciones médicas programadas (turnos).

La Tabla 6 resume los resultados obtenidos, donde para cada aspecto se han contabilizado las características positivas sobre el total de características evaluadas.

Tabla 6. Evaluación final de todos los aspectos del marco comparativo

Aspecto	CRISP-DM	Catalyst
Nivel de detalle en la descripción de las actividades.	4/5 (80%)	4/5 (80%)
Escenarios de aplicación	3/4 (75%)	4/4 (100%)
Actividades específicas de cada fase	20/26 (77%)	21/26 (81%)
Actividades de dirección del proyecto	8/17 (47%)	8/17 (47%)
Total de características cumplidas	**35/52 (67%)**	**37/52 (71%)**

La evaluación final demuestra que se ha obtenido un resultado muy parejo. La metodología Catalyst logró cumplir con 37 de las 52 características que componen el marco (es decir, el 71%). CRISP-DM ha obtenido un resultado levemente inferior cumpliendo el 67% de los puntos evaluados.

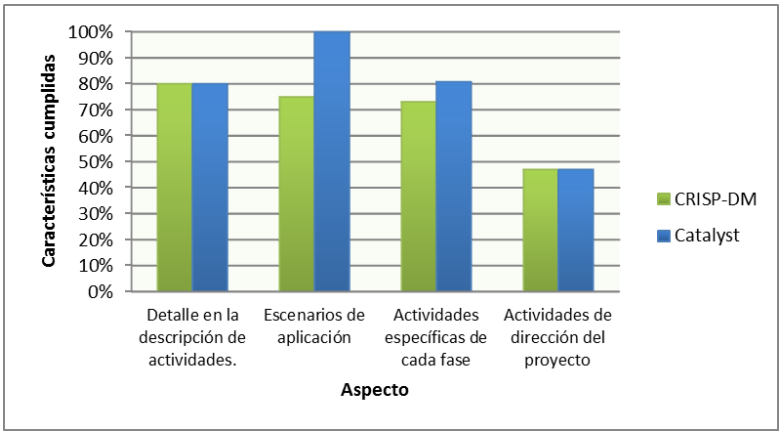


Fig. 2. Porcentaje de características presentes en cada metodología

Como se puede observar en la Figura 2, ambas metodologías resultaron con un puntaje similar en la mayoría de los aspectos, salvo en los escenarios de aplicación donde Catalyst ha logrado cumplir el 100% de las características evaluadas. Se puede apreciar también que ninguna de las dos metodologías tuvo un buen desempeño en las actividades de dirección del proyecto, ya que ambas han obtenido un puntaje inferior al 50%, evidenciando la falta de madurez en este aspecto.

A partir de este análisis, se observa que CRISP-DM y Catalyst reúnen un buen porcentaje de las características propuestas en este marco comparativo, aunque ambos enfoques deberían complementarse con actividades destinadas a la dirección del proyecto.

4 Conclusión

En este trabajo se ha propuesto un marco comparativo como herramienta para la confrontación de metodologías de minería de datos.

El marco presentado incluye cuatro aspectos donde se analiza el nivel de especificación de las tareas, los escenarios de aplicación, las actividades que componen cada fase del proceso y la incorporación de actividades para la dirección del proyecto. Para cada uno de estos aspectos se propone la evaluación de un conjunto de características que deberían estar presentes en una metodología de minería de datos bien definida.

La herramienta se ha utilizado para confrontar las metodologías CRISP-DM y Catalyst. Ambos enfoques han logrado cumplir un gran porcentaje de las

características evaluadas, aunque durante el estudio también se han evidenciado los puntos que se deberían mejorar y seguir desarrollando, como las actividades destinadas a la dirección del proyecto.

Referencias

1. Fayyad, U. y otros: The KDD process for extracting useful knowledge from volumes of data. ACM vol. 39 (1996)
2. SAS Institute: Data Mining and the Case for Sampling. http://nas.uhcl.edu/boetticher/ML_DataMining/SAS-SEMMA.pdf. (1998) Recuperado el 5 de mayo de 2015
3. Chapman, P., Clinton, J., Kerber, y otros: CRISP-DM 1.0 Step-by-step data mining guide (2000)
4. Pyle, D.: Business Modeling and Data Mining. Morgan Kaufmann Publishers (2003)
5. Moine, J. M., Gordillo, S., Haedo, A.: Análisis comparativo de metodologías para la gestión de proyectos de Minería de Datos. CACIC 2011, VIII Workshop Bases de Datos y Minería de Datos (2011)
6. Mariscal, G. y otros: A survey of data mining and knowledge discovery process models and methodologies. The Knowledge Engineering Review, Vol. 25:2, 137–166 (2010)
7. Kurgan, L. A., Musilek, P.: A survey of Knowledge Discovery and Data Mining process models. Knowledge Engineering Review, 21(1), 1-24 (2006)
8. Azevedo, A.: KDD, SEMMA AND CRISP-DM a parallel overview. AIDIS (2008)
9. Moine, J. M., Gordillo, S., Haedo, A.: Metodologías para el descubrimiento de conocimiento en bases de datos: un estudio comparativo. Tesis de Magíster. Universidad Nacional de La Plata, Argentina (2013)
10. Project Management Institute: Guía de los fundamentos de la dirección de proyectos (Guía del PMBOK®). Tercera edición (2005)

Inverted Index Entry Invalidation Strategy for Real Time Search

Esteban A. Ríssola and Gabriel H. Tolosa

Departamento de Ciencias Básicas
Universidad Nacional de Luján
{earissola, tolosoft}@unlu.edu.ar

Abstract. The impressive rise of user-generated content on the web in the hands of sites like Twitter imposes new challenges to search systems. The concept of real-time search emerges, increasing the role that efficient indexing and retrieval algorithms play in this scenario. Thousands of new updates need to be processed in the very moment they are generated and users expect content to be "searchable" within seconds. This lead to the develop of efficient data structures and algorithms that may face this challenge efficiently. In this work, we introduce the concept of *index entry invalidator*, a strategy responsible for keeping track of the evolution of the underlying vocabulary and selectively *invalidate* and *evict* those inverted index entries that do not considerably degrade retrieval effectiveness. Consequently, the index becomes smaller and may increase overall efficiency. We study the dynamics of the vocabulary using a real dataset and also provide an evaluation of the proposed strategy using a search engine specifically designed for real-time indexing and search.

1 Introduction

The impressive rise of social media and other forms of user-generated content during last decade in the hands of sites like Twitter or Facebook [12, 14, 8, 10, 23, 20] reveals us the compelling challenge that traditional search must face. This growth is not only defined by the number of users who consumes these services but also by the vast amount of content they produce[1]. The implications that the concept of *real-time* introduces give us a hint about the significant role that efficient indexing and retrieval algorithms plays in this scenario.

Search and retrieval over this extensive collections, as well as the management of the involved data structures present certain differences and introduce new requirements in comparison to classical Web search operations [6, 3]. On the one hand, because both queries and user behaviour differ from traditional [23, 13]. On the other hand, because real-time search service results indeed to be very challenging in large-scale microblogging systems where thousands of new updates need to be processed in the very moment they are generated. Indexing can not be considered as a batch operation any more as users expect content to be available

[1] According to [20], an average of eight tweets are issued per second.

(*searchable*) within seconds. Thereby, the indexer should be designed to receive a continuous stream of documents (at very high arrival rates, often with sudden spikes) and to achieve both low latency and high throughput. In addition, as documents are ingested the content of the corresponding structures need to be updated to allow the index to serve incoming queries. This implies the existence of concurrent operations that should be handled carefully. Finally, the nature of real-time search means that temporal stamps are important for ranking, beyond the application of other metrics aimed to improve the quality of the result list.

Our particular focus is over microblog services, like Twitter, where users are able to write brief status messages called *posts*[2] that can share with their network of friends and, often, with the general public at the very moment they are generated. Increasingly, this kind of services grows in popularity and therefore, the data volume they have to deal with becomes larger every day. As far as we know, the only practical strategy to cope with the performance requirements cited above consists in maintaining the inverted index and its corresponding structures in main memory [6, 3]. This strategy primarily admit to significantly reduce reading and writing latencies as compared to other devices, such as disks. Nevertheless, memory remains today a scare resource [6] such that becomes essential to ascertain the way to store the index only the necessary information to provide reasonable (or acceptable) effectiveness.

Thus, bearing in mind the context and its inherent requirements we propose the development of a component called *index entries invalidator*, responsible for keeping track of the evolution that the underlying vocabulary presents. It aims to selectively *invalidate* and *evict* those inverted index's entries whose absence won't considerably degrade retrieval effectiveness. Consequently, the index becomes smaller and may increase overall efficiency. In order to design an efficient invalidation algorithm we conduct the analysis of a real sample of Twitter data to understand its growth dynamic. Specifically, we employ the Tweets2011 [17] dataset widely used by scientific community in this field, composed roughly of 16 million tweets crawled during 2011. Moreover, we select 1 million queries from the well-known AOL Query Log[3] [21]. The experiments examine the performance of the retrieval process and the effect on the quality of the results.

Our contributions are as follows: (a) We look into the vocabulary obtained from our tweets dataset and study its underlying dynamics. Furthermore, we identify tree types of tokens and show that the size of the resulting vocabulary can not be fitted by the Heaps' law [16], as traditional collections; (b) We design and build up an *index entries invalidator* inspired by the concepts of cache invalidators [5], based on the time that the entries have persisted in the index without been updated (time-to-live approach [7]); (c) We perform the corresponding evaluations making use of a self-modified version of Zambezi[4] [1] in-memory search engine for streaming documents by implementing the proposed invalidator. We measure both wall-clock time and effectiveness metrics on a per-query basis.

[2] Throughout this work the terms post, document or tweet are used indistinctly.
[3] Available at: `http://imdc.datcat.org/collection/1-003M-5`
[4] Homepage: `http://nasadi.github.io/Zambezi/`

The remainder of the work is organized as follows: Section 2 provides background concepts on real-time index features. Furthermore, we review related work that examine both efficiency and effectiveness; Section 3 describes the employed collection and introduces vocabulary dynamics analysis; Section 4 presents the index entries invalidator approach; Section 5 details the methodology applied during experiments, along with the employed metrics. Also the obtained results are displayed. Section 6 concludes and introduces future work.

2 Background and Related Work

Information retrieval systems rely on efficient data structures to support search, the so-called inverted index [25]. Basically, it stores the set of all unique terms in the document collection (*vocabulary*) associated to a set of entries that form a posting list. Each entry represents the occurrence of a term t within a document d and it consists of a document identifier (DocID) and a payload that is used to store information about the occurrence of t within d. Each posting list is sorted in an order that depends on the specific query resolution strategy [4, 24, 25]. One of the key features of real-time search resides in the fact that new contents should be available for search immediately after their creation, while concurrently supporting low-latency, high throughput query evaluation. This implies that the index needs to be update incrementally as new documents arrive to the system. For this reason, the indexing process requires allocating space for postings in a dynamic fashion, that results in non-contiguous postings lists [2].

Nowadays, Twitter's Earlybird retrieval engine [6], built upon this large scale microblogging service specific needs, represents a point in the space of real-time search engines. According to its design guidelines, a general solution to the problem of dynamically allocating postings for real-time search is proposed in [3]. As Earlybird represents a particular instantiation, they provide a general framework for incremental indexing where all data structures are stored completely in memory. Thus, from a small number of memory pools, increasingly larger slices for postings are allocate as more term occurrences are encountered. This solution is planned not only for indexing tweets but also it is aligned to applications that have really tight index latency requirements.

Likewise, several approaches have been proposed in the literature to improve indexing and ranking phases, in terms of efficiency and effectiveness. Chen et al. [8] introduced an adaptive indexing scheme aimed at reducing the update cost by delaying indexing less useful tweets (*i.e.*, tweets that may not appear in the search results). Otherwise, they are grouped with other unimportant tweets and indexed later as part of an offline batch process. In [11] an online topic modeling framework for querying large microblog corpus is presented. Such models were employed to identify topics in the tweets and compare them with the ones obtained from the incoming queries. Furthermore, discovered topics are applied to rank relevant tweets in the collection. This approach is called *online* in the sense that corresponding topic modeling was not only conducted over hourly batches

of captured tweets in an offline fashion, but also for recent time intervals that has not yet been included in the last batch.

Moreover, [9, 19] have also proposed strategies to improve the overall effectiveness of microblog retrieval systems. In the former, Choi & Croft suggested to extend a previously defined time-based model for pseudo-relevance feedback query expansion, by incorporating the temporal factor into ranking. In particular, they claim that selecting relevant time period for a specific query based on a user behaviour that can be collected easily, like retweeting, and extracting expanded terms by using weights derived from the relevant time can improve retrieval performance. On the other hand, Metzler et al. define a search task called event retrieval, *i.e.*, given a query that describes a particular event the intention is to retrieve a ranked list of structured event representations. These correspond to a series of timespans during which an instance of the event occurred. An unsupervised methodology is proposed to extract high quality event representations by applying a temporal query expansion technique.

Most recently, Nepomnyachiy et al. [20] introduce a search framework for geo-temporally tagged data to support low-latency retrieval for queries with spatial, temporal, and textual components. Mainly, they define an efficient way to organize index content based on the spatial distribution of user-generated data and considering documents timestamp.

These works tackle different ways to organize index structures in order to boost retrieval performance for real-time search systems. However, they do not consider the possibility of invalidate entries based on the idea of terms' update frequency reducing the index resulting size, thus allowing a more efficient utilization of available computing resources.

3 Vocabulary Dynamics

Data Characterization: Tweets2011 [17] is constituted as the reference collection of the TREC-2012 Microblog Track [22]. It comprises of approximately 16 million tweets crawled between January 23^{rd} and February 8^{th}, 2011. This dataset is designed to be a reusable, representative sample of the twittersphere and it is employed in various works [9, 3, 1]. Due to its sampling methodology, the corpus tends to degrade over time so that 11,601,066 tweets were downloaded successfully. Additionally, all non-English tweets were filtered out. The tweets distribution over time is shown in Figure 1, on average $9,352$ posts arrive per hour. Each post is composed of roughly 13.39 useful words and 81.25 characters. As stated in [20], the number of words in a tweet is considerably small and tokens rarely repeat within a document.

Dynamics: Taking into account the context of the proposed analysis, we decided to split each post considering three types of tokens, namely: mentions[5],

[5] Tokens that are preceded by '@' symbol are used to refer to user's alias.

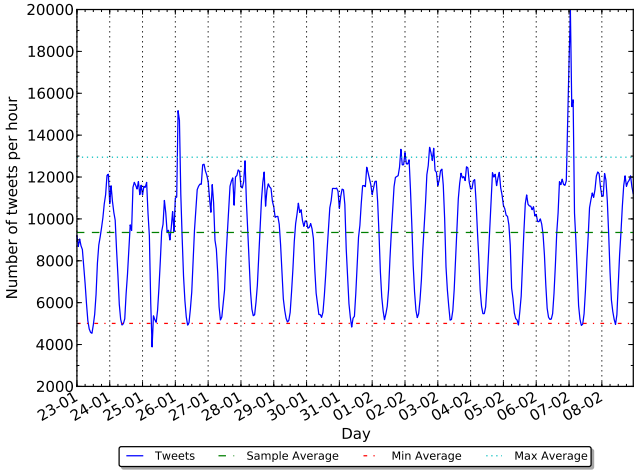


Fig. 1: Tweets arrival rate

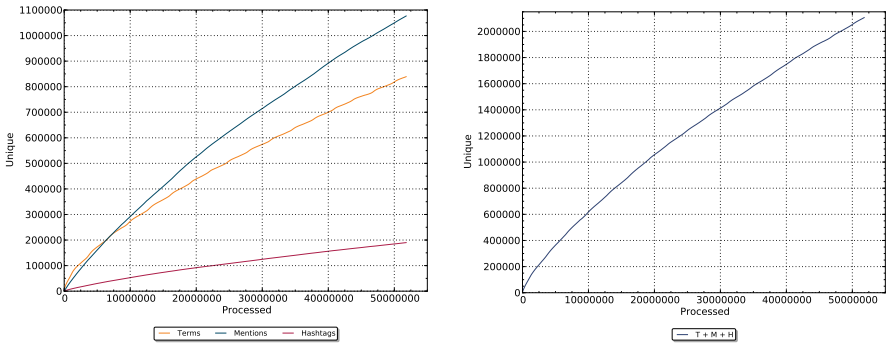
hashtags[6], and general terms. The reason for applying this convention lies behind the fact that both mentions and hashtags have a particular meaning and value inside Twitter [10, 23]. According to traditional IR literature, a practical way to describe how vocabulary and collection size are related corresponds to Heaps' law [16]. In particular, it enables to empirically estimate vocabulary size (and its growth) as a function of the collection size. Figure 2(a) shows that the considered tokens growth faster than this law predictions, and they rather exhibit a linear growth. Similarly, without applying a per token distinction (Figure 2(b)). Despite the number of new encountered tokens raises considerably fast, mostly of those tokens are *hapax*[7], *i.e.*, they appear in only a single tweets. They represent roughly 67.7% of the whole vocabulary.

As various works [9, 19, 20] state, this phenomenon is due to the informal essence that distinguish microblogging activity along with its character limit (140 in Twitter's case). Therefore, abbreviations, elongated words, compound words hashtags, internet slangs and misspellings are common, in many cases deliberate.

In order to proceed with vocabulary dynamics characterization, we classify tokens in three groups by studying how its frequency evolves across the days. To this extent, we apply a *sliding window* approach of s slots. Each slot corresponds to the observation of a token on daily a basis. We set a boolean true value if a token is present in at least one the processed documents of the corresponding sample day. Every day, tokens' frequency information is updated according to

[6] Tokens that are preceded by '#' symbol represent a kind of keyword, and usually adopt the *camel case* convention.

[7] A *hapax legomenon*, or just hapax, is a word that occurs only once within a context.



(a) Per considered token type

(b) Whole vocabulary (Without distinction of token type)

Fig. 2: Vocabulary Growth. The X-axis corresponds to the number of processed tokens. The Y-axis shows unique tokens (That compose the vocabulary).

the ingested tweets. When a token appears for the first time (*i.e.*, it does not exist in the current vocabulary) a new instance of the window is assigned to it and the first slot activates. Thus, as days move forward the window goes over the remaining slots and these may activate (or not) following token's daily frequency behaviour. When the window walks through s slots, a side shift is applied maintaining the last observations. Then, we classify each token according to the following criteria:

Let w_i be the window of s slots that corresponds to token t_i, and w_{ij} be its value at slot j. Let $S_i = \sum_{j=1}^{s} w_{ij}$ and $G(t_i)$ a function that assigns a category (or group) to each token according its occurrence behaviour, defined as:

$$G(t_i) = \begin{cases} stable, & \text{if } S_i = s \\ volatile, & \text{if } 2 \leq S_i \geq (s-1) \\ singular, & \text{if } S_i = 1 \end{cases}$$

In our study we apply a window with $s = 7$ (a whole week) that we consider as a reasonable number to study how tokens behave, giving the dynamic of Twitter. Figure 3 shows the distribution of the classified tokens across the sample. Note that we are able to start with this labeling on January 28^{th}, because until then the window has not reached the 7^{th} slot. Even though we apply a window of seven slots, a deeper study to determine the impact of the window size in this token classification is required.

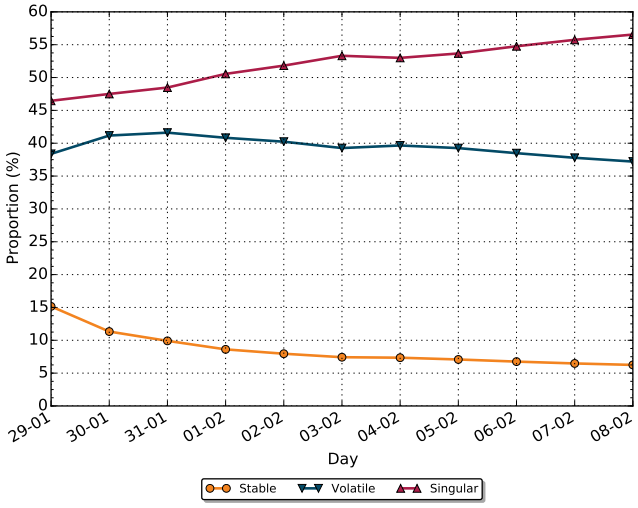


Fig. 3: Proportion of each type of token over days

4 Index Entries Invalidator

According to the preceding analysis it is reasonable to think about pruning certain entries of the inverted index. Specifically, those tokens whose daily frequency behaviour give us a hint about its scarce contribution to search results. Similarly to Lin & Mishne [15] observation, there is a great deal of "churn" in tweets content. To tackle this issue we decide to prune the inverted index by removing the full posting lists of those tokens that sparsely appear over days. Therefore, we define an *index entries invalidator* (IEI) by applying a time-to-live (TTL) strategy, as adopted in result caches [7]. The IEI is based on the time that entries have persisted in the index without been updated. When a token exceeds a certain threshold, the IEI invalidates and evicts this one along with its posting lists. In other words, an entry is dismissed when the difference between the current time and the last time it was updated is larger than the TTL value.

In order to evaluate this approach in a real time search scenario, we modify the Zambezi search engine, whose source code is publicly available. We implement the index entries invalidator to perform different experiments to ascertain the efficiency and effectiveness of the proposed approach.

5 Experimental Evaluation and Results

Methodology: Initially we set the TTL value to 24 hours in order to check whether an index entry has expired. By taking advantage of tweets timestamps, we are able to establish the beginning and end of the days, in terms of hours. Thus, every 24 hours during the 2 weeks covered by the sample we browsed the

index and evict the corresponding entries according to the heuristic previously defined. After every day, we processed 1 million queries[8] measuring efficiency and effectiveness by comparing the results obtained from the original index without pruning (our baseline), with those of the pruned one. We run three series of experiments retrieving the top-k documents that are most relevant to a query (with $k = \{50, 10, 100\}$). We configure Zambezi to use the MBWAND algorithm (basically, it uses timestamps as sorting criteria) that is suitable for microblogging documents. In other words, the answer set comprises a list of relevant tweets ordered from newest to oldest. Finally, to ensure the consistency of the results, we perform five trials of the experiment, and average the outcomes.

Due to lack of a publicly available real-time query set various works build up synthetic ones. Initially, we evaluate our approach with a shred of the synthetic query log generated in [20]. However, the number of queries in the set is not enough to run a robust performance evaluation. For this reason, we decide to use the one million queries extracted from the well-known AOL Query Log [21], employed in other works involving real-time search [15, 3].

Metrics: To assess the overall performance of the approach we evaluate both efficiency (time and space) and and effectiveness. In the first case, the execution time is measured in terms of wall-clock time on a per query basis. We also evaluate the number of invalidated entries per day and the index size reduction. To quantify the effectiveness, we apply the result set intersection between the baseline and our approach. Remember that in a real-time search scenario one of the primary search task consist in presenting the most recent documents related to the query (recency or freshness of the results) [18]. In order to accomplished this, the retrieved tweets are brought to the user in reverse chronological sorting.

Results: The effectiveness evaluation shows that the IEI doesn't degrade the results significantly. Figure 4 exhibits the intersection ratio (averaged from 1M queries) for the three series of experiments. In the case of top-10 retrieval, less than one document (on average) is missing in the pruned result set. This result is proportionally similar in the remaining series (top-50 and 100). A deep analysis of pruned tokens explains that most of them correspond to rare ones that appear sparsely in queries.

Regarding efficiency, our approach reduces the overall execution time in all configurations up to 6% in the best case. Figure 5 shows these results. The increased execution time over days corresponds to the vocabulary growth. The difference also increases conforming more tokens are added to the baseline thus, more tokens are invalidated and pruned. This suggests that a more aggressive pruning strategy is needed to complement the IEI and to control the vocabulary growth, in particular, posting lists pruning may be adequate (this is a direction for future research).

The space consumed by the inverted index in both baseline and pruned versions is also analyzed. Table 1 shows the number of entries in the vocabulary.

[8] Our sample preserves the original query length distribution.

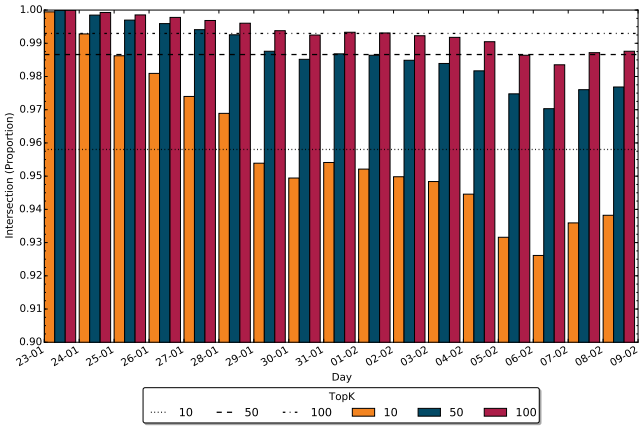


Fig. 4: Intersection proportion when retrieving top 10, 50 and 100 documents. Dotted lines correspond to mean values for each series.

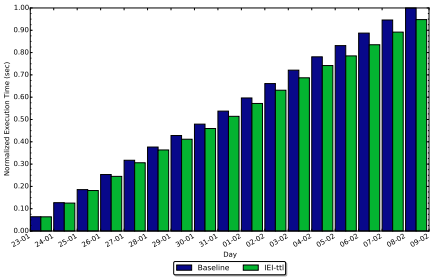
The number of valid entries decreases dramatically over days (up to 88%). This enables faster lookups into the vocabulary. However, the total number of DocIDs that accounts for the resulting posting lists decreases much slower (Table 2), up to 9.1% in the last day. Again, this happens due to the pruned tokens correspond to infrequent ones whose document frequency is very low (typically, one). This result reinforces the idea that a more aggressive pruning strategy that considers the posting list lengths of the tokens that remain into the vocabulary may lead to greater benefits.

Table 1: Number of entries in the inverted index

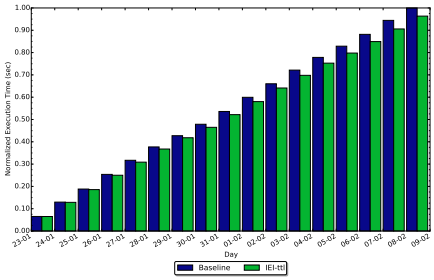
Day	Baseline	Pruned	Diff. %
1	217,044	209,379	3.53
2	390,863	231,976	40.65
3	540,456	238,330	55.90
4	692,677	259,240	62.57
5	833,499	250,156	69.99
6	970,939	256,852	73.55
7	1,081,271	223,664	79.31
8	1,186,153	228,312	80.75
9	1,295,470	239,079	81.54
10	1,407,580	258,560	81.63
11	1,519,236	254,509	83.25
12	1,627,904	258,206	84.14
13	1,736,197	259,738	85.04
14	1,827,029	227,970	87.52
15	1,916,499	230,474	87.97
16	2,012,312	245,609	87.79
17	2,105,807	249,280	88.16

Table 2: Sum of DocIDs in the posting lists of all terms

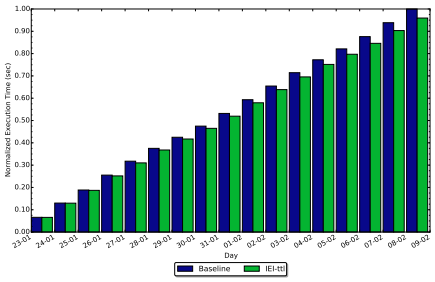
Day	Baseline	Pruned	Diff. %
1	2,392,099	2,384,262	0.33
2	5,174,584	4,992,039	3.53
3	7,864,478	7,467,306	5.05
4	10,919,587	10,294,900	5.72
5	13,892,852	12,991,206	6.49
6	16,859,239	15,678,477	7.00
7	19,413,681	17,914,817	7.72
8	22,037,617	20,270,901	8.02
9	24,934,503	22,903,479	8.15
10	27,976,632	25,684,408	8.19
11	31,168,827	28,576,005	8.32
12	34,215,991	31,326,117	8.45
13	37,198,345	33,998,347	8.60
14	39,793,764	36,240,415	8.93
15	42,532,978	38,669,507	9.08
16	45,659,362	41,508,202	9.09
17	48,608,362	44,153,772	9.16



(a) Top-10 results



(b) Top-50 results



(c) Top-100 results

Fig. 5: Normalized Execution Time for 1.000.000 queries

6 Conclusion and Future Work

In this work we introduced the concept of index entries invalidator, an approach that aims to selectively invalidate and evict those inverted index entries that do not considerably degrade retrieval effectiveness. Consequently, the index becomes smaller and may increase overall efficiency. Our experimental results showed that the proposed approach reduce the number of entries in the vocabulary up to an 88%, enabling faster lookups. The overall execution time for our query set is also reduced up to 6%.

However, the resulting size of the index decreases much slower, up to 9.1%. In order to deal with this issue, we plan to extend the IEI by pruning at both entry and posting list levels. To this extent, will be necessary to consider the posting list lengths of the tokens that remain into the vocabulary and study how they evolve over time. Moreover, we are interested in defining a second kind of invalidator following a different view of the vocabulary dynamics.

Acknowledgement

We give special thanks to the CIDETIC (Centro de Investigación Docencia y Extensión en TIC, UNLu. `http://cidetic.unlu.edu.ar/`) for providing us the necessary computational resources in order to conduct the corresponding experiments.

References

1. Asadi, N., Lin, J.: Fast candidate generation for real-time tweet search with bloom filter chains. ACM Trans. Inf. Syst. 31(3), 13:1–13:36 (2013)
2. Asadi, N., Lin, J.: An exploration of postings list contiguity in main-memory incremental indexing. LSDS-IR '14, New York, NY, USA (2014)
3. Asadi, N., Lin, J., Busch, M.: Dynamic memory allocation policies for postings in real-time twitter search. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1186–1194. KDD '13, ACM, New York, NY, USA (2013)
4. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern Information Retrieval - the concepts and technology behind search, Second edition. Pearson Education Ltd., Harlow, England (2011)
5. Blanco, R., Bortnikov, E., Junqueira, F., Lempel, R., Telloli, L., Zaragoza, H.: Caching search engine results over incremental indices. In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 82–89. SIGIR '10, ACM, New York, NY, USA (2010)
6. Busch, M., Gade, K., Larson, B., Lok, P., Luckenbill, S., Lin, J.: Earlybird: Real-time search at twitter. In: Proceedings of the 2012 IEEE 28th International Conference on Data Engineering. pp. 1360–1369. ICDE '12, IEEE Computer Society, Washington, DC, USA (2012)
7. Cambazoglu, B.B., Junqueira, F.P., Plachouras, V., Banachowski, S., Cui, B., Lim, S., Bridge, B.: A refreshing perspective of search engine caching. In: Proceedings of the 19th International Conference on World Wide Web. pp. 181–190. WWW '10, ACM, New York, NY, USA (2010)
8. Chen, C., Li, F., Ooi, B.C., Wu, S.: Ti: An efficient indexing mechanism for real-time search on tweets. In: Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data. pp. 649–660. SIGMOD '11, ACM, New York, NY, USA (2011)
9. Choi, J., Croft, W.B.: Temporal models for microblogs. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management. pp. 2491–2494. CIKM '12, ACM, New York, NY, USA (2012)
10. Efron, M.: Information search and retrieval in microblogs. J. Am. Soc. Inf. Sci. Technol. 62(6), 996–1008 (2011)
11. Grant, C.E., George, C.P., Jenneisch, C., Wilson, J.N.: Online topic modeling for real-time twitter search. In: Proceedings of The Twentieth Text REtrieval Conference, TREC 2011, Gaithersburg, Maryland, USA, November 15-18, 2011 (2011)
12. Huberman, B.A., Romero, D.M., Wu, F.: Social networks that matter: Twitter under the microscope. CoRR abs/0812.1045 (2008)
13. Java, A., Song, X., Finin, T., Tseng, B.: Why we twitter: Understanding microblogging usage and communities. In: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis. pp. 56–65. WebKDD/SNA-KDD '07, ACM, New York, NY, USA (2007)

14. Kwak, H., Lee, C., Park, H., Moon, S.: What is twitter, a social network or a news media? In: Proceedings of the 19th International Conference on World Wide Web. pp. 591–600. WWW '10, ACM, New York, NY, USA (2010)
15. Lin, J., Mishne, G.: A study of "churn" in tweets and real-time search queries. In: Proceedings of the Sixth International Conference on Weblogs and Social Media, Dublin, Ireland, June 4-7, 2012 (2012)
16. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York, NY, USA (2008)
17. McCreadie, R., Soboroff, I., Lin, J., Macdonald, C., Ounis, I., McCullough, D.: On building a reusable twitter corpus. In: Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1113–1114. SIGIR '12, ACM, New York, NY, USA (2012)
18. McCullough, D., Lin, J., Macdonald, C., Ounis, I., McCreadie, R.: Evaluating real-time search over tweets. In: Proceedings of the Sixth International Conference on Weblogs and Social Media, Dublin, Ireland, June 4-7, 2012 (2012)
19. Metzler, D., Cai, C., Hovy, E.: Structured event retrieval over microblog archives. In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 646–655. NAACL HLT '12, Association for Computational Linguistics, Stroudsburg, PA, USA (2012)
20. Nepomnyachiy, S., Gelley, B., Jiang, W., Minkus, T.: What, where, and when: Keyword search with spatio-temporal ranges. In: Proceedings of the 8th Workshop on Geographic Information Retrieval. pp. 2:1–2:8. GIR '14, ACM, New York, NY, USA (2014)
21. Pass, G., Chowdhury, A., Torgeson, C.: A picture of search. In: Proceedings of the 1st International Conference on Scalable Information Systems. InfoScale '06, ACM, New York, NY, USA (2006)
22. Soboroff, I., Ounis, I., Macdonald, C., Lin, J.: Overview of the trec-2012 microblog track. In: In Proceedings of TREC 2012 (2012)
23. Teevan, J., Ramage, D., Morris, M.R.: #twittersearch: A comparison of microblog search and web search. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining. pp. 35–44. WSDM '11, ACM, New York, NY, USA (2011)
24. Witten, I.H., Moffat, A., Bell, T.C.: Managing Gigabytes (2Nd Ed.): Compressing and Indexing Documents and Images. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1999)
25. Zobel, J., Moffat, A.: Inverted files for text search engines. ACM Comput. Surv. 38(2) (2006)

Comportamiento de Integración de Algoritmos para Descubrimiento de Reglas de Pertenencia a Grupos

José David Panchuk[1], Sebastian Martins[2], Horacio Kuna[1], Ramón García-Martínez[2]

1. Departamento de Informática. Facultad de Ciencias Exactas Químicas y Naturales. Universidad Nacional de Misiones
2. Laboratorio de Investigación y Desarrollo en Ingeniería de Explotación de Información. Grupo de Investigación en Sistemas de Información. Departamento Desarrollo Productivo y Tecnológico. Universidad Nacional de Lanús
josepanchuk@gmail.com, smartins089@gmail.com, hdkuna@gmail.com, rgm1960@yahoo.com

Resumen. Los procesos de explotación de información utilizan distintos algoritmos de minería de datos para obtener patrones de conocimiento a partir de los datos que se tienen sobre el dominio de problema. Uno de los supuestos con los que se trabajan estos algoritmos es que la complejidad del dominio de pertenecía de los casos que utilizan, no incide en la calidad de los resultados obtenidos. Es de interés analizar el comportamiento del proceso de explotación de información: Descubrimiento de Reglas de Pertenencia a Grupos, el cual utiliza algoritmos de clustering y algoritmos de inducción. En este trabajo se caracteriza la complejidad de los dominios en términos de las piezas de conocimiento que los describen y que los procesos de explotación de información buscan descubrir. Se muestra experimentalmente que en el caso del proceso de descubrimiento de reglas de pertenencia a grupos la calidad de los patrones que se obtiene difiere en función de los algoritmos que se utilizan en el proceso y de la complejidad de los dominios al cual aplican.

Palabras Clave. Ingeniería de Explotación de Información. Proceso de Explotación de Información. Complejidad de Dominios. Performance de par de algoritmos clustering e inducción.

1. Introducción

Varios autores [1-5] han señalado la necesidad de disponer de procesos de explotación de información que permitan obtener conocimiento a partir de las grandes masas de información disponible, su caracterización y tecnologías involucradas. En [6] se han definido cinco procesos de explotación de información: descubrimiento de reglas de comportamiento, descubrimiento de grupos, descubrimiento de atributos significativos, descubrimiento de reglas de pertenencia a grupos y ponderación de reglas de comportamiento o de pertenencia a grupos.

Los procesos de explotación de información utilizan distintos algoritmos de minería de datos para obtener patrones de conocimiento a partir de los ejemplos (instancias) que se tienen sobre el dominio de problema. Una de las hipótesis implícitas con las

que trabajan estos algoritmos es que fijados los algoritmos para el proceso de explotación de información, la complejidad del dominio sobre cuya información se aplican, no incide sobre la calidad de los patrones obtenidos. Sin embargo, hay indicios [7] que muestran que la complejidad de los dominios en términos de las piezas de conocimiento que los describen y que los procesos de explotación de información buscan descubrir, emerge como un componente no despreciable al momento de analizar la calidad de los resultados a obtener.
En este contexto, se busca demostrar mediante un experimento que para el caso del proceso de explotación de información descubrimiento de reglas de pertenencia a grupos, la calidad de los patrones producidos difieren en función de la complejidad de los dominios sobre los cuales se aplica y de los algoritmos que se utilizan en el proceso.
En suma, en este artículo se caracterizan los distintos tipos de complejidad de dominios y se expone el proceso de explotación de información descubrimiento de reglas de pertenencia a grupos (sección 2), se introduce el problema a partir del planteamiento de las preguntas de investigación (sección 3), se plantea el diseño experimental (sección 4), se describe el desarrollo del experimento (sección 5), y se presentan las conclusiones (sección 6).

2. Estado de la Cuestión

En esta sección se presentan los conceptos de proceso de explotación de información (sección 2.1) y de complejidad de dominios (sección 2.2).

2.1. Proceso de Explotación de Información: Descubrimiento de Reglas de Pertenencia a Grupos

En [6] se definen una serie de procesos de explotación de información los cuales guían al ingeniero de explotación de información en la identificación de los algoritmos de minería de datos a utilizar para la resolución de un problema de negocio particular. Para este artículo es de interés describir el proceso de descubrimiento de reglas de pertenencia a grupos el cual utiliza algoritmos de agrupamiento (clustering) y algoritmos de inducción para la extracción del conocimiento.
El proceso de descubrimiento de reglas de pertenencia a grupos aplica cuando el problema de negocio requiere identificar las características, descritas en términos de atributos y sus posibles valores, de un conjunto de clases a priori desconocidas, pero presente en la masa de información disponible sobre el dominio de problema. El proceso se puede describir de la siguiente manera: el primer paso consiste en identifican aquellas las fuentes de información existentes en la organización que posean datos relevantes para el problema en cuestión. Luego de identificadas las variables, estas se integran en una misma fuente de información (denominada datos integrados). Los algoritmos de agrupamiento se aplican una vez integrado los datos, obteniendo como resultado una partición del conjunto de registros en distintos grupos a los que se llama grupos identificados, generando posteriormente archivos asociados a cada grupo identificado (denominados como grupos ordenados). El atributo grupo de cada grupo ordenado se utiliza como el atributo clase de dicho grupo,

constituyéndose este en un archivo con atributo clase identificado (GR). Por último se implementa un algoritmo de inducción a partir del cual se obtienen un conjunto de reglas que definen el comportamiento de cada grupo.

2.2. Clasificación de Dominios por Complejidad

Para abordar el tema de la complejidad de los dominios, en [8] se propone caracterizar a los mismos en términos de piezas de conocimiento (reglas) que explican la pertenencia de una determinada instancia (ejemplo) a un determinado dominio. Es así que la complejidad del dominio queda caracterizada por la cantidad de clases que lo describe, la cantidad de reglas que definen la pertenencia a cada clase, la cantidad de atributos que puede tener cada regla y la cantidad de valores (distintos) que puede tener cada atributo.
Con base en los atributos de clasificación enunciados y el protocolo de clasificación de dominios propuesto en [7] se pueden clasificar en función de la complejidad de estos en los siguientes tipos:

"Dominios de Complejidad Simple: son aquellos dominios en que el aumento de la cantidad de ejemplos por regla, mejora el cubrimiento de reglas independientemente de las demás dimensiones utilizadas.

Dominios de Complejidad Mediana: son aquellos dominios que se explican con ejemplos con pocos atributos y pocas clases, ó pocos atributos y muchas clases ó pocas clases y pocas reglas por clase.

Dominios Oscilantes: son aquellos dominios que se explican con ejemplos donde pueden variar el número de atributos por ejemplo, ó cantidad de ejemplos soportados por una regla, o valores comunes de atributos en un conjunto de ejemplos cubiertos por la misma regla.

Dominios Complejos: son aquellos dominios que se explican con ejemplos con pocos atributos y muchos valores posibles por atributo, ó con muchos atributos y pocos valores posibles por atributo, ó con muchos atributos y muchos valores posibles por atributo.

Dominios Hipercomplejos: son aquellos dominios que se explican con ejemplos donde pueden variar la cantidad de posibles valores que pueden tomar los atributos, el número de atributos que cubren ejemplos, la cantidad de las reglas que cubren ejemplos, ó la cantidad de clases".

3. Preguntas de Investigación

En este proyecto se han planteado las siguientes preguntas de investigación:

- ¿Es correcta la suposición que la performance de un par cualesquiera de algoritmos utilizados para el desarrollo del proceso de descubrimiento de reglas de pertenencia a grupos es independiente de la complejidad del dominio?
- En caso de falseamiento de la suposición previa: ¿Cuál es el par <algoritmo de agrupamiento, algoritmo de inducción> que brinda una mejor comprensión de la situación según la complejidad del dominio?

4. Diseño Experimental

Para el diseño experimental se ha desarrollado una herramienta, basándose en el banco de pruebas propuesto en [9]. Dicha herramienta fue desarrollada en lenguaje JAVA 1.8 y como gestor de base de datos PostgreSQL 9.3. Además, la herramienta hace uso de distintas librerías para la ejecución de los algoritmos de agrupamiento e inducción, estas son: WEKA 3.6 y 3.7, WEKA Classification Algorithms 1.8, KEEL, y otras librerías propias de JAVA.

La herramienta permite establecer la complejidad del dominio sobre el cual realizar la experimentación, para ello se deben definir de manera manual (o el sistema posibilita la asignación automática) de los parámetros del dominio relevantes, generando las variables libres (sin restricciones en sus valores) al azar. Los parámetros fundamentales a definir para la determinación de la complejidad del dominio son:

Cantidad de clases (CC): la cantidad de clases diferentes que creará la herramienta, cada clase contará con un conjunto de reglas propias y por consiguiente un conjunto de ejemplos, la cantidad de clases dictará también la cantidad de grupos en los que se agruparan la totalidad de los ejemplos generados en un experimento.

Cantidad de atributos (CA): la cantidad de atributos que tendrán los ejemplos a generarse, cada atributo a su vez tendrá una cierta cantidad de valores posibles que podrá tomar.

Cantidad de valores posibles por atributo (CVA): la cantidad de valores diferentes que puede llegar a tomar un atributo, tanto en el momento de generar una regla que impondrá una condición sobre cierto atributo (en ese caso la cantidad de valores posibles estará acotada, como se verá más adelante) o al generar los ejemplos.

Porcentaje de valores posibles por atributo en reglas (PCVA): el porcentaje sobre la cantidad de valores posibles por atributo que se tendrán en cuenta para armar las reglas.

Porcentaje de ejemplos usados para el entrenamiento de los algoritmos (PENT): el porcentaje de los ejemplos generados que se utilizarán para entrenar los algoritmos de agrupamiento e inducción antes de ejecutarlos sobre la totalidad de los mismos, los ejemplos de entrenamiento son seleccionados aleatoriamente a partir del conjunto de ejemplos completo.

Cantidad de reglas a generar por cada clase (CRC): la cantidad de reglas que se generarán para cada clase, las reglas generadas describen a cada clase, ya que a parir de estas se generan los ejemplos del dominio.

Cantidad de ejemplos a generar por cada regla (CER): la cantidad de ejemplos a generarse para cada regla, debe tenerse en cuenta que la cantidad de ejemplos dependerá de la cantidad de clases que se hayan definidos, de la cantidad de reglas a generar por clase que se haya definido y a este parámetro en si, por lo tanto la cantidad de ejemplos generados se dispara rápidamente dependiendo de los valores dados a los parámetros nombrados.

Cantidad de atributos a utilizar en cada regla (CAR): este parámetro puede traducirse a la cantidad de condiciones que tendrá cada regla generada por la herramienta, el valor de este parámetro deberá ser menor o igual a la cantidad de atributos definidos.

Semilla de generación de números aleatorios (SEM): la semilla que utilizará la herramienta para generar los números aleatorios que alimentan a todo el proceso de experimentación.

En la figura 1 se muestra la sección de carga de datos del dominio a generar de la herramienta, en la cual se observan los parámetros a configurar y la complejidad del dominio seleccionada.

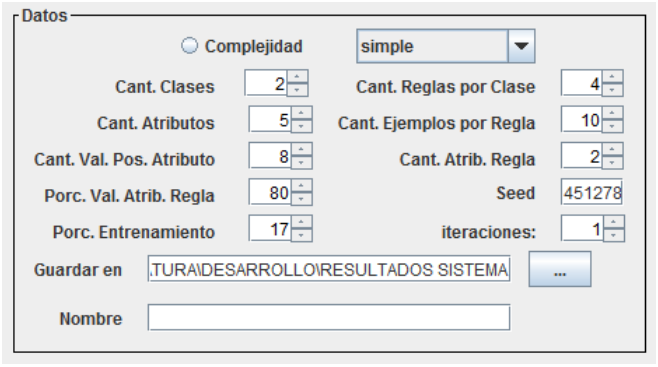


Fig. 1. Carga de Parámetros/Dominio

La definición de los dominios en condiciones de laboratorio conlleva la generación de un conjunto de reglas "originales", a partir de estas reglas se generarán ejemplos que den soporte a las mismas, cada regla tiene el siguiente formato "*if att1 = val1 and att2 = val2… and attn = valn then Cn"*. La cantidad de reglas a generar y la cantidad de condiciones de las reglas, así como la cantidad clases, atributos y valores diferentes, se definen en base a los valores establecidos por los parámetros del experimento.

El experimento está conformado por 3 pasos (figura 2):

[i] En el primero se define la complejidad del dominio que se desea analizar y se determinan los parámetros de los mismos, como resultado de este caso se generan las reglas del dominio y la cantidad de casos que respaldan dichas reglas.

[ii] El segundo paso consiste en la ejecución de cada uno de los posibles pares de algoritmos de minería de datos <clustering ; clasificación>, obteniendo así el conjunto de reglas descubiertas.

[iii] El tercer y último paso consiste en la comparación entre el conjunto de reglas de clasificación generadas en el primer paso y las reglas descubiertas en el segundo paso. El porcentaje de reglas descubiertas de forma correcta, define el éxito del experimento [9].

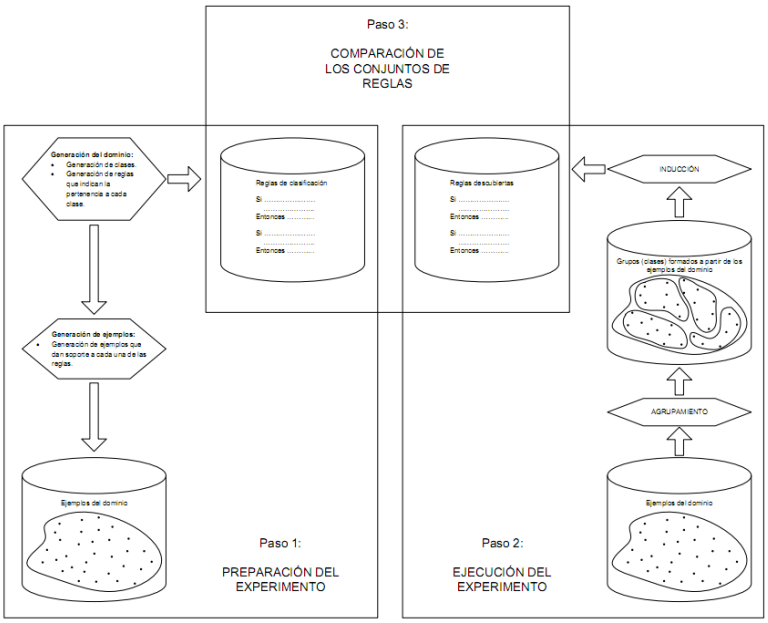


Fig. 2. Esquema de los pasos experimentales [9]

5. Desarrollo del Experimento

Para el desarrollo de la experimentación se realizaron 500 experiencias (100 escenarios para cada uno de los 5 tipos de dominios) a partir de las cuales se obtuvieron los resultados de análisis para cada una de las 30 combinaciones posibles de pares de algoritmos. Las variables de cada escenario fueron definidas de forma aleatoria según las variaciones posibles entre los dominios detallados en [7, 9]. Para la realización de los mismos se definieron categorías que establecen la variación de valores de cada variable de interés (tabla 1) y la relación entre dichas categorías (por parámetro) y la complejidad del dominio (tabla 2).

Tabla 1. Categorización de variación de Variables

VARIABLES	BAJO	MEDIO	ALTO
CC	2-4	5-7	8-10
CRC	1-5	6-10	11-15
CER	1-7	8-14	15-21
CAR	1-2	3-4	5-6
CVA	1-4	5-7	8-10
PCVA	1-40	41-70	71-100
CA	1-4	5-7	8-10

Tabla 2. Relación entre complejidad y variables independientes según el rango de variación

Complejidad\Variables	CC	CRC	CAR	CVA	CER	PCVA	CA
Simple					Alto		
Medianos	Bajo	Bajo	Bajo				Bajo-Medio
Oscilantes			Medio-Alto	Alto	Alto		
Complejos			Medio	Bajo-Medio	Alto	Medio	Medio
Hipercomplejos	Bajo	Bajo		Alto	Alto	Alto	Bajo

De la tabla 2 se deriva las posibles composiciones de cada escenario según su complejidad:

Escenarios de complejidad Simple: para los escenarios de complejidad simple la variable CER varía desde 15 hasta 21 mientras se mantiene fijo el valor del resto de las variables.

Escenarios de complejidad Media: en los escenarios de complejidad media, se varían las variables: Cantidad de Clases en el rango de 2 a 4, Cantidad de Reglas por Clase en el rango 1 a 5, Cantidad de Atributos Regla en el rango 1 a 2 y Cantidad de Atributos en el rango 1 a 5.

Escenarios de complejidad Oscilante: en los escenarios de complejidad Oscilante, se varían las variables: Cantidad de Atributos Regla en el rango 3 a 6, Cantidad de Valores Atributo en el rango 8 a 10 y Cantidad de Ejemplos por Regla en el rango 15 a 21.

Escenarios de complejidad Compleja: para los escenarios de complejidad Compleja, se varían las variables: Cantidad de Atributos Regla en el rango de 3 a 4, Cantidad de Valores Atributo en el rango 4 a 7, Cantidad de Ejemplos por Regla en el rango 15 a 21, Porcentaje de Valores Posible Atributo en el rango 45 a 70 y Cantidad de Atributos en el rango 5 a 7.

Escenarios de complejidad Hipercompleja: en los escenarios de complejidad Hipercompleja, se varían las variables: Cantidad de Clases en el rango de 2 a 4, Cantidad de Reglas por Clase en el rango 1 a 5, Cantidad de Valores Atributo en el rango 8 a 10, Cantidad de Ejemplos por Regla en el rango 15 a 21, Porcentaje de Valores Posibles Atributo en el rango 75 a 100 y Cantidad de Atributos en el rango 2 a 4.

De la ejecución de la experimentación, se generó una matriz por complejidad en la cual se detallan los resultados promedio de acierto de cada par de algoritmos a partir de los 100 casos generados. En las tablas 3 a 7 se presentan los resultados obtenidos.

En la tabla 3 se observan los resultados promedios obtenidos para dominios de complejidad Simple con cada combinación de algoritmos, se observa que la mejor combinación para este tipo de complejidad son los algoritmos SOM y ID3 con un 74.5% de reglas correctamente cubiertas en promedio, seguido por la combinación Farthest First y ID3 con un 66.75%, los resultados más altos desde el punto de vista inducción pueden verse en la columna del algoritmo ID3 y desde el punto de vista agrupamiento en la fila que pertenece al algoritmo SOM con sus respectivas combinaciones, los resultados más bajos se encuentran en las combinaciones de los algoritmos de agrupamiento con los algoritmos de inducción AQ15 y CN2

Tabla 3. Resultados Complejidad Simple

	PART	J48	ID3	A priori	CN2	AQ15
KMeans	50.5	57.25	60.37	51.37	15.25	8
EM	38.75	40.87	39.25	62.75	11.12	8
Farthest First	45.25	49	66.75	61.12	10.37	6.75
SOM	52.87	61	**74.5**	48.75	25.12	10.87
KNN	36.12	39	63.25	54.5	11.37	9.5

En la tabla 4 se encuentran los resultados promedios obtenidos para los dominios de complejidad Media usados, la mejor combinación de algoritmos para este tipo de complejidad fue Farthest First y ID3 con un 52.32% de reglas correctamente cubiertas en promedio, seguido por la combinación SOM y ID3 con un 51.70%, en general se puede observar que para este tipo de complejidad los resultados son menores en relación a los obtenidos para dominios de otras complejidades. Es interesante notar que las combinaciones de los algoritmos CN2 y AQ15 a pesar de seguir siendo los menores mejoraron en relación a los observados en la Tabla 5.5 de dominios Simples.

Tabla 4. Resultados Complejidad Media

	PART	J48	ID3	A priori	CN2	AQ15
KMeans	39.27	45.73	51.01	43.03	18.39	12.85
EM	32.84	34.97	41.72	49.63	15.18	9.71
Farthest First	35.05	38.11	**52.32**	46.08	15.70	8.94
SOM	40.61	45.16	51.70	40.03	18.09	11.26
KNN	38.66	41.45	45.74	34.10	18.76	13.01

En la tabla 5 se encuentran los resultados promedios obtenidos para los dominios de complejidad Oscilante, la mejor combinación de algoritmos para este tipo de complejidad fue KNN y A priori con un 94.25% de reglas correctamente cubiertas en promedio, seguido por la combinación EM y A priori con un 90.87%. Desde el punto de vista de los algoritmos de inducción claramente las combinaciones del algoritmo A priori presenta los resultados más altos, mientras que desde el punto de vista de los algoritmos de agrupamiento, el algoritmo KMeans con sus respectivas combinaciones tiene los promedios más altos, se observa que las combinaciones del algoritmo AQ15 fueron las más bajas y las combinaciones del algoritmo CN2 volvieron a aumentar.

Tabla 5. Resultados Complejidad Oscilante

	PART	J48	ID3	A priori	CN2	AQ15
KMeans	49.5	59.75	70.25	83.75	25.5	13.25
EM	39.62	48.5	48.75	90.87	21.87	11.5
Farthest First	48.75	53.5	71.5	83.75	22.37	9.25
SOM	29.62	30.87	61.62	84	20.12	2.5
KNN	9.62	9.37	19.5	**94.25**	13	1.5

En la tabla 6 se encuentran los resultados promedios obtenidos para los dominios de complejidad Compleja, la mejor combinación de algoritmos para este tipo de complejidad fue EM y A priori con un 90.25% de reglas correctamente cubiertas en promedio, seguido de la combinación KNN y A priori con 87%, nuevamente en este tipo de complejidad los resultados más altos se presentan en las combinaciones del algoritmo A priori con los algoritmos de agrupamiento, cabe destacar que en para este complejidad las combinaciones de ID3 también presentan valores altos y que desde el punto de los algoritmos de agrupamiento, las mejores combinaciones son aquellas donde esta presenta el algoritmo Farthest First.

Tabla 6. Resultados Complejidad Compleja

	PART	J48	ID3	A priori	CN2	AQ15
KMeans	50	53.37	68.87	81.75	19	12.37
EM	22.12	24.12	27.87	**90.25**	16.87	5
Farthest First	54.75	54.63	77.75	82.5	16.75	10.75
SOM	46.62	50	72.25	80.62	17.37	8.25
KNN	27.12	27.25	54.62	87	16.87	4.625

En la tabla 7 se observan los resultados promedios obtenidos para dominios de complejidad Hipercompleja con cada combinación de algoritmos, la mejor combinación de algoritmos para este tipo de complejidad fue **EM** y **A priori** con un 76.66% de reglas correctamente cubiertas en promedio, seguido de la combinación **Farthest First** y **A priori** con 70.84%, se observa además que en este tipo de complejidad se presentan los resultados más altos para las combinaciones con el algoritmo **CN2**, llegando a alcanzar y superar en ciertos casos a las combinaciones del algoritmo **PART**.

Tabla 7. Resultados Complejidad Hipercompleja

	PART	J48	ID3	A priori	CN2	AQ15
KMeans	36.74	41.60	49.21	66.86	37.04	18.24
EM	25.54	28.64	29.65	**76.66**	30.19	10.97
Farthest First	30.31	35.46	40.65	70.84	31.81	12.81
SOM	34.67	40.96	52.17	61.34	37.051	19.65
KNN	22.65	25.59	34.24	70.31	31.051	12.28

6. Conclusiones

A partir de los resultados previamente presentados, es posible aseverar que la performance alcanzada por los pares de algoritmos varía de acuerdo a la complejidad del dominio. Los mejores pares se presentan ordenados por la complejidad de manera creciente: <SOM, ID3>, <Farthest First; ID3>, <KNN, A PRIORI>, <EM, A PRIORI> y <EM, A PRIORI>. De los resultados obtenidos, se puede observar que el

algoritmo A PRIORI presenta mejoras significativas para las últimas 3 complejidades sin importar el algoritmo de clustering con el cual se lo empareje. Destacando que en las mismas obtuvo el máximo nivel de performance. El par <EM, A PRIORI> si bien presenta los mejores niveles de performance en las 2 últimas complejidades de dominio, en todas ellas presenta niveles aceptables de acierto. Los algoritmos PART y J48 presenta el mayor nivel de acierto general en el dominio más simple. Los algoritmos CN2 y AQ15 presentan los niveles más bajos de acierto, para todas las complejidades y combinaciones de algoritmos a nivel global. El algoritmo CN2 presenta mejoras en sus resultados a medida que la complejidad incrementa, si bien todos ellos están significantemente por debajo de la performance.
Como futuras líneas de investigación se prevé extender los algoritmos de agrupamiento e inducción en el software desarrollado y en una segunda etapa incorporar los procesos de explotación de información actualmente no considerados.

Financiamiento

Las investigaciones que se reportan en este artículo han sido financiadas por la beca PROMINF-UNLa-2015-2017 del Ministerio de Educación Argentina y parcialmente por el Proyecto UNLa-33A205 de la Secretaria de Ciencia y Tecnología de la Universidad Nacional de Lanús.

Referencias

[1] Chen, M., Han, J., Yu, P. (1996). *Data Mining: An Overview from a Database Perspective*. IEEE Transactions on Knowledge and Data Engineering, 8(6): 866-883.
[2] Chung, W., Chen, H., Nunamaker, J. (2005). *A Visual Framework for Knowledge Discovery on the Web: An Empirical Study of Business Intelligence Exploration*. Journal of Management Information Systems, 21(4): 57-84.
[3] Chau, M., Shiu, B., Chan, I., Chen, H. (2007). *Redips: Backlink Search and Analysis on the Web for Business Intelligence Analysis*. Journal of the American Society for Information Science and Technology, 58(3): 351-365.
[4] Golfarelli, M., Rizzi, S., Cella, L. (2004). *Beyond data warehousing: what's next in business intelligence?*. Proceedings 7th ACM international workshop on Data warehousing and OLAP. Pág. 1-6.
[5] Koubarakis, M., Plexousakis, D. (2000). *A Formal Model for Business Process Modeling and Design*. Lecture Notes in Computer Science, 1789: 142-156.
[6] García-Martínez, R., Britos, P., Rodríguez, D. (2013). *Information Mining Processes Based on Intelligent Systems*. Lecture Notes on Artificial Intelligence, 7906: 402-410.
[7] Lopez-Nocera, M., Pollo-Cattaneo, F., Britos, P., García-Martínez, R. (2011). *Un Protocolo de Caracterización Empírica de Dominios para Uso en Explotación de Información*. Proceedings XVII Congreso Argentino de Ciencias de la Computación. Pág. 1047-1055.
[8] Rancan, C., Pesado, P., García-Martínez, R. (2010). *Issues in Rule Based Knowledge Discovering Process*. Advances and App. in Statistical Sciences Journal, 2(2): 303-314.
[9] Kogan, A., Rancán, C., Britos, P., Pesado, P. y García Martínez, R. (2007). Algunos resultados experimentales de la integración de agrupamiento e inducción como método de descubrimiento de conocimiento. Proceedings IX WICC, Pág. 11-15. Universidad Nacional de la Patagonia San Juan Bosco, Trelew, Argentina.

Keyword Identification in Spanish Documents using Neural Networks

Germán Aquino[1, 2], Laura Lanzarini[1],

[1]Instituto de Investigación en Informática – III–LIDI Facultad de Informática – Universidad Nacional de La Plata - Argentina
[2] CONICET– Consejo Nacional de Investigaciones Científicas y Técnicas
{gaquino, laural}@lidi.info.unlp.edu.ar

Abstract. The large amount of textual information digitally available today gives rise to the need for effective means of indexing, searching and retrieving this information. Keywords are used to describe briefly and precisely the contents of a textual document. In this paper we present an algorithm for keyword extraction from documents written in Spanish.This algorithm combines autoencoders, which are adequate for highly unbalanced classification problems, with the discriminative power of conventional binary classifiers. In order to improve its performance on larger and more diverse datasets, our algorithm trains several models of each kind through *bagging*.

Keywords: Keyword Extraction, Neural Networks, Autoencoders.

1 Introduction

The large amount of textual information digitally available today gives rise to the need for effective means of indexing, searching and retrieving text documents quickly and without having a user to read them entirely, which in many cases is not feasible. Keywords are used to describe briefly and precisely the contents of a text document, so that a user can find documents relevant to him/her without having to read them beforehand. Keywords are widely used in search engines as they help in the process of searching, indexing, and retrieving information [1]. However, there are many documents without keywords and the task of manually assigning keywords to them is slow, difficult and highly subjective. For this reason it is beneficial to have tools that assist professional indexers by providing a list of terms candidates to be keywords [2].

In this paper a new algorithm for keyword extraction from text documents written in Spanish language is presented. This algorithm is based on a classification model capable of learning the structural features of the terms considered keywords, and to recognize terms having these features in unseen documents. A combination of discriminant classifiers and autoencoders is used to build a classification model that assigns a score to each term of a document. This score is used to construct a ranking of the terms considered most informative for a given document.

This paper is organized as follows. Some algorithms for keyword extraction are described in Section 2. The proposed algorithm is explained in detail in Section 3.

The results of the experiments carried out are presented in Section 4, and Section 5 summarizes the obtained conclusions and future work.

2 Related Work

The problem of keyword extraction has been treated from the machine learning discipline since a few decades ago [2][3][4]. This approach aims to transform text data into a structured representation suitable for learning algorithms. Such algorithms work with a feature set computed for each term of a document and consider keyword extraction as a classification problem, determining whether each term is a keyword or not. Supervised learning methods usually use the terms designated as keywords by the authors of the training documents as examples of one class, and the rest of the terms as examples of the other class. The class of the terms that are not keywords is naturally much more numerous than the other class. This imbalance in the number of elements of each class and the inherent ambiguity of natural language makes keyword extraction a very difficult problem to solve. Many of the mistakes made by the keyword extraction algorithms, specially those which apply supervised classification schemes, are due to redundancy (in the case of several semantically-equivalent terms are selected) and over-generalization (in the case of selection of terms that contain important terms but are not keywords themselves). The flexibility of the vocabulary used and the ambiguity of the human language makes very difficult for automatic classifiers to distinguish between two seemingly equivalent terms, and to see a relation between subtly related terms [5].

In order to find a suitable representation for learning algorithms, many keyword extraction methods apply *stemming*, which consists of reducing each term to its morphological root, and filter terms using a *stoplist*, which is a list of terms with low semantic value (*stopwords*) such as articles, prepositions, conjunctions and pronouns.

One of the first advances in considering keyword extraction as a classification problem to be solved through machine learning was reported by Peter Turney [2]. Turney developed an algorithm called GenEx that applies a set of rules whose parameters are tuned in a first stage using a genetic algorithm. These rules are used to rank terms and select the ones that have the highest score in the second stage. GenEx has a pre-processing step in which *stemming* is applied to terms and *stopwords* are filtered.

Among the most recent algorithms for keyword extraction there is Maui, developed by Olena Medelyan [6][7]. Maui is also a supervised classification algorithm that computes a set of features for the candidate terms. Maui uses a *stemmer* and a *stoplist* of the given language and it is built on top of the machine learning platform Weka [8] and uses bagged decision trees to classify terms.

In a previous work [9] we introduced a keyword extraction algorithm that relies on auto-associative neural networks or autoencoders [10] to identify keywords. This algorithm uses only the elements belonging to the minority class, the class of the keywords, to build a recognition model as opposed to discriminative models obtained using conventional neural networks and other machine learning algorithms. The autoencoder approach has the advantage that it handles naturally the imbalance

inherently present in the keyword extraction problem, and also it enables to control the number of keywords extracted from each document and to rank them. Also, it is much faster than other algorithms as it processes only the examples of the minority class.

The algorithm presented in this paper is also a supervised machine learning algorithm, and it is an improvement over our previous approach as it combines qualities of both discrimination-based (supervised) and recognition-based (unsupervised) classifiers in order to improve performance on larger and less regular datasets. The potentially large variance present in the training and testing examples is handled through the use of *bagging* [11] in order to average the classification decisions of different classifiers. As its predecessor, the proposed algorithm does not use *stoplists* to rule out insignificant or malformed terms but instead it applies *part-of-speech (POS) tagging* to allow the correct identification of noun phrases present in the text.

3 Description of the Algorithm

In this work autoencoders are used to classify terms in two classes, 'keyword' and 'non-keyword'. Autoencoders are adequate for unbalanced classification problems and one-class recognition problems [12]. To enhance their recognition capabilities, several autoencoders are combined by the use of bagging and also a set of discriminant classifiers is used.

An autoencoder processes examples of only one class. Autoencoders try to find an approximation of the training set to itself, finding in the process an approximation to the identity function of such training set. This allows them to assign a reconstruction error that characterizes the similarity between a new element and the training set. On the other hand, discriminant classifiers attempt to find a possibly non-linear boundary in the feature space of the training examples in order to define regions in such space for each class. Here, the decision of discriminant classifiers is used to weight the reconstruction error assigned to the examples by the autoencoders. Both kinds of classifiers made decisions through a voting scheme, which will be explained further in Section 3.3.

3.1 Pre-processing

The first step of the proposed algorithm consists in splitting the text in sentences and words using two list of delimiters provided as parameters. These delimiters can be any character sequence and will not be part of extracted terms. Once the sentences and words are obtained the algorithm proceeds to compute the features for the terms.

Terms are represented by N-grams, which are sequences of N consecutive words in the same sentence, and for each one we compute a set of features relative to position and frequency of the term in the document. In this work we will use 'term' and 'N-gram' interchangeably. In the N-grams extraction task the Fürnkranz algorithm [13] is applied for avoiding the generation of every possible N-gram from the text and increasing the efficiency in the generation of N-grams. This algorithm requires the

specification of the maximum length of the terms considered and the minimum frequency such terms must have in a document to be eligible as keywords.

In order to further reduce the number of terms to be processed, after the feature calculation phase we apply a filter which discards N-grams that do not start or end with nouns or adjectives. This filtering discards sequences of words that are not eligible as keywords, for example 'de forma que'. This process is similar to the application of a *stoplist*, with the difference that we do not use an exhaustive list of terms to rule out but instead we assign *POS* tags to each word of the document based on its use. To this end we apply a *maximum entropy* model trained with the tool OpenNLP [14] using a tagged corpus as training set. This filtering greatly reduces the required processing time, since it discards an important number of terms that should not be considered as keywords.

The POS tagging model for Spanish was trained using the tagged corpus Conll-2002 [15] and the grammatical tags defined by the EAGLES group [16]. The corpus was provided in the 2002 *Conference on Computational Natural Language Learning* to be used to train and evaluate algorithms of Named Entitity Recognition (NER), which is the problem of finding person names, places, organizations and similar information in the text.

3.2 Term characterization

The features computed for each N-gram consist of several frequential and positional quantities extracted from the text. Most of these features are computed using only the information present in each document, but some of them require the processing of the entire training corpus for their computation. The features are:

1. **Term length:** the number of individual words composing the N-gram.
2. **Term Frequency (TF):** the rate between the frequency of the term and the number of words in a document.
3. **Inverse Document Frequency:** it measures how common is a given term by counting how different documents in the corpus contain it.
4. **Term Frequency – Inverse Document Frequency (TF-IDF) [17]:** consists in weighting the term frequency with the inverse document frequency. TF-IDF favors terms that are infrequent in the corpus but frequent in the given document.
5. **First Occurrence:** the relative position of the first occurrence of the term in the text. It is calculated as the ratio between the number of words that appear before the first occurrence of the given term and the number of words of the document.
6. **Position in Sentence:** a measure of the relative position of a term in the sentences it appears in. For each sentence *s* that contains term *t*, we count the number of words that appear in *s* before *t*, and we average these values.
7. **Occurrence in Title:** this attribute is set to 1 if the term appears literally in the document title and 0 otherwise. It represents the notion that terms appearing in the title are important and hence are candidates to be keywords.
8. **Occurrence of Members in Title:** this attribute, like the previous one, relates the importance of a term with its appearance in the title. The difference is that this attribute considers occurrences in the title of the individual words of the term. This allows considering terms whose occurrences in the title are not literal, such

as when the words are in a different order or that have more or less lexical words. It is the ratio between the number of words of a term *t* that appear in the title and the length of *t*.

9. **Normalized Sentence Length:** it is a measure of the length of the sentences in which a given term appears in, calculated by averaging the lengths of these sentences. Such lengths are also normalized by dividing them by the length of the longest sentence in the document.
10. **Normalized Frequency (Z-Score) [18]:** consists in normalizing the term frequency using its mean frequency in the training corpus and its standard deviation. It measures the difference between the frequency of a term and its mean frequency in the corpus.
11. **Last occurrence:** the last position in the text in which the term appears.
12. **Spread:** the difference between first and last occurrences.
13. **Normalized frequency:** the frequency of the term normalized by the highest frequency of any term in the document.
14. **Lowest position in sentence:** considering all the positions a term occupied in each of its sentences, this is the closest to the beginning of the sentence, normalized using the sentence length.
15. **Highest position in sentence:** similar to the previous one, but considering the position closest to the end of the sentence.
16. **Shortest sentence length:** the length of the shortest sentence a term appears in, normalized by the highest length of any sentence.
17. **Longest sentence length:** similar to the previous one, but considering the longest sentence a term appears in.
18. **Log frequency:** a non-linear monotonic function is applied to the term frequency in order to reduce the impact of its absolute value but at the same time to keep its magnitude.
19. **Condition of being a named entity:** this is a boolean feature that indicates if the term is a named entity or not. To identify named entities in the document a NER OpenNLP model is applied.
20. **Keyphraseness [3]:** the number of times a given term was chosen as a keyword in the training set. It makes sense if the testing documents belong to the same domain as the training documents, which should be the case to obtain a reasonable performance.

3.3 Keyword Identification

As mentioned earlier, the proposed method is a supervised classification algorithm. It uses the feature vectors of the terms of the training document set in order to build a classification model to be applied to the feature vectors of a testing document set.

In the proposed method three ensembles of classifiers are used. The first ensemble is composed of conventional bagged *multi-layer perceptrons*, trained using sampling with replacement from the training set. In order to cope with the imbalance problem, the number of elements that are sampled from the majority class is proportional to the sampled number of elements in the minority class. As all of these sampled smaller training sets are different, the resulting classifiers will yield different views on the

original feature space. Given the large variance present in the problem domain and the intrinsic non-deterministic nature of neural networks, *bagging* helps to improve the performance of the obtained models, giving more consistent and more robust predictions. These classifiers are trained to distinguish important terms from non-important ones.

The other two ensembles are composed of autoencoders. The first of these two ensembles attempts to characterize the set of elements belonging to the minority class (the positive set), which in our case are the feature vectors of the terms designed as keywords in the training set. The other ensemble attempts to characterize the set of elements belonging to the majority class (the negative set), which is naturally much more diverse. Both ensembles are also trained applying *bagging*, and the autoencoders of the majority class are trained with larger samples in order to provide more accurate estimates of the complete set.

Autoencoders are neural networks that have as many output units as they have input units, so given an input vector X they can produce an approximate vector X'. The difference between the original vector and the approximate vector can be characterized by the reconstruction error, which is the sum of the squared differences between both vectors. As training is carried out using the elements of the class of interest it is expected that new elements that are similar to the ones in the training set have a lower reconstruction error than those that are not.

The autoencoders are trained in the same way as conventional neural networks. In this work we used *Resilient Backpropagation* [19] as training algorithm, both for the autoencoders and the multi-layer perceptrons. This algorithm allows a faster convergence, providing better results, and at the same time it eliminates the need to specify a learning rate.

As we mentioned earlier, the autoencoder assigns a reconstruction error to each element of a testing set, which represents the similarity between the element and those of the training set. Instead of determining a cutoff threshold to accept or reject a term as keyword we opted to select the R terms with lowest reconstruction error from each document of the testing set. As we are using two sets of autoencoders, one for the positive class and one for the negative class, we have two scores for each term of the testing set. Let Pos_e be the reconstruction error of the term in respect to the positive set, and Neg_e the reconstruction error in respect to the negative set. An informative term should minimize Pos_e, as it should be similar to the elements in the positive set, and at the same time it should maximize Neg_e, its dissimilarity to the negative set. Hence, an informative term should minimize $Pos_e - Neg_e$, and this is the score used to construct the term ranking. The selection scheme employed gives preference to the terms chosen by the discriminant classifiers as informative terms, and then their reconstruction error is considered.

The use of the reconstruction error as a selection mechanism provides two benefits: first, we obtain a *ranking* of the extracted terms, and second, it is guaranteed that each document of the testing set will have terms to represent it, which does not necessarily hold with the use of a global threshold or a discriminant classifier. Besides, R is a parameter of the algorithm which gives more control and allows the user to adjust the output of the algorithm when more precision or more recall is preferred. By default, the number of terms to extract is the average number of keywords of the documents of the training set.

4 Experimental Results

Some experiments were carried out to assess the performance of the proposed method. A dataset formed by a set of scientific articles published between 2005 and 2013 in Argentine Congress of Computer Science (CACIC) [20] was used in these experiments. The dataset includes 888 documents written in Spanish language and contains 130792 terms from which 1683 are labeled as keywords, giving an imbalance rate of 1.28%, that is, less than 2% of all terms belong to the minority class. We also used a dataset composed of 166 scientific articles from the Workshop of Researchers in Computer Science (WICC) [21]. This dataset was used to measure the performance of the previous version of our method [9], and it is used here to assess that the new version is indeed superior.

The metrics used were precision, recall and f_1-measure calculated for each of the four algorithms.These metrics were applied considering as a hit the match between a term selected by an algorithm and a term designated as keyword by the authors of the given document. Thus, a false positive occurs when a method identifies as keyword a terms that is not included in the list of keywords by the author, and a false negative when the method fails to extract a keyword contained in that list. In our case precision measures the proportion of extracted terms that match assigned keywords, and recallmeasures the proportion of keywords correctly identified by the method. F_1-measure is the harmonic mean between precision and recall, and therefore it is a good measure of the global performance of a given method.

The evaluation methodology we applied is 10-fold cross validation. This evaluation process was repeated 30 times to obtain a significative sample over which we can average the results. We configured both algorithms to extract 5 keywords as this is the average number of keywords per document on the dataset.

In our experiments we used 15 multi-layer perceptrons as discriminant classifiers, 5 autoencoders for the positive set, and 10 autoencoders for the negative set. All these neural networks were trained using 20 hidden neurons, a maximum of 50 epochs, and the logistic function as activation function in the hidden and output layers. The implementation used of Maui is the one developed by its authors. For Maui we applied the Spanish stemmers and stoplists provided with the implementations. For the previous version of our method, the autoencoder was configured to use 15 hidden neurons, a maximum of 100 epochs, and the same activation functions as the new version. In these experiments the terms extracted by all methods have a maximum length of 4 words and a minimum frequency of 3 occurrences in their respective documents.

The results of the 30 runs of the cross-validation for each algorithm on each dataset are shown in the Figure 1, identifying the proposed algorithm as AE*, for autoencoder. The previous version of our method is simply denoted as AE.

The tests results show that the proposed algorithm outperforms Maui on these datasets. It can be seen also that it handles properly larger and more diverse datasets than its predecessor. One of the main goals of our algorithm is to capture the largest possible number of descriptive terms, and this goal is quantified by the recall metric. A high recall is important because it allows capturing the maximum possible of eligible terms, which in turn gives the possibility of suggesting descriptive terms that

were not chosen by the authors. However, getting a high recall at the expense of precision is not beneficial, since the quality of the extracted terms will be inferior. Therefore it is necessary to find a balance between precision and recall.

In order to verify that these differences are statistically significant, we ran a Kolmogorov-Smirnov test on the results of the precision, recall and f-measure obtained from the cross-validation procedure for both methods, and we ran a t-test on the difference of the means of the samples for the three metrics. The tests showed that the mean for the three metrics obtained by our method are higher than the ones obtained by Maui with a significance level of 0.05, as the obtained p-values are 1.3669e-30, 3.7699e-40 and 4.2676e-35 respectively.

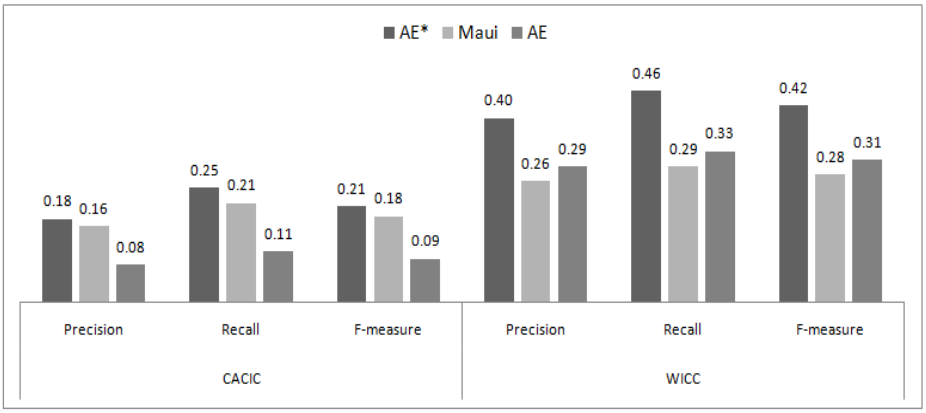


Figure 1. Average precision, recall and f_1-measure of the three methods on the CACIC dataset.

In the Table 2 there are shown the lists of keywords extracted of both methods for a set of documents from the CACIC dataset, and these keywords are compared to the real keywords assigned by the authors of the respective documents. The matches between an extracted keyword and a real one are highlighted in bold. It is important to notice that some of these documents have fewer keywords than the specified number of keywords to extract. This necessarily means that the methods will have false positives errors, despite the selected terms may be considered descriptive by a human observer. It is also noteworthy that some terms are semantically equivalent to the true keywords, but as they are not exact matches are hence considered false positives too. The high variability of the keyword assignment criteria of the authors, combined with the ambiguity of the human language contributes to the high difficulty of the keyword extraction problem. These issues could be addressed by the use of semantic knowledge bases that could map related terms to the same concept, and by the definition of more advanced scoring criteria for performance assessment than exact matching.

Table 2. Comparative results of the keyword extraction methods performance on some sample cases.

Documents in dataset	Keywords assigned by authors	Keywords extracted by AE*	Keywords extracted by Maui
Una implementación paralela de las Transformadas DCT y DST en GPU.	-procesamiento paralelo **-GPU** **-CUDA** **-procesamiento de señales** **-DCT**	-transformadas **-GPU** **-CUDA** -GPU CUDA **-procesamiento de señales**	-MPI **-DCT** -transformadas -DST **-CUDA**
Programación híbrida en clusters de multicore.	-arquitecturas paralelas **-programación híbrida** **-cluster** **-multicore** **-jerarquía de memoria**	**-cluster** **-multicore** -programación **-programación híbrida** **-jerarquía de memoria**	**-jerarquía de memoria** **-cluster** **-multicore** -pasaje de mensajes -caso de estudio
Evaluación de variantes en modelo destinado a anticipar la conveniencia de trazar proyectos de software.	**-ingeniería de software** **-análisis ROC** **-trazabilidad de requerimientos**	-trazabilidad -métricas **-análisis ROC** **-ingeniería de software** **-trazabilidad de requerimientos**	-ROC -trazabilidad -métricas -variantes -factores
Autorregulación del aprendizaje en entornos mediados por TIC.	**-autorregulación** **-TIC** **-aprendizaje**	**-autorregulación** **-TIC** **-aprendizaje** -intervención -autorregulación del aprendizaje	**-aprendizaje** **-TIC** -propuesta de intervención **-autorregulación** -intervención
Integración segura de MANETs con limitaciones de energía a redes de infraestructura.	-MANET **-bluetooth** **-IPSec** **-energía** **-seguridad**	**-bluetooth** **-IPSec** -MANETs **-energía** -ad hoc	**-seguridad** **-Bluetooth** **-IPSec** -consumo -consumo de energía

5 Conclusions and Future Work

In this paper we presented a new algorithm for keyword extraction from Spanish documents. The main feature of our proposal is the use of autoencoders to capture the properties of important terms, yielding comparable or even better results than other well known keyword extraction algorithms. Autoencoders classification decisions are further reinforced by the use of discriminant classifiers. We consider important to achieve a high recall so that the algorithm can capture more terms eligible by different human observers, with the goal to act as a recommendation system of possible keywords. The only language-dependent of our method are the POS tagging and NER models, thus replacing these models with models trained with documents in another language would allow us to apply our method in such language.

Given that the number of terms to extract is a parameter of the algorithm the user can adjust the expected level of precision or recall from the terms suggested by the system.

We are currently working on the term representation to include features related to the grammatical structure of a given language, as the use of parsing trees in order to find head noun phrases in sentences. We are also interested in incorporating the use of knowledge bases in order to find semantic relations between pairs of terms and to identify their degree of generality or specificity in a given domain.

References

1. Gutwin, C., Paynter, G., Witten, I., Nevill-Manning, C., Frank, E.: Improving Browsing in Digital Libraries with Keyphrase Indexes. Journal of Decision Support Systems, Vol.27, no 1-2, pp.81--104. (1999)
2. Turney, P.D.: Learning Algorithms for Keyphrase Extraction. Information Retrieval, vol. 2,303--336 (2000).
3. Witten, I. H., Paynter, G. W., Frank, E., Gutwin C., Neville-Manning, C. G.: KEA: Practical Automatic Keyphrase Extraction. In Proceedings of the 4th ACM Conference on Digital Libraries, pp. 254--255 (1998).
4. Hulth, A.: Improved automatic keyword extraction given more linguistic knowledge. In Proceedings of the 2003 Conference on Empirical Methods in NLP, pp. 216--223 (2003).
5. Hasan, K. S., Ng V.: Automatic Keyphrase Extraction: A Survey of the State of the Art. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1262--1273 (2014).
6. Medelyan, O.: Human-competitive automatic topic indexing. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, vol. 3, pp. 1318--1327, Association for Computational Linguistics (2009).
7. Kim, S. N., Medelyan, O., Kan, M., Baldwin, T. SemEval-2010 Task 5: Automatic Keyphrase Extraction from Scientific Articles. In Proceedings of the 5th International Workshop on Semantic Evaluation. pp. 21--26 (2010).
8. WEKA, http://www.cs.waikato.ac.nz/ml/weka/, accessed in July 2015.
9. Aquino, G, Hasperué, W, Lanzarini, L. Keyword Extraction using Auto-associative Neural Networks. XX CongresoArgentino en Ciencias de la Computación (2014).
10. Japkowicz, N, Myers, C, Gluck, M.: A Novelty Detection Approach to Classification. Proceedings of the Fourteenth Joint Conference on Artificial Intelligence, pp. 518--523 (1995).
11. Breiman, L.: Bagging Predictors. Machine Learning, pp. 123--140 (1996).
12. Japkowicz, N.: The Class Imbalance Problem: Significance and Strategies. Proceedings of the 2000 International Conference on Artificial Intelligence (ICAI), pp. 111--117 (2000).
13. Fürnkranz, J.: A Study Using n-gram Features for Text Categorization (1998).
14. OpenNLP, http://opennlp.apache.org/, accessed in July 2015.
15. Conference on Computational Natural Language Learning (CoNLL-2002), http://www.clips.ua.ac.be/conll2002/ner/, accessed in July 2015.
16. Expert Advisory Group on Language Engineering Standards (EAGLES), http://www.ilc.cnr.it/EAGLES96/home.html, accessed in July 2015.
17. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing and Management, pp. 513--523 (1988).
18. Andrade, M.A., Valencia, A.: Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. Bioinformatics, vol. 14, no. 7, pp. 600--607 (1998).
19. Riedmiller, M.: Advanced Supervised Learning in Multi-layer Perceptrons - From Backpropagation to Adaptive Learning Algorithms (1994).
20. Congreso Argentino en Ciencias de la Computación, http://redunci.info.unlp.edu.ar/cacic.html, accessed in July 2015.
21. Workshop de Investigadores en Ciencia de la Computación, http://redunci.info.unlp.edu.ar/wicc.html, accessed in July 2015.

Iris Information Management in Object-Relational Databases

Carlos Alvez, Graciela Etchart, Silvia Ruiz, Ernesto Miranda, Juan Aguirre, Marcelo Benedetto, Santiago Delfin

Facultad de Ciencias de la Administración, Universidad Nacional de Entre Ríos
Concordia, 3200, Argentina
{caralv, getchart, silruiz, emiranda, juaagu, marben, sdelfin}@fcad.uner.edu.ar

Abstract. Biometrics is a technology under development that has been enhanced by the increasing security concerns in organizations at all levels. Public agencies that employ this technology need to consult the biometric data efficiently and share them with other agencies. Hence the need for data models and standards that allow interoperability between systems and facilitate data searches. The objective of this work is to develop a generic architecture using object-relational database technology (ORDB), according to international standards, for identifying people by means of iris recognition. In addition, a model expressed in Unified Modeling Language (UML) class diagram where the domain data types defined for use in architecture is proposed. This architecture will allow interoperability between organizations efficiently and safely.

Keywords: Object-relational database, biometrics, iris.

1 Introduction

Currently, biometrics is a technology that is in full development, both in terms of everyday life and research. This rapid evolution is due to the growing concerns about security and the relationship that this technology has to ensure it. Public agencies that employ this technology need to consult the biometric data efficiently and share it with other agencies [1]. Hence the need for data models and standards which allow interoperability between systems and facilitate data searches.

As expressed above, this paper suggests a generic architecture based on an object-relational database adapted to international standards for identifying people by means of iris recognition [2]. For this, a model expressed in UML class diagram is also suggested, in which the data types used in the architecture, with its attributes and behavior are defined.

The article is outlined as follows: section 2 presents the adopted standards. Then, in section 3 the suggested architecture is presented. Next, the motivation and importance for the use of object-relational database is detailed for the implementation of the model. In section 5, the suggested architecture model is laid out. Finally, conclusions and future work are presented.

2 Adopted standards

This section describes the adopted standards for the management architecture of iris information. Firstly, in subsection 2.1 the ANSI / NIST ITL 1-2011 standard, focused to interoperability between systems is presented. This standard also sets the requirements for the representation and iris image compression, which will be discussed in subsection 2.2

1.1 Standards for interoperability between systems

The ANSI / NIST ITL 1-2011 [3] standard is a biometric standard published in November 2011, which defines how to ensure the interoperability of biometric data between different systems. This standard defines the content, format, and units of measurement for the electronic exchange of fingerprint, palmprint, plantar, face recognition, iris, deoxyribonucleic acid (DNA), and other biometric samples and forensic information that may be used in the identification or verification process of a subject. It is the standard mostly used by state entities, largely due to the intervention of the national state that since the Chief of Cabinet of Ministers establishes strict compliance with international standards and the adoption of best practices in this field [4].

This standard defines the composition of the records comprising a transaction that may be transmitted to another organization. The standard currently supports several different biometric modalities, and has reserved record identifiers for the possible future addition of other methods. All records in a transaction shall pertain to a single subject. Biometric data used to identify another individual requires a separate transaction. A transaction is comprised of records. Transactions shall consist of one Type-1 record and one or more of the Type-2 to Type-99 records. The Type-1 record is used to describe the transaction. In this work it is working with iris biometrics, so the transaction must contain at least one record Type-1 and Type-17 records. The Type-17 record specifies interchange formats for biometric authentication systems that use iris recognition.

Among its requirements, the standard ANSI / NIST ITL 1-2011 also has the requirements for representation and iris image compression. They are discussed below in the following subsection.

2.2 Standards for the representation and compression of images

An aspect to bear in mind for the exchange of biometric images is its quality of representation. In many instances, image exchange requires web transactions, so that the size thereof is important. In this sense, the data exchange standards also require compression formats to use methods that meet specific requirements. In many systems the baseline JPEG algorithm (ISO / IEC 10918) is used. However, this algorithm is not currently supported by the ANSI / NIST ITL 1-2011 to iris images. This is because it has been shown that false match rates are increased. As mentioned

above, the ANSI / NIST ITL 1-2011 standard only accepts JPEG 2000 (ISO / IEC 15444: 2004) [5] and PNG (ISO / IEC 15948: 2004) [6].

The international standard (JPEG 2000) ISO / IEC 15444: 2004 is an image coding system that allows compression, transmission and efficient storage of still images and sequences of images.

JPEG 2000 represents a family of specifications, developed jointly by the International Organization for Standardization (ISO), International Electrotechnical Commission (IEC) and the International Telecommunication Union (ITU). This international standard defines a set of lossless (bit-preserving) and lossy compression methods for coding bi-level, continuous-tone grey-scale, palletized color, or continuous-tone color digital still images.

The JPEG 2000 standard specifies:

- Decoding processes to turn compressed image data into reconstructed image data.
- A codestream[1] syntax containing information for the interpretation of compressed image data.
- A file format.

In addition, it provides guidance on:

- Encoding processes for converting source image data to compressed image data.
- How to introduce these processes into practice.

This international standard describes an image compression system which allows great flexibility, not only for the compression of images, but also for the access into the codestream. This codestream provides a number of mechanisms to locate and extract portions of compressed image data with the purpose of retransmitting, storing, displaying, or editing. This access enables the storage and retrieval of compressed image data appropriate for any application, without decoding.

The division of both the original image data and the compressed image data in several ways leads to the ability to extract image data from the compressed image data to form a reconstructed image with lower resolution or lower precision, or selecting regions of the original image. This enables the adaptation for a codestream to the transmission channel, storage device, or display device, regardless of the size, number of components, and sample precision of the original image.

Thus, the features of this specification allow a single codestream to be used efficiently by a number of applications. The largest image source devices can provide a codestream that is easily processed for the smallest image display device, for example.

There are four main elements described in this standard:

Encoder. An encoder takes digital source image data and parameter specifications as an input, and by means of a set of procedures generates a codestream as an output.

Decoder. A decoder takes compressed image data and parameter specifications as an input, and by means of a specified set of procedures generates digital reconstructed image data as an output.

Codestream syntax. It consists of a set of rules for the representation of compressed image data that includes all parameter specifications required by the decoding process.

[1] Sequence of bits representing the compressed image information.

Optional file format. It provides an optional file format for exchange between application environments. The codestream can be used by applications with other file formats.

The ISO/IEC 15948:2004 PNG (Portable Network Graphics) International Standard specifies a datastream and an associated file format for a lossless, portable, compressed image which can be broadcasted on the Internet. Indexed-color, grayscale, and true color (RGB) images are supported, with optional transparency. Besides, it supports a Sample depths range from 1 to 16 bits. PNG is fully streamable with a progressive display option. It is robust, providing both full file integrity checking, and detection of common transmission errors. PNG can store gamma and chromaticity data as well as a full ICC[2] color profile for an accurate color matching on heterogeneous platforms.

This Standard defines the Internet Media type "image/png".

Portability is among the main features of PNG. Encoding, decoding and transmission should be independent from software and hardware platforms. It may display true color, indexed-color, and grayscale images, with or without transparency, in addition to color space information, and auxiliary information such as textual comments.

The format was developed to be used on the Internet, so the encoding and decoding processes can be generated serially as the data stream is received on a serial communication channel. It is also designed to display an approximation of the whole image, which is progressively enhanced as the data stream is received.

By including a CRC it is possible to detect errors during transmission of the data streams.

The format uses filtering and compression processes which are lossless. In any of these processes decoding speed on the coding rate is prioritized, in order to accelerate the image display.

The algorithms involved in the process of encoding and decoding should be freely available.

The format is flexible to accept future additions and extensions without compromising the exchange of PNG data stream.

3 Proposed architecture

For biometric samples captured by any organism can be exchanged with other organisms, the system managing the biometric data must support transaction according to ANSI / NIST ITL 1-2011 standard, that is, it should be able to generate and / or use transactions that are morphological, syntactic and semantic as regards the requirements of the standard.

In Fig. 1, a generic architecture for managing iris images is presented.

In the proposed architecture, the tasks of generating, transmitting and receiving standard records are within the iris image exchange system. This system has the module that handles the generation and transmission of records necessary for an

[2] International Color Consortium: http://www.color.org/.

ANSI / NIST ITL 1-2011 transaction and another module for receiving these records from associated organisms [7]. In this paper, the standard Type-17 record corresponding to iris feature is used.

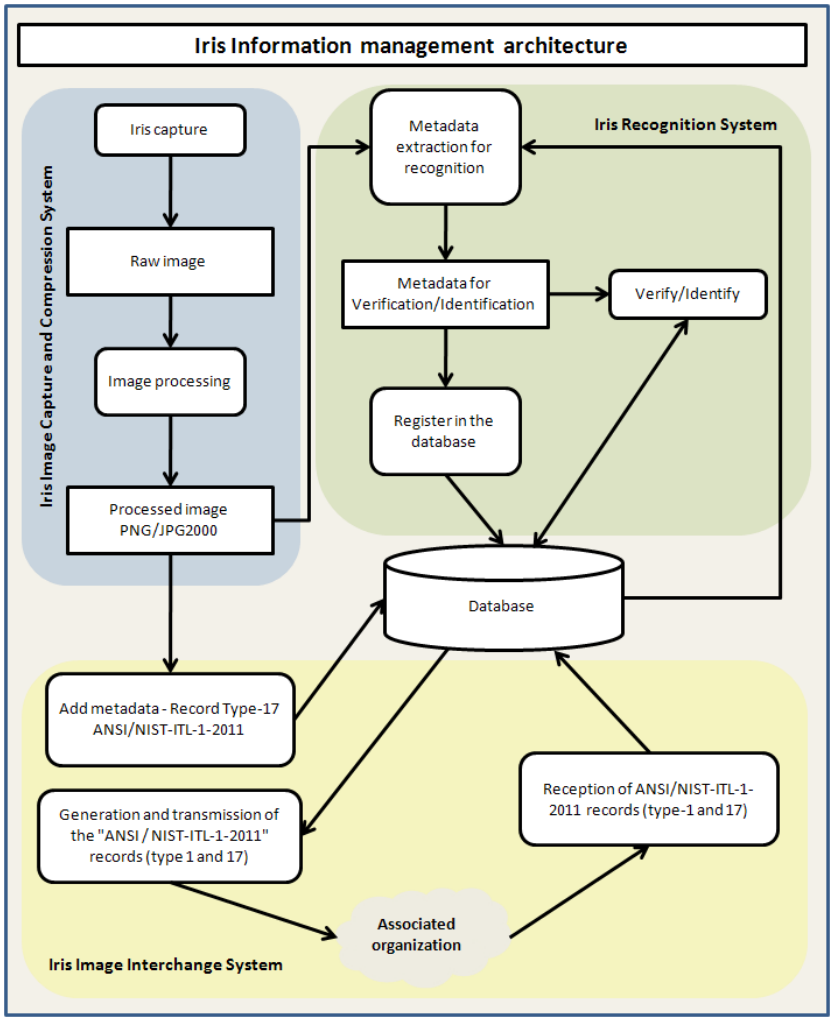


Fig. 1: Iris Information Management Architecture.

The Type-17 record of the aforementioned standard is used only for information exchange. However, you should also consider the generation and storage of metadata required for recognition, i.e. for the processes of identification and / or verification of people by means of the iris (IrisCode) [8] [9] which includes the subsystem for iris recognition.

The images that manage the systems mentioned can be obtained either from transactions with other agencies or from the system captures itself. For this, the capture system and iris image compression is used.

Both the structure Type-17 record and the metadata for iris recognition are complex structures. This can cause some problems when working with relational data model, because of the limitations imposed by it. Therefore, here we have opted for object-relational (OR) technology [10], as detailed in the following section.

4 Object-Relational Databases

The proposed architecture includes the representation of metadata related with iris biometrics data, so that they can be shared, accessed, retrieved and compared in a simple and efficient way. Furthermore, as mentioned above, the rendering quality of the images must be taken into account, particularly those compressed images for exchange. This also follows the specifications of the adopted standards. The representation of these complex data structures can cause some problems when working with relational data model by the following.

1. The structures of the metadata required, for both ANSI / NIST ITL 1-2011 records and the comparison of biometric data (IrisCode), are non-atomic structures, so in relational databases (in order to respect the first normal form) they must be separated into multiple tables. This makes it a less efficient treatment. For example, when comparing two sets of metadata it will be necessary to perform join operations.
2. The data processing must be done from particular languages (C++, Java, etc.), which presents the problem of compatibility and dependence on programming languages, since they must transfer the data generated by methods implemented in different languages to the database for storage. This issue is not trivial, and should be treated for each particular language.
3. In view of the above, so that a language to deal with data such as the method of comparing two biometric features, cursors must be created to be transferred through a network to return a single value. This can result in unnecessary network traffic, since these processes return a single logical value. For example, in a comparison used in the verification process, it returns whether or not it verifies.

Given the above drawbacks of the relational database, here we have chosen to use the OR technology. This technology can provide solutions to the limitations of relational databases since they provide the following abilities [11]:

1. *To define data types*. These types may involve complex structures such as collections, large objects, etc., without the limitation of the first normal form.
2. *To define and implement the data behavior*. Methods can be created to manage the data of defined types, thus facilitating safe access from applications that use, for example, the verification method that compares two IrisCodes.
3. *To define and implement a domain access methods*. It improves the access time in recovering data from a specific domain, as in the case of biometric data.

The mentioned abilities lead to the creation of infrastructures that extend the Database Management System services [12]. These infrastructures let the system deal with data from a specific domain, in this case, with biometric data.

As regards the project, this infrastructure should include:

- The creation of specific types for biometric data: This includes types for raw data (iris images in our case), types for encoding them (IrisCode), and all the metadata required to generate ANSI / NIST ITL1-2011 transaction records.
- Defining and implementing the methods that manage the types previously defined: This includes, among others, the generation of specific encodings, comparison methods (matching) and methods of generation and import of ANSI / NIST ITL1-2011 records.
- The creation of domain indexes to improve the response time in the identification processes: in biometric identification, you need to access multiple records to identify a person. Improving the efficiency in this access is not simple to be achieved with traditional indexes like B-trees, hash, etc. This is because the codes (templates) generated for comparison, are multidimensional data and methods for specific access domain are needed.

The following section suggests a model for Iris Information Management Architecture introduced in Fig. 1. In this model, we suggest the specific data types for representation biometric feature in question and methods necessary for the proper handling of them.

5 Proposed model

This section describes the proposed model for the representation of images and iris metadata adapted to standards. The model not only provides the representation of images with metadata, but also their behavior.

The Type-17 record from the ANSI / NIST ITL 1-2011 standard establishes the metadata required for iris feature. Fig. 2 shows a UML class diagram with the definition of the data types associated with that metadata. Fig. 3 shows a UML class diagram with the types of data needed to capture and process image, as well as the iris recognition system.

The UML class diagram in Fig. 2 shows the IrisImageRecord class with the necessary attributes for the transactional record's generation. This class consists of several other classes which represent different data types. Generally, it is about compound structures or repetitive structures which represent different kind of information about the captured images, devices used for the capture, places and date of capture. Also the diagram shows different enumerations and his literals, which are used by the main class and the other classes.

In the UML class diagram in Fig. 3 classes where images are captured and stored with their respective attributes is observed. The IMGJPG2000 and IMGPNG clasess inherit from the ProcessedImage class and are related with OpenJPG[3] and LIBPNG[4] packages.

[3] http://www.openjpeg.org/.
[4] http://www.openjpeg.org/.

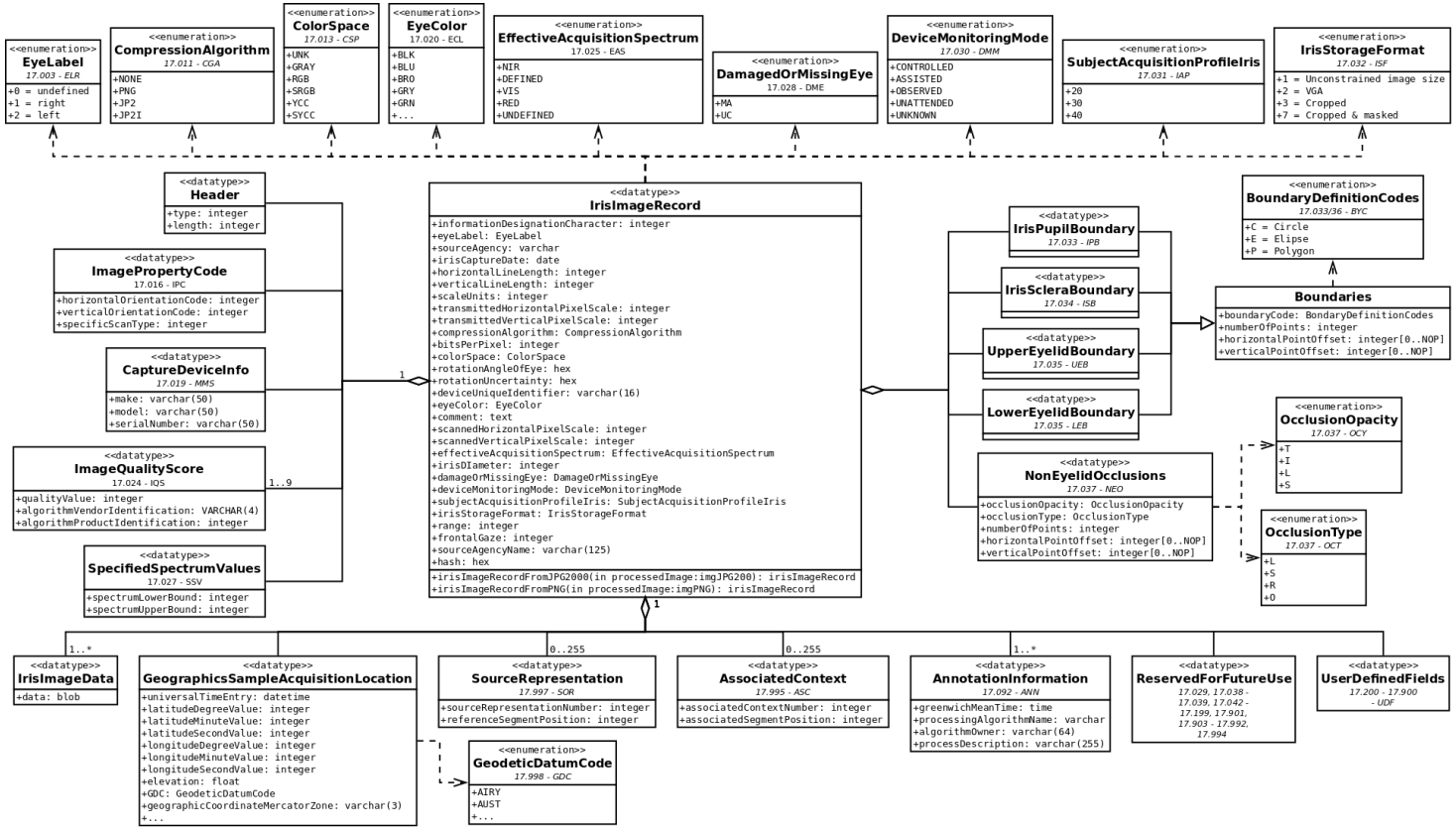


Fig. 2. Class diagram necessary for metadata required by Type-17 record from ANSI / NIST ITL 1-2011 standard.

OpenJPEG is an open-source JPEG 2000 codec written in C language. It has been developed in order to promote the use of JPEG 2000, a still-image compression standard from the Joint Photographic Experts Group (JPEG). The OpenJPEG code is released under the 2-clauses BSD license. Since May 2015, it is officially recognized by ISO/IEC and ITU-T as a JPEG 2000 Reference Software.

The PNG reference library (known as LIBPNG) is the official PNG reference library. LIBPNG is available as ANSI C (C89) source code. LIBPNG was written as a companion to the PNG specification, as a way of reducing the amount of time and effort it takes to support the PNG file format in application programs. The LIBPNG code is released under Open Source license.

The ProcessedImage class generates two subclasses by inheritance according to its implementation, which are associated with their respective libraries packages. This class, in turn, generates an IrisCode class with its respective methods of verification and identification.

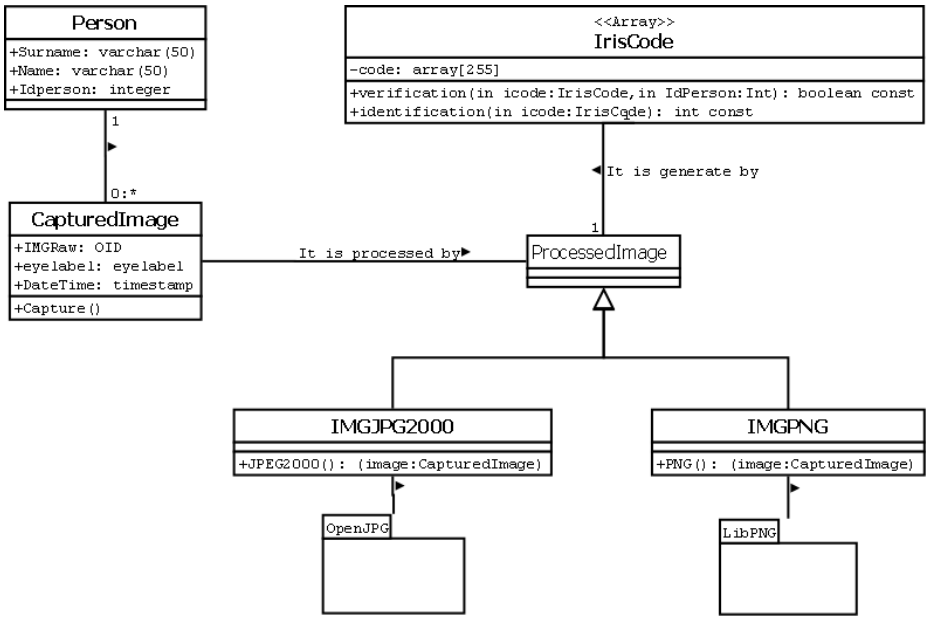


Fig. 3. Class diagram with data types required for capture and image processing and iris recognition system.

6 Conclusions and Future Work

This paper introduces a generic architecture for managing information in the iris biometric feature. This architecture uses object-relational technology. In addition, it suggests a model expressed in UML class diagram defining the domain data types to be used in the architecture. The suggested solution aims to obtain greater reliability,

performance and interoperability, using standards-based systems. It also provides a model in compliance with standards to capture, process and store data, ensuring the exchange of information between different agencies and / or applications.

In later stages, work will be focused on the implementation of methods to manage specific types for raw biometric data, data types for encoding them, and all the necessary metadata to generate records of ANSI / NIST ITL 1-2011 transactions. In addition, alternatives for creating domain indexes to improve response times in recognition processes will be studied.

References

1. Casal Gabriel, Rovolta Mercedes. Biometrías. Herramientas para la Identidad y la Seguridad Pública. Jefatura de Gabinete de Ministros. Presidencia de la Nación. (2010).
2. Ruíz, Silvia; Etchart, Graciela; Alvez, Carlos E.; Miranda, Ernesto; Benedetto, Marcelo Gabriel; Aguirre, Juan José. Representación e interoperabilidad de imágenes biométricas. XVII Workshop de Investigadores en Ciencias de la Computación, Salta, Argentina (2015).
3. Wing B. ANSI/NIST-ITL 1-2011. Update: 2013. Information Technology: American National Standard for Information Systems Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information. (2013).
4. Julio Fuoco, Tendencias Biométricas, desafíos y oportunidades. En Biometrías 2. Jefatura de Gabinete de Ministros. Presidencia de la Nación (2011).
5. ISO/IEC 15444-1:2004 Information Tecnology-JPEG 2000 Image Coding System: Core Coding System, Second Edition, (2004).
6. ISO/IEC 15948:2004 Information Tecnology-Computer Graphics and Image Processing-Portable Netware Graphics (PNG): Functional Specification, First Edition, (2004).
7. Graciela Etchart, Lucas Luna, Rafael Leal, Marcelo Benedetto, Carlos Alvez. "Sistema adecuado a estándares de reconocimiento de personas mediante el iris". CGIV - XIV Workshop de Investigadores en Ciencias de la Computación (WICC 2012), 25 y 26 de Abril de 2012. Universidad Nacional de Misiones. Posadas – Argentina. Páginas 321-325. (2012).
8. Daugman, J.: High condence visual recognition of people by a test of statistical independence. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(11), 1148-1161 (1993).
9. J. Daugman and C. Downing, "Effect of severe image compression on iris recognition performance," IEEE Transactions on Information Forensics and Security, vol. 3, no. 1, p. 52–61, (2008).
10. Melton Jim, "(ISO-ANSI Working Draft) Foundation (SQL/Foundation)", ISO/IEC 9075-2:2003 (E), United States of America (ANSI), (2003).
11. Carlos E. Alvez, Aldo R. Vecchietti. Combining Semantic and Content Based Image Retrieval in ORDBMS. Knowledge-Based and Intelligent Information and Engineering Systems Lecture Notes in Computer Science, 2010, Volume 6277/2010, pp. 44-53. Springer-Verlag Berlin Heidelberg (2010).
12. Alvez Carlos E. Models for the recovery of similarity-based images in Object-Relational Databases. Doctoral thesis. Santa Fe, Argentina (2012).

An experimental study for the Cross Domain Author Profiling classification

María José Garciarena Ucelay, María Paula Villegas, Leticia Cecilia Cagnina, Marcelo Luis Errecalde

Laboratorio de Investigación y Desarrollo en Inteligencia Computacional
Facultad de Ciencias Físico, Matemáticas y Naturales,
Universidad Nacional de San Luis – Ejército de los Andes 950
(D5700HHW) – San Luis – Argentina, Tel.: (0266) 4420823 / Fax: (0266) 4430224
emails: {mjgarciarenaucelay, villegasmariapaula74, lcagnina, merrecalde}@gmail.com

Abstract. *Author Profiling* is the task of predicting characteristics of the author of a text, such as age, gender, personality, native language, etc. This is a task of growing importance due to the potential applications in security, crime detection and marketing, among others. An interesting point is to study the robustness of a classifier when it is trained with a dataset and tested with others containing different characteristics. Commonly this is called *cross domain* experimentation. Although different cross domain studies have been done for datasets in English language, for Spanish it has recently begun. In this context, this work presents a study of cross domain classification for the author profiling task in Spanish. The experimental results showed that using corpora with different levels of formality we can obtain robust classifiers for the author profiling task in Spanish language.

Keywords: Author Profiling, Natural Processing Language, Cross Domain Classification

1 Introduction

The evolution of the World Wide Web sites to the Web 2.0 has mainly implied a proliferation of contents created and shared from all kinds of users in different social networks. Also, it has facilitated the increment of falsification of identity, plagiarism and a significant increase in the traffic of spam data through the Internet. For this reason, automatic methods are needed to detect if a given text belongs to a specific author, if the gender and age stated by a user of social media is compatible with his/her writing style, etc. In this context, the Author Profiling task refers to the identification of different demographic aspects like gender [1], age [2, 3], native language [4], emotional state [5, 6] or personality [5, 7] of an anonymous author of a text [8].

A particular problem concerned with the author profiling task in Spanish language is the lack of data for experimentation. For that, it is important to take in advantage of all the available data in order to obtain good and enough general classifiers for the task and then, to use those for new data that can be collected.

Traditional machine learning methods construct reliable and accurate models using available labeled data. These models are generally tested with data drawn from the underlying distribution or domain. Then, a classification model working well in one domain could not work as well in another one [9]. *Cross domain* classification is used to tackle that problem.

For *domain* we can consider the source of the documents (Twitter, blogs, chats, magazines, news, etc) [9], topics (places, politic, food) [10], products (books, furniture, movies) [11], research areas (computer science, biology, physics) [11], etc. In the present work we define the domain such as the level of "informality" of a text.

In PAN-2014 competition an extra experiment of cross domain was held, for both English and Spanish languages, which served as a previous work [12]. Thus, here we perform several experiments in order to determine the corpus we can obtain a general classifier with.

In this paper we present the results obtained from carrying out cross domain experiments. Such tests have not been previously performed in Spanish due to the lack of resources in this language and because training with a corpus and then testing with another is a recently studied approach. However, cross domain experimentation becomes an interesting field for researchers working in actual classification tasks as author profiling is. We have used available corpora provided for PAN competitions (2013 and 2014) which present a high level of informality in the texts contained. Also we have considered a formal corpus named SpanText [13] with similar characteristics with respect to those of PAN competitions, in terms of genre and age of people who wrote the texts. The results obtained with the experimentation with cross domain in terms of informality of the texts demonstrate that reliable classifiers can be obtained for the author profiling classification task.

The cross domain experiments may be helpful for other tasks, besides contributing to the author profiling itself. For example, it could be used to generate a classifier from a large collection of different types of texts, properly selected. Then, that classifier could be used to analyze texts hard to obtain or for analyzing online data.

The rest of the paper is organized as follows. In Section 2, we briefly introduce the author profiling task and some concepts related to cross domain experiments. In Section 3, the main characteristics of the different data collections used in the experimentation are presented. Section 4 describes an experimental study about cross domain among the available corpora in Spanish language. Finally, in Section 5 some conclusions are drawn and future works are proposed.

2 Author Profiling Task

Nowadays, the evolution of the Web sites on the Internet and the increasing use of social networks like Facebook and Twitter have made available a huge amount of information. A large part of this information is in plain text and it can be used to infer about the writer. The Author Profiling Task (APT) consists in knowing as much as possible about an unknown author, just by analyzing the given text [5]. In this regard, profiling tries to determine the author's gender, age, level of education, geographic origin, native language and personality type [1-8].

The APT has mainly focused on documents written in English but, according to our knowledge, this situation has started to change with papers presented at PAN-2013 competition [14], when the organizers considered the gender and age aspects of the author profiling problem, both in English and Spanish.

Now, we have several collections in Spanish with different kind of "formality". That is, a corpus is "informal" if the texts have noise like typos, images, hyper-links, emoticons, contractions, etc. This noise becomes the corpus in a very challenging dataset for any classifier. However, from the results of the competition PAN-2013 it can be seen that some approaches like the one used in [15] (the winner of the competition), can obtain interesting results even when the nature of the documents makes very difficult the classification. Unfortunately, it is unclear how these techniques work when these are trained with some corpora and tested with data with a different distribution. This was the reason that motivated us to study the cross domain approach.

When we talk about a *domain*, in the data mining field, it could be loosely defined as a specialized area of interest for which we can develop ontologies, dictionaries and taxonomies of information. We can refer to the different scopes (very broad or more narrowly specialized domains), or also the type of source from which the texts come from (like blogs, forums, etc.), or simply, a domain could be considered as the writing style (formal, informal, scientific, etc.). Thus, cross domain can be interpreted in several ways.

However, this paper simply assumed that a *domain* is a texts collection with a particular level of informality. Therefore, a *cross domain* experiment indicates a classification where you train with a corpus with certain level of informality and test with other with a different level of informality. Cross domain tests are also called by others authors as Domain Transfer experiments [16]. These consist in generating a classifier from texts that belongs to a source domain (training set) to apply it to a different target domain (test set). In other words, the underlying purpose of this concept is to check how well the trained classifier generalizes when it run on a different collection of documents.

3 Data Collections

We consider three different corpora in Spanish for the experimental study: *SpanText* and others two which were provided by the PAN-CLEF competition in the years 2013 [14] and 2014 [12]. These latter are called *PAN-2013* corpus and *PAN-2014* corpus. Also, we use a sub-corpus of PAN-2013 which we have proposed for this experimentation. The characteristics of each one are presented below.

SpanText is a set of "formal" documents written in Spanish extracted from the Web [13]. In this context, we use the term "formal" (as opposed to "informal") to refer to those documents whose content has a low percentage of "non-dictionary" words, abbreviations, contractions, emoticons, slang expressions, etc. that are typical in messaging and the social Web. This dataset consists of a variety of texts that one supposes to find in newspapers, students' reports, books and so on. These "speak" about different topics and they were written by Spanish speakers from Spain and Latin American countries. Besides, there are only one document (file) per author.

Two versions of this collection were presented in [13]. They are called "balanced" and "unbalanced" versions. However, there is another one called "semi-balanced", in which we are interested. Spantext (like PAN-2013) considers age and gender as the basic demographic information for the authors. All the documents are labeled with both characteristics. For age detection, it contemplates three classes: 10s, 20s and 30s. In the semi-balanced version, the number of documents per class is proportional to the amount of PAN-2013's documents. These are only uniformly distributed with respect to gender.

Regarding the PAN-2013 collection, it was built automatically with texts from blogs and other social networks [14]. The organizers of the competition provided two corpora: one in English and other in Spanish language. The dataset was divided into the following sub-sets: training, early bird evaluation and final testing. In this work, PAN-2013 will refer to the training and test sets of the Spanish language. Documents in PAN-2013 considered a wide spectrum of topics and they include "informal" text. The posts were grouped by author selecting those authors with at least one post and chunking in different files with more than 1000 words in their posts. But it also included some authors with few and shorter posts. For age classification, this collection considers the three same classes as SpanText and it is balanced by gender and imbalanced by age group, having more texts in class 20s than in 30s, and more in 30s than in 10s.

However, due to the difference between the sizes of SpanText and PAN-2013, it was needed to separate a sub-corpus of the latter (called sub-PAN2013), so it has the same number of documents per category as the semi-balanced version of the former. Thus, the results of diverse experiments can be fairly compared and the difference in the results will be limited to other variables, such as the quality of the texts.

The collection of texts written in Spanish in the PAN-2014 corpus was collected semi-automatically from four different sources: social media, blogs, Twitter and hotel reviews (the last only provided in the English corpus). In the competition of the year 2014, the PAN-CLEF organization opted for modeling age in a more fine-grained way and considered the following ranges (classes): 18-24, 25-34, 35-49, 50-64 and 65+ years old. The full collection was also divided into training, early bird evaluation and final testing parts. It is worth noting that we could access only to the training set and we use that part in the experimental study because the test corpus is not available at the time of writing this article.

Table 1. Vocabulary (number of words without repetition), number of terms (words), number of files and average number of terms for each collection.

Collection	SpanText	PAN-2013	PAN-2013 sub-corpus	PAN-2014
#Vocabulary	31 504	342 068	29 616	306 809
#Terms	294 434	22 868 586	294 596	17 686 634
#Files	1 000	84 060	1 000	1 500
Average Terms	294	301	294	11 806

Table 1 shows the statistics for each full collection. We can observe that PAN-2013 corpus presents the biggest numbers except in average number of words. This is because of its structure, it has more files (or authors) and more wealth in terms of writing styles but, the texts are not too long.

If we compare SpanText with PAN-2014, the latter is 50% bigger than the former, but SpanText only has a 10% of vocabulary than its counterpart. PAN-2014 prioritized the amount of texts from the same author, rather than the number of authors. It was probably because these are often short texts due to the source from which they came from (e.g. Twitter). This is verified in the amount of average terms for document that overcomes highly the other two corpora.

However, we must emphasize that in this regard SpanText is not far from the PAN-2013 collection. Perhaps if we could increase the number of documents of SpanText, maintaining its characteristics, this corpus would become the most useful. Since the proportion between, the amount of repeated words and the vocabulary is 10% for SpanText and 1% for both PAN-2013 and PAN-2014.

4 Experimental Study

In this section, we describe the cross domain experiments performed using the software WEKA [17]. Basically we performed two kinds of studies: APT as a classification of documents by gender, and then, considering both together age and gender. This is because, as we previously mentioned, the corpus PAN-2014 considered different age ranges from the ones defined in PAN-2013; in such way that we cannot make a join or separation of categories in order to consider the same ranges of age for both corpora.

Table 2(a) shows the information about the cross domain experiments: name of the corpus used for training and amount of documents considered, and name of the corpus used for testing with the corresponding amount of documents for performing the classification only by gender. The same information for the classification by gender and age considered together is shown in Table 2(b). From now on, to refer to a particular experiment, first we will mention the name of the corpus that was used to train, followed by the name of the collection employed to test (short forms of the original names of the corpora). For example, SPAN-PAN13 corresponds to the experiment which uses SpanText to generate the model and PAN-2013 to validate it.

Table 2. List of the cross domain experiments carried out.

(a) Classifications only by gender

Training	Docs	Test	Docs
PAN-2014	1 500	SpanText	1 000
SpanText	1 000	PAN-2014	1 500
PAN-2014	1 500	PAN-2013	84 060
PAN-2013	84 060	PAN-2014	1 500
SpanText	1 000	PAN-2013	84 060
PAN-2013	84 060	SpanText	1 000

(b) Classifications by age and gender

Training	Docs	Test	Docs
PAN-2013	84 060	SpanText	1 000
SpanText	1 000	PAN-2013	84 060
Sub-PAN13	1 000	SpanText	1 000
SpanText	1 000	Sub-PAN13	1 000

We used two traditional models of representation of documents: *bag of words* (BoW) [18] and *character trigrams* [19]. Regarding the weighting schema, we employed: *Boolean* [18] and *tf-idf* [20]. We also considered the *Second Order Attributes* (SOA) representation [13] because it has been demonstrated to be effective for this task. We have constructed the models and performed the classification using N*aïve Bayes* [21] and *LibLINEAR* [22] methods. Besides those, we considered an interesting approach *Sistema de Perfiles* (SP) [23] which generates its own model (profiles) using the most frequent character trigrams of the texts (L value) and then evaluates the belonging of the test documents in the profiles. It is important to note that due to the characteristics of its functioning, we could not use SP for those experiments which required to train with the PAN-2014 collection, because it was not able to generate the required profiles for the classification. The values for the L parameter of SP mentioned in the tables were chosen from carrying out prior executions for different values of this, choosing the one with we obtained the best accuracy. All approaches were evaluated considering the accuracy as metric.

4.1 Classifications only by gender

The percentages of correctly classified instances (accuracy) obtained in the cross domain classification only by gender are shown in Table 3. The table is divided into three sub-tables (a), (b) and (c) considering three different cross domain experiments. The highest accuracy values obtained are highlighted in boldface. The first value is the accuracy obtained with Naïve Bayes algorithm and the one after the slash corresponds to the accuracy obtained with the LibLINEAR algorithm.

Table 3. Accuracy obtained in cross domain classifications only by gender with "Naïve Bayes / LibLINEAR" algorithms.

	(a) PAN-2014 and SpanText		(b) PAN-2013 and SpanText		(c) PAN-2014 and PAN-2013	
	PAN14-SPAN	SPAN-PAN14	PAN13-SPAN	SPAN-PAN13	PAN13-SPAN	SPAN-PAN13
Boolean words	50,0 / 48,2	54,1 / 52,6	53,0 / 58,1	**53,1** / 52,1	50,3 / 52,4	57,5 / **67,5**
TF-IDF words	51,7 / 53,6	52,7 / 52,9	51,6 / **60,9**	51,1 / 51,9	52,4 / 53,3	58,3 / 64,9
SOA words	**61,2** / 59,4	54,9 / **55,4**	60,1 / 53,0	50,1 / 50,0	**59,1** / 58,5	61,1 / 62,6
Boolean 3grams	50,0 / 49,8	48,9 / 51,5	51,0 / 55,8	**57,7** / 51,6	50,1 / 49,5	50,7 / 58,8
TF-IDF 3grams	50,1 / **53,7**	51,3 / 51,3	54,8 / **60,3**	50,6 / 50,1	**54,9** / 54,6	56,3 / **62,2**
SP 3grams	-	**54,3**	58,7	51,3	-	57,8

The baseline used by PAN-CLEF Lab competition to determine if a two-class classifier is acceptable is 50%. Table 3 shows that almost all percentages exceeded or

equaled this value (48,2; 49,8; 48,9 and 49,5 are the exception). Note that with PAN13-SPAN it was not obtained percentages lower than the 50%.

Figure 1 provides a visual summary of Table 3. The bars with no plot at the left correspond to the representation of documents and the bars with plot (dots and rhombus) at the right with classifiers. The accuracy shown is the average of all the accuracies obtained for each approach for every training corpus used. Furthermore, results are shown from the baseline so that it would highlight better the differences obtained. It is important to note that words strategies dominate character trigrams approaches.

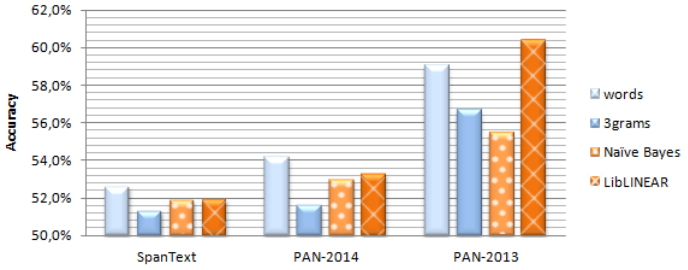
If we analyze the document representations, in general SOA accomplish the best performing, which precisely works with words. Next it follows the SP with character trigrams, then, in third and fourth places are the tf-idf representation with words and character trigrams respectively. Certainly, with a little more elaborated approaches than the simple use of frequencies, it achieves better results.

Regarding the classifiers, it can be concluded that LibLINEAR is superior. Out of the eleven best results in bold, six were obtained with it. Moreover, if an average of executions is calculated grouping them by classifier, it results that using LibLINEAR the average accuracy is around 60%, while with Naïve Bayes reaches only 56%.

The highest results were achieved when we trained with PAN-2013 and tested with PAN-2014, 67.5% for words and 62.2% for character trigrams. If we make an average of all executions in which this corpus was used to train the model, we found that this obtained the best percentage. This is also exhibited in Figure 1. Therefore, with 57.8% against 52.9% training with PAN-2014 and 51.9% with SpanText, we can say that the PAN-2013 collection is the one that generates a more general classifier.

At the PAN-CLEF competition in 2014, they tested the approaches of the participants who participated in 2013 (the approaches were trained with PAN-2013 corpus) using the 2014 collection (testing with PAN-2014). The SP achieved 69.4% of accuracy taking the first position in the final ranking [12]. Observing the results obtained we conclude that with the PAN-2013 collection we can get a general model able to classify documents from different corpora. Additionally, the results of the experiments accomplished in this work, at least for classifications only by gender are promising and overcome at least in a 3% the experiments performed on a single domain.

Figure 1. Summary of the results obtained for the cross domain classification only by gender distinguished by representations and classifiers.



4.2 Joint classifications by age and gender

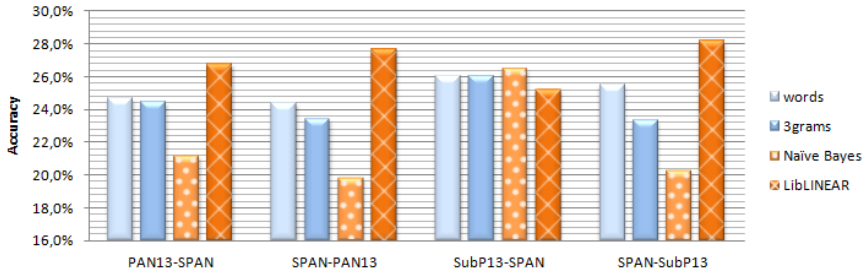
The results obtained for the cross domain classification considering age and gender are shown in Table 4. The best result of each section is highlighted in boldface. The baseline for this case is 16% because there are six categories (the combination of female and male with the three ranges of age). In Table 4 there are three cases in which the percentage does not reach the baseline. The first correspond to SPAN-PAN13 combination employing words-tf-idf representation and Naïve Bayes. Then, the second and third cases use character trigrams-Boolean representation with Naïve Bayes classifier, SpanText to train and PAN-2013 (or its sub-corpus) to test. However, when an average of the results is calculated, for example based on the classifiers, we can say that all the values are over the baseline.

Table 4. Accuracy obtained in cross domain classifications by age and gender with "Naïve Bayes / LibLINEAR" algorithms.

	(a) SpanText and PAN-2013		(b) SpanText and sub-PAN2013	
	PAN13-SPAN	SPAN-PAN13	subP13-SPAN	SPAN-subP13
Boolean words	19,7 / 26,5	20,1 / 28,1	20,0 / 25,7	19,8 / 28,3
TF-IDF words	19,3 / **29,5**	14,0 / 28,3	24,0 / 24,8	17,5 / **29,6**
SOA words	25,6 / 27,4	**28,7** / 27,1	**35,1** / 26,4	29,2 / 28,8
Boolean 3grams	20,5 / 22,8	11,5 / **28,0**	25,0 / 23,7	15,1 / 26,9
TF-IDF 3grams	20,7 / 27,7	24,8 / 26,9	**28,4** / 25,5	19,8 / 27,3
SP 3grams	**30,5**	25,9	27,5	**27,6**

Figure 2 summarizes the information of Table 4. In the bars the different models of representation (words and character trigrams) are at the left and they do not have a plot. Whereas the bars with dots and rhombus that are at the right, represent the behavior of the classifiers. As compared to the cross domain classifications only by gender, where words always predominated, here the bars exhibited are more similar among them. So it seems that the character trigrams help to distinguish better out the six categories.

Figure 2. Summary of the results obtained for cross domain classifications by age and gender distinguished by representations and classifiers.



If we analyze the traditional representations, i.e. Boolean and tf-idf, we obtained better results using words when we trained with the complete PAN-2013 corpus (Table 4 (a)). In particular, the combination of the tf-idf representation with the classifier LibLINEAR has worked considerably well. However, when it is trained with SpanText, the character trigrams strategy achieves a higher percentage on average. Now if we consider slightly more elaborated approaches in SPAN-PAN13 combination, the SOA representation is the best at discriminating the different classes. Nevertheless, the best overall result for the joint classification by gender and age is reached in PAN13-SPAN with the SP.

Table 4 (b) shows the results obtained with the sub-corpus of PAN-2013 which are different than those obtained with the complete corpus of PAN-2013. Even though, this case is a specific one thereof.

In general, regarding the classifiers, Naïve Bayes obtained poor results, highlighting even more the difference in performance respect to its counterpart. As we mentioned above, LibLINEAR with tf-idf representation using words obtained the second best result for cross domain classification by gender and age using the whole corpora.

Thus, in these experiments the same behavior is observed as in the classifications only by gender in which the approaches that use words are better. This is evidenced by the 35.1% obtained with the SOA representation in PAN13-SPAN combination. In addition, the highest percentage is accomplished again using the sub-corpus PAN-2013 to train the model.

5 Conclusions and Future Work

Cross domain experimentation has started to raise the interest of researchers turning their attention to the possibility of building a general enough classifier to classify any type of text documents. Hence its importance in the APT in which it is difficult to find properly labeled and lesser noise collections of texts, particularly for the Spanish language, is significant. For example, to detect pedophiles on the network or other kind of tasks that require a real-time response, and where the previous training with information which is not necessarily of the same type of the task to evaluate, is limited or non-existent.

In this paper we present a preliminary study considering cross domain author profiling classification. We made different experiments considering some corpora for training and testing using others considering different level of formality.

We analyzed the corpora available for APT in Spanish language using different representations and classification algorithms. Aiming not only to see how well a corpus generalizes a model, but also to evaluate the desirable characteristics that should have them, we conclude that the PAN-2013 collection is the one which better serves for that purpose. The highest accuracies were obtained with more elaborate representations such as SOA and approaches such as SP. Therefore, the results of the cross domain experiments obtained in this study turn to be promising, since they get close and even exceed the values obtained in experiments conducted in a single domain (or inter-domain).

Finally, it would be interesting to verify how the SP approach would behave when it trained with the PAN-2014 collection, and instead of using character trigrams, using words or more sophisticated representations.

References

1. Koppel, M., Argamon, S., and Shimoni, A. R. Automatically Categorizing Written Texts by Author Gender. Literary and Linguistic Computing. Vol. 17, no 4, pp. 401–412, 2002.
2. Argamon, S., Koppel, M., Fine, J., and Shimoni, A. R. Gender, Genre, and Writing Style in Formal Written Texts. TEXT. Vol. 23, pp. 321–346, 2003.
3. Schler, J., Koppel, M., Argamon, S., and Pennebaker, J. W. Effects of Age and Gender on Blogging. In AAAI Spring Symposium: Comp. Approaches to Analyzing Weblogs. Vol. 6, pp. 199–205, 2006.
4. Koppel, M., Schler, J., and Zigdon, K. Determining an Author's Native Language by Mining a Text for Errors. In Proc. of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining. ACM, pp. 624–628, 2005.
5. Rangel, F. Author Profile in Social Media: Identifying Information about Gender, Age, Emotions and beyond. Proc of the 5th BCS IRSG Symposium on Future Directions in Information Access, pp.58–60, 2013.
6. Bo, P., and Lee, L. Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval. Vol. 2, issue 1-2, pp. 1–135, 2008.
7. Pennebaker, J.W., Mehl, M.R., and Niederhoffer, K. Psychological Aspects of Natural Language Use: Our Words, Our Selves. Annual Review of Psychology. Vol. 54, pp. 547–577, 2003.
8. Argamon, S., Koppel, M., Pennebaker, J. W., and Schler, J. Automatically Profiling the Author of An Anonymous Text. Communications of the ACM. Vol. 52, pp. 119–123, 2009.
9. Ramakrishna Murty, M., Murthy, J.V.R., Prasad Reddy, P.V.G.D., and Satapathy, S.C. A survey of cross-domain text categorization techniques. In Recent Advances in Information Technology (RAIT), 1st International Conference, pp. 499–504, 2012.
10. Li, L., Jin, X., and Long, M. Topic Correlation Analysis for Cross-Domain Text Classification. In Proc. AAAI, 2012.
11. Pan, S. J., Ni, X., Sun, J., Yang, Q., and Chen, Z. Cross-Domain Sentiment Classification via Spectral Feature Alignment. In proc. of the 19th international conference on World wide web, pp. 751–760, 2010.
12. Rangel, F., Rosso, P., Chugur, I., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., and Daelemans, W. Overview of the 2nd Author Profiling Task at PAN 2014. Proc. of the Conference and Labs of the Evaluation Forum (Working Notes), 2014.
13. Villegas, M. P., Garciarena Ucelay, M. J., Errecalde, M. L., and Cagnina, L. C. A Spanish text corpus for the author profiling task. In Proc. of XX CACIC. San Justo, Buenos Aires, Argentina, pp 1-10, 2014.
14. Rangel, F., Rosso, P., Koppel, M., Stamatatos, E., and Inches, G. Overview of the Author Profiling Task at PAN 2013. Notebook Papers of CLEF, pp. 23–26, 2013.
15. López-Monroy, A. P., Montes-y-Gómez, M., Escalante, H. J., Villaseñor-Pineda, L., and Villatoro-Tello, E. Inaoe's Participation at PAN'13: Author Profiling Task. Notebook PAN at CLEF 2013, 2013.
16. Finn, A., Kushmerick, N., and Smyth, B. Genre classification and domain transfer for information filtering. In Advances in information retrieval, Springer, pp. 353–362, 2002.
17. WEKA. Data Mining Software in Java. http://www.cs.waikato.ac.nz/ml/weka/
18. Feldman, R., and Sanger, J. The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, 2007.
19. Cavnar, W. B., and Trenkle, J. M. N-gram-based text categorization. Ann Arbor MI, 48113(2):161–175, 1994.
20. Manning, C. D., Raghavan, P., and Schütze, H. Introduction to Information Retrieval. Cambridge University Press, New York, USA, 2008.
21. Lin, J. Automatic author profiling of online chat logs. Doctoral thesis, Monterey, California. Naval Postgraduate School, 2007.
22. Fan, R.-E., Chang, K.-W., Hsieh, C.-J.,Wang, X.-R., and Lin, C.J. LIBLINEAR: A library for large linear classification. The Journal of Machine Learning Research, 9:1871–1874, 2008.
23. Fúnez, D., Cagnina, L., and Errecalde, M. Determinación de Género y Edad en Blogs en Español Mediante Enfoques Basados en Perfil. In XVIII CACIC, 2013.

List of Clustered Permutations in Secondary Memory *

Patricia Roggero[1], Nora Reyes[1], Karina Figueroa[2], and Rodrigo Paredes[3]

[1] Departamento de Informática, Universidad Nacional de San Luis, Argentina
{proggero,nreyes}@unsl.edu.ar
[2] Facultad de Ciencias Físico-Matemáticas, Universidad Michoacana, México
karina@fismat.umich.mx
[3] Departamento de Ciencias de la Computación, Universidad de Talca, Chile
raparede@utalca.cl

Abstract. Similarity search is a difficult problem and various indexing schemas have been defined to process similarity queries efficiently in many applications, including multimedia databases and other repositories handling complex objects. Metric indices support efficient similarity searches, but most of them are designed for main memory. Thus, they can handle only small datasets, suffering serious performance degradations when the objects reside on disk. Most real-life database applications require indices able to work on secondary memory.
Among a plethora of indices, the List of Clustered Permutations (LCP) has shown to be competitive in main memory, since groups the permutations and establishes a criterion to discard whole clusters according the permutation of their centers. We introduce a secondary-memory variant of the LCP, which maintains the low number of distance evaluations when comparing the permutations themselves, and also needs a low number of I/O operations at construction and searching.

Keywords: metric spaces, permutation-based algorithm, secondary memory

1 Introduction

"Proximity" or "similarity" searching is the problem of looking for objects in a dataset, that are "close" or "similar enough" to a given query object, under a certain (expensive to compute) distance. Similarity search has become a very important operation in applications that deal with unstructured data sources. For example, multimedia databases that manage objects without any kind of structure, such as images, fingerprints or audio clips. This approximation has applications in a vast number of fields. Some examples are non–traditional databases, text searching, information retrieval, machine learning and classification, image quantization and compression, computational biology, and function prediction. These problems can be mapped into a *metric space model* [3]. That is, there is a universe $\mathbb{X}$ of objects, and a non negative real valued distance function $d : \mathbb{X} \times \mathbb{X} \longrightarrow \mathbb{R}^+ \cup \{0\}$ defined among them. This distance satisfies the three axioms that make the set a *metric space*: *strict positiveness* ($d(x, y) \geq 0$ and $d(x, y) = 0 \Leftrightarrow x = y$), *symmetry* ($d(x, y) = d(y, x)$), and *triangle inequality* ($d(x, z) \leq d(x, y) + d(y, z)$).

* Partially funded by Fondecyt grant 1131044, Chile.

The smaller the distance between two objects, the more "similar" they are. We have a finite *database* $\mathbb{U} \subseteq \mathbb{X}$, $|\mathbb{U}| = n$, which is a subset of the universe and can be preprocessed to build an index. Later, given a new object from the universe (a *query* $q \in \mathbb{X}$), we must retrieve all similar elements found in the database. There are two typical similarity queries:

- *Range query* (q, r)*:* retrieve all elements within distance r to q in $\mathbb{U}$.
- k*-Nearest neighbor query (*k*-NN):* retrieve the k closest elements to q in $\mathbb{U}$.

Our focus is on *approximate proximity searching*, where accuracy can be traded off for efficiency, as opposed to exact similarity search algorithms. However, there are generic techniques to convert any exact algorithm into approximate by using a form of aggressive pruning, as described for example, in [6].

For general metric spaces, there exist a number of methods to preprocess the database in order to reduce the number of distance evaluations [3, 14, 13]. In general metric spaces, the (black-box) distance function is the only way to distinguish between objects, and usually, the function of distance is expensive to calculate (in time and/or resources), compared to the CPU time to traverse the index and decide which elements are relevant. However, when the index is located in secondary memory the I/O operations are also very significant [12]. Therefore, the goal of similarity search algorithms for metric spaces in secondary memory is to solve queries using the minimum number of distances computations and I/O operations.

Since this kind of datasets lacks of total order to avoid a full linear scan, it is necessary to preprocess the database to build an index which allows answering queries with less effort. The *List of Clusters* (LC) [2] is one of the most efficient algorithms in high dimensional spaces (difficult spaces), however it takes $O(n^2)$ distance calculations to build the index. In other hand, the *Permutation Based Algorithm* (PBA) [4, 5, 9] is an approximate method that has been showed unbeatable in practice, but it only works well in high dimensions, as the authors claim. Once the index is built by calculating the "permutation" of each database object, during searches we have to calculate the permutation of the query object q and compare it with all permutations of database objects, to compute the order to review permutations. This takes at least $O(|\mathbb{P}|)$ distance calculations, where $|\mathbb{P}|$ is the permutation size, and $O(n)$ evaluations of the "permutation distance". There have been several proposals to avoid the sequential scan in PBA, however all of them lost accuracy regarding the original technique [7, 10]. In [9], a combination of the main ideas of LC and PBA is presented, designing a new metric index to answer approximate similarity search. This new index, called as *List of Clustered Permutations* (LCP), achieves a good search performance and beats both LC and PBA.

However, when we want to answer approximate similarity queries on large volumes of data, working in secondary memory and considering distance and I/O costs is necessary. The I/O time is composed of the number of disk pages read and written; we call B the size of the disk page in bytes. Given a dataset of $|\mathbb{U}| = n$ objects of total size N bytes and disk page size B, queries can be trivially answered by performing n distance evaluations and N/B I/Os. The goal of a secondary-memory index is to preprocess the dataset so as to answer queries with as few distance evaluations and I/Os as possible.

Therefore, in this article we use the idea of LCP [9], built on LC and PBA, but considering the index have to be located in secondary memory. So, the idea is to keep

each cluster of the list on a disk page in secondary memory. Hence, the cluster size must consider the disk page size. Besides, in order to accelerate searches the information of centers (permutants) and some few more data are also maintained in main memory.

The rest of this paper is organized as follows. In Section 2 we describe the previous works and some basic concepts. Next, in Section 3 we detail the *List of Clustered Permutations* (LCP) and in Section 4 we present our secondary-memory variant of LCP. In Section 5 we show the experimental evaluation of our proposal. Finally, we draw some conclusions and future work directions in Section 6.

2 Previous Works

In order to introduce our secondary-memory index, we describe briefly the main aspects of the previous works used as basis.

2.1 Permutation-Based Algorithm

In [4, 5] the authors introduce the permutation based algorithm (PBA), a novel technique that shows a different way to sort the space. At preprocessing time, a subset of objects $\mathbb{P} = \{p_1, p_2, .., p_{\|\mathbb{P}\|}\} \subseteq \mathbb{U}$ is selected out of the database, which are called the *permutants*. Each object $u \in \mathbb{U}$, computes its distance to all the permutants (i.e., computes $d(u, p)$ for all $p \in \mathbb{P}$) and sorts them increasingly by distance. Then, for each object $u \in \mathbb{U}$, just the order of the permutants (not the distances) is stored in the index.

If we define Π_u as the permutation of $(1, \ldots, \mathbb{P})$ for the object u, so $\Pi_u(i)$ is the i-th cell in the u's permutation and $p_{\Pi_u(i)}$ denotes the i-th permutant. For example, if $\Pi_u(i) = (5, 2, 1, 3, 4)$ then $p_{\Pi_u(2)} = p_2$. Within the permutation, for all $1 \leq i < \|\mathbb{P}\|$ it holds either $d(p_{\Pi_u(i)}, u) < d(p_{\Pi_u(i+1)}, u)$ or, if there is a tie $(d(p_{\Pi_u(i)}, u) = d(p_{\Pi_u(i+1)}, u))$, then the permutant with the lowest index appears first in Π_u. We call the i-th permutant $\Pi_u(i)$, the *inverse permutation* Π_u^{-1}, and the position of i-th permutant $\Pi_u^{-1}(p_i)$. The set of all the permutations stored in the index needs just $O(n|\mathbb{P}|)$ memory cells.

During searches, we compute the distance from the query $q \in \mathbb{X}$ to all the permutants in $\mathbb{P}$ and obtain the query permutation Π_q. Next, Π_q is compared with all the permutations stored in the index, which takes $O(n)$ permutation distances.

Authors in [4] claim that the order induced by the permutation of the query object q (Π_q) is extremely promising and reviewing a small fraction of the dataset is enough to get a good answer.

The *permutation distance* is calculated as follows: let Π_u and Π_q permutations of $(1, \ldots, |\mathbb{P}|)$. We compute how different is a permutation from the other one using *Spearman Rho* (S_ρ) metric. In [8] the S_ρ distance is defined as:

$$S_\rho(\Pi_u, \Pi_q) = \sqrt{\sum_{1 \leq i \leq \mathbb{P}} (\Pi_u^{-1}(i) - \Pi_q^{-1}(i))^2}$$

The main disadvantage of the PBA is that its memory requirement could be prohibitive in some scenarios, especially where n is actually huge. Also, like other indices, the dimension of the space impacts on its performance; particularly, it has an effect on how long is the fraction to consider when solving the approximate similarity query.

2.2 List of Clusters

There are many indices for metric spaces [3, 14, 13, 1]. One of the most economical in space used and rather efficient is the *List of Clusters* (LC) [2], because it needs $O(n)$ space and has an excellent search performance in high dimension. Regretably, its construction requires $O(n^2)$ distance evaluations, which is very expensive. The LC index is built recursively. The LC has two variants, one with a fixed cluster size and the other with a fixed cluster radius. We describe here the variant of fixed cluster size that sets the maximum number of elements that fits in a disk page included into a cluster, because the LCP is based on it [9], has better search performance [2], and it is more convenient to secondary memory.

Firstly, a center c is selected from the database and a bucket size b is given. c chooses its b-closest elements of the database and build the subset I, which is the answer of a b-nearest neighbor query of c in $\mathbb{U}$. Let cr_c be the distance from c to its farthest neighbor in I. The tuple (c, I, cr_c) is called a *cluster*. This process is recursively repeated with the rest of the non-clustered objects; in this case with $\mathbb{U} \setminus (I \cup \{c\})$. Finally, we have a set of centers $\mathbb{C}$ with their cluster elements and their covering radii, organized as a list.

To answer queries, the query object q is compared with all the cluster centers $\mathbb{C}$. During a range search (q, r), for each cluster with center c_i, if the distance from its c_i to the query q is larger than its covering radius cr_{c_i} plus the query radius r we can discard its whole bucket, otherwise we review it exhaustively. Formally, if $d(q, c_i) > cr_{c_i} + r$ the cluster of c_i can be completely discarded.

3 List of Clustered Permutations

As it is aforementioned, LC is a good search index but is costly to build and PBA gives a way to answer approximate similarity queries, while trades accuracy or determinism for faster searches. The simplest way to reduce the construction time of LC is to avoid distance computations. For this sake, we have two possibilities: a bigger bucket size, or using another, cheaper, way to build the index. Follow the second possibility, in [9], we propose to combine the PBA with the LC. We choose a set of permutants, where each one within this set has a double role, as permutant and as a cluster center. Besides, only the cluster centers store their permutation. This index is called *List of Clustered Permutations* (LCP) [9].

As we mention previously, when we solve a similarity query q with the standard PBA, we need to spend $|\mathbb{P}|$ evaluations of the distance d to compute the query permutation Π_q, n evaluations of the permutation distance S_ρ to compute the order induced by Π_q on $\mathbb{U}$, and $O(fn)$ distance evaluations (of d) to compare q with the fraction f of the dataset objects that are the most promising to be relevant for the query. With the LCP index, only $|\mathbb{P}|$ ($\ll n$) evaluations of the permutation distance S_ρ are needed to compare Π_q with the permutation of each cluster center. Then, some distances evaluations are needed to review non-discarded clusters. In [9] the authors shown experiments that verify the improvement of LCP over the traditional LC.

The building process of the index is done as follows: a set $\mathbb{P} = \{c_1, \ldots, c_{|\mathbb{P}|}\}$ of centers (permutants) is randomly selected, and for each database object $u \in \mathbb{U}$,

$d(u, c_i), \forall c_i \in \mathbb{P}$ is calculated. Hence, we can compute the permutations for all the objects u in the dataset $\mathbb{U}$. Then, the first center is chosen and grouped its $b = \frac{n}{|\mathbb{P}|} - 1$ most similar objects according to the permutation distance S_ρ (excluding all the cluster centers, so that no center can be inside the bucket of another center). The process continues iteratively with the rest of elements in $\mathbb{P}$ until every element in $\mathbb{U} \setminus \mathbb{P}$ is clustered. Every center c_i maintains its covering radius cr_{c_i} (that is, the distance to the farthest object in its bucket), its bucket (its subset of elements) and its permutation. All the permutations of elements in $\mathbb{U} \setminus \mathbb{P}$ are discarded; that is, the permutations of all the objects within a bucket will not be stored.

Figure 1 shows an example of LCP index for a little set of points in $\mathbb{R}^2$, where the set $\mathbb{P} = \{c_1, c_2, c_3, c_4\}$ of centers (permutants) is selected and $b = 2$ (only two points belong to each cluster). We also show the covering radii and the permutations of centers.

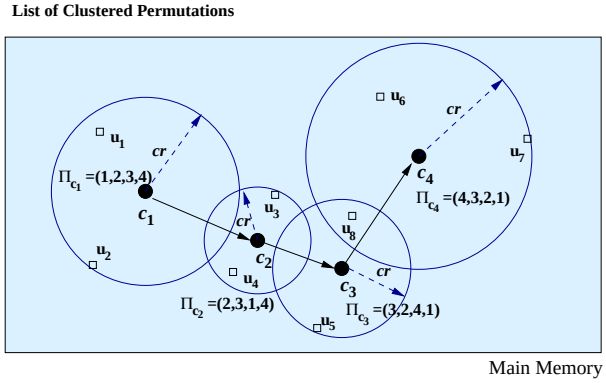


Fig. 1. An example of LCP.

Therefore, the space used for the index is $n + \mathbb{P}^2$ cells, and the construction time is $O(n|\mathbb{P}|)$ evaluations of both the space distance d and the permutation distance S_ρ. It can be noticed that the whole LCP index can be packed using only $(n+\mathbb{P}^2)log_2|\mathbb{P}|$ bits.

As it is mentioned, the standard LC discards clusters during a range search (q, r) by using the covering radii criterion. Let $d(q, c)$ be the distance between the query q and the center of the cluster c and cr_c the covering radius of center c. So, if $d(q, c) > r + cr_c$ the cluster whose center is c can be discarded.

Since the centers of LCP have permutations, a heuristic method can be introduced to discard clusters, modifying the criteria explained in [2]. In [9] authors mention that their preliminary experimental results have shown that if an object (for instance, a cluster center), and its permutation have (just) one permutant that moved far away with respect to its position inside query permutation, then this object is not relevant, so it can be discarded (and also its bucket). For example, if the permutation of the query is $(1, 2, 3, 4)$ and the permutation of the center is $(4, 1, 2, 3)$, even though most of both permutations are similar, the position shifting of permutant 4 suggests that the object can be discarded.

Basically, it is necessary to know how much could a permutant move away inside the permutation of an object. So, by using the query permutation Π_q and the range query radius r, it can be estimated how far a permutant could shift. To do that, for a pair of permutants c_i, c_j , where c_i is closer to the query q than c_j, and $d(c_j, q) - d(c_i, q) \leq r$, the method does not discard an object whose permutation has an inversion of these permutants; this is, it does not discard an object that is closer to c_j than to c_i. But, if the distance difference is larger, although permutant inversion is possible there as a big chance that the object were irrelevant, so the object can be discarded. During query process, a cluster center (and its bucket) is discarded when a permutant shifts more than tolerated.

4 List of Clustered Permutations in Secondary Memory

In order to obtain an efficient variant of LCP for secondary memory, we have to consider some important aspects of using a disk as storage. An I/O operation on disk involves three main times: the time of head positioning, lattency, and transfer time. The transfer unit of a disk is called a *disk page*. Therefore, a way of reducing times is to read/write few disk pages. One key aspect for this objective is to use as few disk pages as possible, and other is to read/write disk pages when it is strictly necessary.

Therefore, our proposal consider the design of a secondary-memory variant of LCP [9] that occupies the smallest possible amount of disk pages and it only reads/writes a disk page when it is actually necessary. Hence, we take advantage of main memory to store some information of the index and we set the size of the clusters as how many database elements fits in a disk page of size B. We called our variant as *LCP**.

The build process of LCP* is almost the same used for LCP. As we mentioned, we set the cluster size as a function of disk page size and the size of the representation of an object. Therefore, at this point LCP* is different from LCP, because b is a fixed value defined mainly by B and not as a function of the number of centers selected. On the other hand, the number of centers needed is determined as a function of the resulting size b of a cluster; that is, $|\mathbb{P}| = \frac{n}{b+1}$. Thereby, we force that each cluster fits completely in a disk page and, because of that, when we need to review the elements of a cluster we only have to read only one disk page. In each cluster only the real objects are stored. As LCP does, all the permutations of elements in $\mathbb{U} \setminus \mathbb{P}$ are discarded. Furthermore, taking advantage of main memory storage, we replicate some information of LCP* in main memory, in order to avoid reading unnecessarily a page (cluster) only to compare the query object q with the center c of a cluster and then determine its cluster is non relevant. Hence, we maintain in main memory the list of selected centers $\mathbb{P}$. For each center $c \in \mathbb{P}$ we store its covering radius, its permutation, the actual number of elements in its cluster, and the number of disk page where is stored its cluster.

Then, when we process a range query (q, r), we can determine without reading any disk page the set of candidate clusters that can be relevant to the query. This stage needs $|\mathbb{P}|$ distance computations to obtain the permutation Π_q in addition to $|\mathbb{P}|$ calculations of S_ρ distance to compare Π_q with the permutation of each center and determine which clusters have to be reviewed. Next, in order to optimize the necessary time to retrieve all the candidate clusters, we sort the number of disk pages that will be read, because it

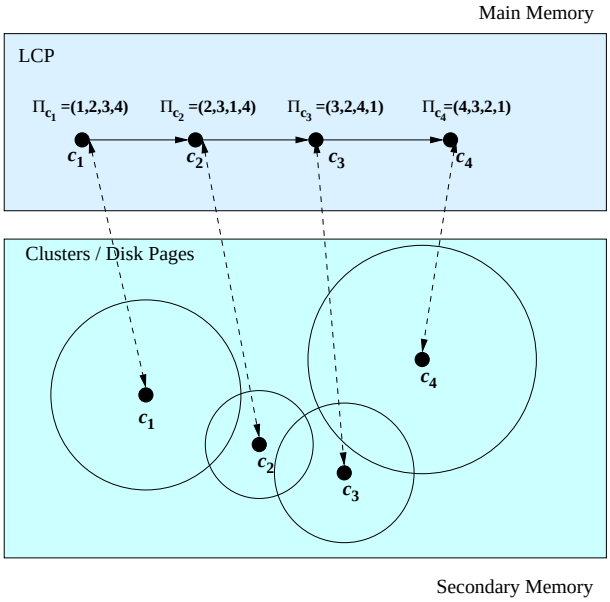


Fig. 2. A simplified example of LCP*.

is cheaper to read disk pages in a sequential way. Then, we order the elements retrieved from the clusters read and we compare q with the ordered set of elements as LCP does.

Figure 2 depicts the same example of Figure 1, but simplified because we want to show mainly the two parts of our index. As in LCP, we can use the parameter f to limit the fraction of more promising database objects that will be compared, via d, with the query q. Besides, if it appears as necessary we can add another parameter s to LCP* that limits the number of disk pages that we will read. In this case, among the list of candidate clusters we select the s more promising. Therefore, it is possible to trade accuracy with distance evaluations and I/O operations as we need, and limit to f the number of distance evaluations and/or to s the number of I/O operations.

5 Experimental Results

In order to evaluate the performance of our LCP*, we select a sample of different metric spaces from SISAP [11]: sets of synthetic vectors on the unitary cube and a real-life database. For lack of space, we only show the results obtained with the synthetic metric spaces. Since our LCP* is an approximated method, we can relax the discarding criteria by accepting bigger shifts. We tabulate these results.

Synthetic Databases: In these experiments we used synthetic databases with vectors uniformly distributed on the unitary cube. We use 100,000 points in different dimensions (5, 10, 15, and 20) under Euclidean distance. As we can precisely control the

dimensionality of the space, we use these experiments to show how much the predictive power of our technique varies with the dimensionality. We call these spaces as C5, C10, C15, and C20, considering its dimension.

In all cases, we build the index with the 90% of the database elements and we use the remaining 10%, randomly selected, as queries. So, the elements used as query objetcs are not in the index. We average the search costs of all these queries. We also evaluate the effect of using different page sizes of 4KB and 8KB, that produce different clusters sizes and number of centers.

Figure 3 illustrates the search costs of query for the synthetic spaces, measured in distance evaluations (left) and number of pages read (right), as the shifting criterion is relaxed. We also evaluate the effect of page sizes for the search performance: 4KB (up) and 8KB (down). As it can be noticed, the number of distance evaluations grows as dimension increases, but sublinearly. Besides, the number of pages read is very low, between 1 an 5 for all cases. Surprisingly, the number of pages read does not decrease as page size increases. This odd behavior can be because as page size increases cluster sizes grows, but the clusters are so big that they can not be discarded easily. As it is

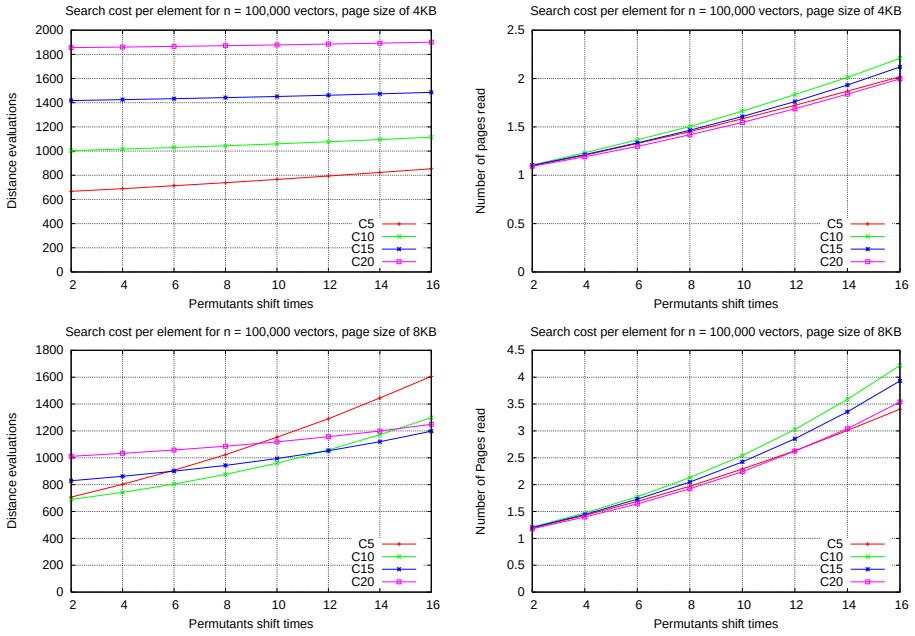


Fig. 3. LCP* Search costs for the synthetic spaces, considering page sizes of 4KB and 8KB.

aforementioned, an approximate similarity searching can obtain an inexact answer. That is, if a 1-NN query of an element $q \in \mathbb{U}$ is posed to the index, it answers with the closest element from $\mathbb{U}$ between only the elements that are actually compared with q. However, as we want to save as many distance calculations as we can, q will not be compared

against many potentially relevant elements. If the exact answer of 1-NN$(q) = \{x_1\}$, it determines the radius $r_1 = d(x_1, q)$ needed to enclose x_1 from q. An approximate answer of 1-NN(q) could obtain an element z whose $d(q, z) > r_1$.

Recall is a measure commonly used to evaluate the retrieval effectiveness of a method. It is defined as the ratio of relevant elements retrieved for a given query over the number of relevant elements for that query in the database. This measure take on values between 0 and 1. So, for each query element q the exact 1-NN$(q) = Rel(q)$ is determined with traditional LC. The approximate-1-NN$(q) = Retr(q)$ is answered with LCP* index, let be the set $Retr(q) = \{y_1\}$. It can be noticed that the approximate search will also return one element in this case, so $|Retr(q)| = |Rel(q)| = 1$. Thus, we determine the number of elements obtained which are relevant by verifying if $d(q, y_1) = r_1$. We use recall in order to analyze the retrieval effectiveness of our proposal in 1-NN queries.

Figure 4 illustrates the recall obtained for the synthetic spaces, as the shifting criterion is relaxed. We evaluate the effect of page sizes for this measure: 4KB (left) and 8KB (right). For this matter, better results are obtained with 8KB than with 4KB. For the other hand, recall decreases as dimension grows with both sizes. For example, for C5 space and 8KB of disk page size, we can obtain a recall of almost 80% only evaluating approximately 1600 distances and just reading 3.5 disk pages.

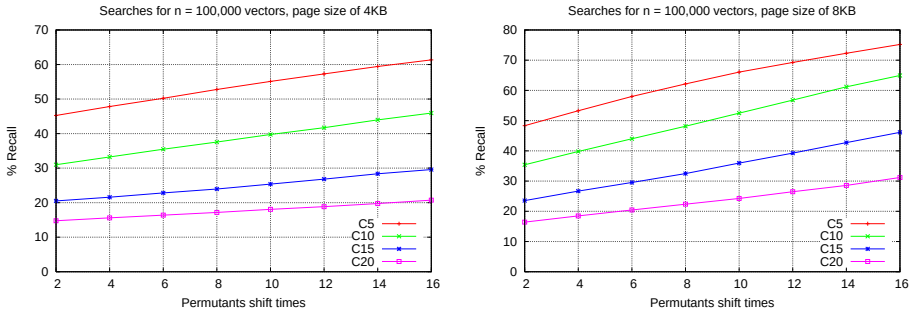


Fig. 4. Recall of LCP* for the synthetic spaces, considering page sizes of 4KB and 8KB.

6 Conclusions

We have presented a new index for approximate similarity search in secondary memory. The LCP* structure extend an in-memory approximate data structure LCP [9], that offers a good balance between construction and search time. The secondary-memory version also supports approximate searches by calculating few distances and reading very few disk pages. So, we have obtained a more practical index, because it maintain the good characteristics of LCP, but it can be applicated on massive datasets that require secondary memory storage.

As future works we have to analyze if there is a best disk page size for each metric space and to validate our results over larger real-life databases. We also have to check

how the performance is affected when we limit the number of disk pages read and the number of distance calculations, by using the parameters f and s. As it is mentioned in [9], we also want to explore the possibility of using short permutations for objects inside the clusters. This is supported by the facts that the beginning of the permutation is the most important data portion to process and that we can trade space in order to improve the recall results.

References

1. C. Böhm, S. Berchtold, and D. Keim. Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases. *ACM Computing Surveys*, 33(3):322–373, September 2001.
2. E. Chávez and G. Navarro. A compact space decomposition for effective metric indexing. *Pattern Recognition Letters*, 26(9):1363–1376, 2005.
3. E. Chávez, G. Navarro, R. Baeza-Yates, and J. Marroquín. Searching in metric spaces. *ACM Computing Surveys*, 33(3):273–321, September 2001.
4. Edgar Chávez, Karina Figueroa, and Gonzalo Navarro. Proximity searching in high dimensional spaces with a proximity preserving order. In *MICAI 2005: Advances in Artificial Intelligence, 4th Mexican International Conference on Artificial Intelligence, Monterrey, Mexico, November 14-18, 2005, Proceedings*, pages 405–414, 2005.
5. Edgar Chavez, Karina Figueroa, and Gonzalo Navarro. Effective proximity retrieval by ordering permutations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:1647–1658, 2008.
6. Edgar Chávez and Gonzalo Navarro. Probabilistic proximity search: fighting the curse of dimensionality in metric spaces. *Inf. Process. Lett.*, 85(1):39–46, January 2003.
7. Andrea Esuli. Mipai: Using the pp-index to build an efficient and scalable similarity search system. In *Second International Workshop on Similarity Search and Applications, SISAP 2009, 29-30 August 2009, Prague, Czech Republic*, pages 146–148, 2009.
8. Ronald Fagin, Ravi Kumar, and D. Sivakumar. Comparing top k lists. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '03, pages 28–36, Philadelphia, PA, USA, 2003. Society for Industrial and Applied Mathematics.
9. K. Figueroa and R. Paredes. List of clustered permutations for proximity searching. In *Proceedings of the 6th International Conference on Similarity Search and Applications (SISAP 2013)*, LNCS 8199, pages 50–58. Springer, 2013.
10. K. Figueroa, R. Paredes, and R. Rangel. Efficient group of permutants for proximity searching. In *Proceedings of the 3rd Mexican Conference on Pattern Recognition (MCPR 2011)*, volume 6718 of *LNCS*, pages 42–49. Springer, 2011.
11. Karina Figueroa, Gonzalo Navarro, and Edgar Chávez. Metric spaces library, 2007. Available at `http://www.sisap.org/Metric_Space_Library.html`.
12. Gonzalo Navarro and Nora Reyes. Dynamic list of clusters in secondary memory. In AgmaJuciMachado Traina, Jr. Traina, Caetano, and RobsonLeonardoFerreira Cordeiro, editors, *Similarity Search and Applications*, volume 8821 of *Lecture Notes in Computer Science*, pages 94–105. Springer International Publishing, 2014.
13. H. Samet. *Foundations of Multidimensional and Metric Data Structures (The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2005.
14. P. Zezula, G. Amato, V. Dohnal, and M. Batko. *Similarity Search: The Metric Space Approach*, volume 32 of *Advances in Database Systems*. Springer, 2006.

X Workshop Arquitectura, Redes y Sistemas Operativos (WARSO)

7419 | IP Queueing Delay Applied to TCP Congestion Control
Guillermo Rigotti (UNCPBA)

7434 | BATMAN Adv. Mesh network emulator
Daniel Britos (UNC), Silvia Edith Arias (UNC), Nicolás Echániz, Lucas Aimaretto (UNC), Gisela Hirschfeld (UNC)

7536 | Un enfoque multiobjetivo en redes periódicas de tiempo continúo
Diego Bosch (UCA), Benjamín Barán (UNE), Yael Uri (UNA)

7544 | Localidad estructural, criterio de división para la ejecución de redes de Petri no autónomas en IP-Core
Orlando Micolini (UNC), Marcelo Cebollada y Verdaguer (UNC), Luis O. Ventre (UNC)

7589 | Módulo gráfico de un Visualizador de Estructuras de un Sistema Operativo Educativo a través de GDB-Stub
Graciela De Luca (UNLaM), Martín Cortina (UNLaM), Nicacnor B. Casas (UNLaM), Esteban Carnuccio (UNLaM), Sebastián Barillaro (UNLaM), Daniel Alberto Giulianelli (UNLaM), Pablo Barboza Carvalho (UNLaM)

7601 | Integración de un periscopio militar a un simulador de realidad virtual para entrenamiento táctico
José Marone (UNCPBA), Marcelo A. Tosini (UNCPBA)

7604 | Evaluación de la calidad de la experiencia para servicios de datos móviles
Brenda R. Quiñonez (UNA), Luis F. Fernández (UNA), Diego Pinto (UNA), Carlos Núñez (UNA), María E. García (UNA)

7618 | Diseño e Implementación de Red Académica Experimental aplicando IPv6
Gustavo Mercado (UTN-FRM), Javier Gitto (UTN-FRM), Rubén Soria (CONICET), Carlos Taffernaberry (UTN-FRM), Cristian Perez Monte (UTN-FRM), Marcela Orbiscay (CONICET), Sebastián Tobar (UTN-FRM), Ana Diedrichs (UTN-FRM)

7638 | Estrategias de control topológico para reducir la interferencia en Redes Inalámbricas Multisalto
Nelson R. Rodriguez (UNSJ), María Antonia Murazzo (UNSJ), Olinda Edilma Gagliardi (UNSL)

7644 | Uso de dispositivos móviles en zonas rurales aisladas abastecidas con energía solar fotovoltaica
Sergio Rocabado (UNSa), Javier Diaz (UNLP), Carlos Cadena (UNSa)

7650 | Análisis de despliegue de una IaaS utilizando Openstack
Brian Galarza (UNAJ), Zaccardi Gonzalo (UNAJ), Diego Encinas (UNLP), Martin Morales (UNAJ)

IP Queueing Delay Applied to TCP Congestion Control

Guillermo Rigotti
ISISTAN, Fac. de Ciencias Exactas
Universidad Nacional del Centro de la Pcia. de Bs. As.
grigotti@exa.unicen.edu.ar

Abstract

TCPś congestion control was traditionally based on packet losses. This approach leads to an inefficient bandwidth use, because losses must be produced to infer network load. In last years, another approaches have been considered to infer that load, mainly the estimation of sender-receiver delay, known as "one way delay" (OWD). The estimation of this delay enables the sender anticipate losses and lower its sending rate before they occur. A known problem is to accurately estimate OWD, because it requires synchronization between sender and receiver. Various alternatives exist to achieve synchronization (NTP, GPS), but these are not applicable to TCP's flows due to the significant number of flows and because they mainly reside on users' equipment. Another approach that has been recently incorporated in TCP's flow control mechanisms, consists in infer OWD from the RTT value by means of some heuristic. However, this approach has lead to develop complex heuristics that produce inaccurate results, due mainly to the assymetry in the paths and the variability of network load.
In this paper a simple mechanism to estimate OWD is proposed. It is based in the cooperation of the network layer, and consists in obtain OWD as the sum of queueing delays experienced by packets in each router along the path from sender toward receiver.
Keywords: Congestion control, TCP, OWD

1 Introduction

TCP's [1] congestion control mechanisms aim that each flow obtains the higher possible bandwidth, without compete unfairly with another flows (TCP friendly). To meet this objective, each flow has to adjust its sending rate according the network load he perceives.
Initially, TCP congestion control mechanisms (TAHOE, RENO), relied completely on packet losses to adapt its sending rate. Additionally, losses affect all flows that share the congested router, causing that those flows synchronize, generating cycles composed of congestion periods followed by periods of sub-use of network capacity.
In 1989 Jain proposed the use of round trip time (RTT) to anticipate congestion events. This approach enables the sender anticipate a congestion event and to

diminish its sending rate, avoiding packet loss [2]. Some of the most currently used congestion control mechanisms such as TCP Vegas and TCP Fast adopt this approach in addition to packet loss detection.
As described below, the network load is closely related to OWD. Various methods to estimate its value have been proposed, mainly: some RTT-based heuristics, and the use of timestamps (TCP timestamp option) plus synchronization between sender and receiver.
RTT estimation is a simple process because it does not require clocks synchronization. It consists in finding out the elapsed time between sending a packet and receiving its acknowledgement (with certain considerations) [9]. RTT's value is of fundamental importance and is employed to derive parameters such as retransmission timeout (RTO) [3].
However, its value is not useful to estimate the delay from sender to receiver, mainly due to possible asymmetry between sender-receiver and receiver-sender paths. This asymmetry has three main components: 1 - asymmetry in transmission links, for example optic fiber from sender to receiver and satellite from receiver to sender; 2 - asymmetric paths due to routing protocols (acknowledgement packets traverse different routers than data packets); 3 - asymmetry in load, due to transient states of the network.
Due to the low correlation between RTT and OWD [4], the various proposed heuristics to infer OWD based on RTT do not produce accurate results, and in most cases are too complex to be executed on line with the arrival of packets.
The approaches that try to estimate OWD experienced by packets in function of the difference between the sending time at the sender and the receiving time at the receiver, have to resolve the problem of clocks synchronization between sender and receiver. There are two alternatives to achieve synchronization: Network Time Protocol (NTP) and Global Positioning System (GPS). These are suitable for a variety of applications such as VoIP, but cannot be applied satisfactorily to TCP flows, due to its significant number, and its location (user equipment generally unable to GPS).
In this paper we propose an approach to estimate network load. It is based on the buffer occupancy in routers involved in the sender-receiver path. The number of packets waiting for transmission in routers is directly related to network load. To implement the proposed alternative, we define and implement a mechanism at network level (IP) that enables to register the load in each router, from sender to receiver. This is implemented as a new IP option, referred to as "QD option". This is complemented by a second mechanism at transport level (TCP), that enables that information collected by IP be accessible to the congestion control mechanism in use.
These mechanisms are implemented as follows: the former as a new IPv4 [5] option and a new option in the IPv6 [6] hop-by-hop header; the last as a backward compatible modification of TCP timestamp option. Although our proposal implies to carry out some process in each router, we can state that it is negligible (a few instructions in Linux kernel). Furthermore, no additional state needs to be created or maintained in routers.
The rest of the paper organized as follows. In Section 2 related work is presented; in Section 3 different components of delay are analyzed and a way to quantify it is proposed. Section 4 explains how the proposed mechanisms are incorporated in IP and TCP levels, while Section 5 presents some aspects of the implementation in Linux. Section 6 describes simulations carried out to conduct

a first evaluation of our proposal. Section 7 presents our conclusions. Finally, Section 8 refers to relevant bibliography.

2 Related Work

The mechanism proposed in this paper can be characterized as follows; 1- It does not generate additional traffic. It should be noted however, that a minimal overhead is generated (only one IP option) and is only present in certain datagrams. 2- Requires cooperation of routers to compute the estimated delay. It should be noted however that does not require state in the routers. 3- Our method is not based on clocks' synchronization as those using NTP, GPS or time derived from CDMA (radio links).
Among similar work, we can quote [7] . This approach aims to estimate delays in wireless ad-hoc networks to improve TCP efficiency. Unlike our proposal, it maintains generic state (valid for any TCP stream) on routers. This state is specific to the protocol. The delay is calculated through the cooperation of the involved routers. Each router maintain two variables: Q, that represents the exponential average of the queuing delay experienced by packets traversing the router, and T, that represents the exponential average of the queuing delay experienced by the head-of-line packet. For every packet that a router forwards, the sum Q + T is stamped if that of the packet is smaller.
In [8] ,the authors propose an algorithm to estimate characteristics of OWD without the need of synchronization. A difference of our approach, they use ICMP probe packets with two different lengths, sent from two sources to a target. The target dont need any additional functionality. To carry the necessary information, ID field of the IP datagrams is used.

3 Estimation of network load

Our goal is to measure the network load. This is directly related to the size of transmission queues in routers belonging to the path from sender toward receiver. To compute this value, various parameters could be considered, such as number of packet waiting for transmission, percentage of use of queues, transmission delay, etc,
Although any of them could be selected, we choose the last (delay) due to the following: 1- It is a generic measure, suitable to be added by each router. 2- Transmission delay can be used to calculate some commonly used parameters, such as link asimmetry, variations on delay, etc. 3-Taking in account that transmission delay is a frequently used parameter in congestion control mechanisms; our measure could be applied to those mechanisms in place of the original ones (for example RTT). This could be useful to compare their performance with respect to their original parameters.
Mechanisms that depend on network level support, require that all involved routers be able to collaborate, to achieve optimal results .If some routers are not able to do so, it is likely that wrong or distorted results be obtained. In cases such as ECN, where the result may depend on a single router, erroneous results can be obtained (for example, in the particular case that the congested router does not implement ECN). In the case of transmitter-receiver delay, the

situation is different because the value does not depend on a single router. If one or more routers are unable to register delays, the obtained result gets distorted, but possibly can be useful. The above situation may arise for example when certain routers do not implement the QD option or within a MPLS domain, where routers (acting as switches), do not have access to the IP level. The amount of distortion depends, among other factors, on the proportion of routers that do not cooperate.In order to quantify that distortion, simulations described in Section 6 were carried out.
Another aspect to consider is the overhead incurred in routers to estimate the queueing delay experienced by packets; this is discussed in Section 5.
To relate the queuing delay with other components of the total sender-receiver delay, consider a sending host A and a receiving host B. If we indicate the intermediate routers as $R_{\underset{\rightarrow}{xy},i}$, where the first subscript represents the direction (from x to y) and the second the order of the router along the way, we can express the paths from A to B and B to A as follows:

$$R_{\underset{\rightarrow}{ab},1},\ \ldots,R_{\underset{\rightarrow}{ab},i},\ \ldots,R_{\underset{\rightarrow}{ab},n} \quad and \quad R_{\underset{\rightarrow}{ba},1},\ \ldots,R_{\underset{\rightarrow}{ba},i},\ \ldots,\ R_{\underset{\rightarrow}{ba},m}$$

If we consider the total delay experienced by a packet sent from A toward B, as mentioned in Section 1, we can express this delay as the sum of several factors, as detailed below: Let P the number of bits in a packet, $Vt_{(R_x,R_y)}$ the bandwidth of the link that connects adjacent routers R_x and R_y, $dp_{(R_x,R_y)}$ the propagation delay of the link connecting adjacent routers R_x and R_y, $Q_{(R_x,R_y)}$ the number of packets queued to be sent from the router R_x to the router R_y and PR_x the processing time of the packet in the router R_x (delay since the packet is received by the router until it is ready for shipping), we can express the delay (OWD) from A toward B (and analogously from B toward A) as follows:

$$\begin{aligned}
&OWD_{\underset{\rightarrow}{ab}} = \\
&P/Vt_{(A,R_{\underset{\rightarrow}{ab},1})} + dp_{(A,R_{\underset{\rightarrow}{ab},1})} + && (t)\\
&\sum_{i=1}^{n-1} P/Vt_{(R_{(\underset{\rightarrow}{ab},i)},R_{\underset{\rightarrow}{ab},i+1})} + dp_{(R_{\underset{\rightarrow}{ab},i},R_{\underset{\rightarrow}{ab},i+1})} + && (t)\\
&P/Vt_{(R_{(\underset{\rightarrow}{ab},n)},B)} + dp_{(R_{(\underset{\rightarrow}{ab},n)},B)} + && (t)\\
&\sum_{i=1}^{n} PR_{(R_{\underset{\rightarrow}{ab},i})} + && (p)\\
&\sum_{i=1}^{n-1} Q_{(R_{\underset{\rightarrow}{ab},i},R_{\underset{\rightarrow}{ab},i+1})} * P/Vt_{(R_{\underset{\rightarrow}{ab},i},R_{\underset{\rightarrow}{ab},i+1})} + && (q)\\
&Q_{(R_{\underset{\rightarrow}{ab},n},B)} * P/Vt_{(R_{\underset{\rightarrow}{ab},n},R_B)} && (q)
\end{aligned}$$

.

In the above equation we distinguish three types of components that integrate the delay: those identified with references (t) -transmission time, (p) -time of processing- and (q) time spent in transmission queues.
Those identified by (t), represent the time taken to send the packet from host A to the first router, from each router to the following, and from the last router to host B; This transmission time depends only on the characteristics of the links traversed by the packet and kept constant while the route do not change; It is also independent of the network load.

The term identified by (p) represents the packet processing time in each of the routers: from the arrival of the packet until it is queued for transmission in the corresponding output interface. This time depends on several factors, such as hardware, operating system, equipment load, etc. It may be considered independent of the network load.
The components of the delay (t) and (p) depend on the path followed by the packages and change when the routing decision is modified.
Finally, the components identified by (q) correspond to queuing delays in each of the routers: these delays are directly dependent on the number of packets waiting to be transmitted by the corresponding output interface, and provide an accurate measure of the load on the network.

4 Functions in routers and hosts

As mentioned above, the proposed mechanism is based on defining a new IP option - queueing delay:QD - (see Section 4.1) that records the delay experienced by the datagram in the queues of routers, from sender toward receiver. A datagram that carries this option is modified in each router, which adds its own delay to the delay carried in the IP option.
TCP sender decides according to the congestion control algorithm used, when measure the delay toward the receiving side (in all segments carrying data, at certain intervals of time, etc.). To do so, it asks to IP generate a QD option initialized to zero in the corresponding datagram. When the datagram reaches its destination, the QD option contains the accumulated queuing delay recorded by routers, which is made accessible to the TCP receiver. The receiver should immediately send back that information, so it can be known by the sender in a period equal to RTT. To achieve this, the delay is sent into the next segment. Timely receipt of the delay is critical to correctly adjust congestion control parameters at the sender side

4.1 Changes to IP level

The amendments necessary at the IP level consist of the addition of the option "queuing delay" (QD) . It is described below for IP versions 4 and 6.
According to the IPv6 specification [6], a new type of option is defined (QD) within the hop by hop extension header. This header (and therefore the QD option) is processed by all routers. The presence of this header is specified in the IPv6 header, with a next-header value of zero. The hop-by-hop header, including their fields and the QD option, are shown in Fig.1.
Like the other options, QD is encoded in TLV format. It consists of the following fields:
type: the two first bits set to 0 indicate that if a router does not recognize the option, he must process the next header; the third bit, set to 1, indicating that the value of the option (cumulative delay) can be modified by routers; the 5 bits remaining indicate the type of option (QD);
length: set to 4, indicating data length,
value: the last field, which corresponds to data, composed of 32 bits containing the cumulative queuing delay experienced by the packet.
The option must be aligned to 8n + 2, according to the IPv6 specification.

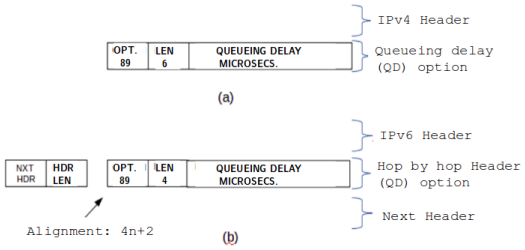


Figure 1: Queueing delay option (a):IPv4 (b):IPv6

According to IPv4 specification [5], we define the option "queuing delay" (QD) with the following fields:
1-copy: set to 0 (do not copy the option in fragments),2-class: set to 2 (debugging and measurement),3-length: set to 8, indicating that 4 bytes are reserved to hold the value of the option, that represents the accumulated queuing delay expressed in microseconds.
The selected option number is 25, still unused. Thus, the value of the first byte is 89 (decimal). value: finally there are 4 bytes corresponding to the value of the option.

4.2 Changes to TCP level

The TCP sender must be able to ask IP generate and send a QD option. On the receiving side, TCP should be able to access the value of the QD option. The interaction between TCP and IP is performed through kernel resident code. Additionally, in the connection-establishment phase, both TCP sides should negotiate the use of QD option in order they be able to exchange delay information in the connection-established phase.
To perform these two functions an alternative use of the timestamp option is proposed. This is compatible with RTTM and PAWS [9].
The initial negotiation is based on the fact that the value of tsecr (timestamp option) is set to zero in the segment that carries SYN flag. The interchange of delay information carried out during the connection-established phase, is based on the fact that not every segment have to carry a timestamp value in order to perform traditional processes. We propose a mixed use of tsecr field: in some segments, it contains time information; in others, information about delays.

At the beginning of the connection, each TCP checks whether he is in position to announce its ability to process the IP QD option; the following conditions must be satisfied: TCP has been configured to use QD, and its local IP must support the QD option. The side that initiates the connection, if able to use the option, announces this fact in the SYN segment sent for establish the connection: the echo field (tsecr) of the timestamp option, instead of being sent with zero value, is sent with its first two bits set to 1. The side receiving the connection request, if able to accept QD processing, send a modified value in the echoed tsval (tsecr field of its generated timestamp option): the new value is obtained xoring the two most significant bits of the original one (tsval field of the received timestamp option) with "11". This result is sent in tsecr field.

In this way both TCP sides agree to the use of QD option. This negotiation is also true in the case of simultaneous connections.

As previously mentioned, contrary to the assertion in [10], it has been shown that taking multiple samples of RTT does not improve the calculation of RTO [11]. In [3] the new way to calculate RTO based on fewer RTT samples is explained. Therefore, it is possible to use the timestamp option to carry the queuing delay besides time information.
The sending rate of QD samples from side A to side B, is determined by the congestion control algorithm used in A. When B receives a datagram carrying the QD option, gets its value and immediately sends it to A in the timestamp option corresponding to the next segment to be sent. This ensures that the delay is received within an interval equal to RTT from the time it is requested.
To sum up, the timestamp option returned by B to A has the following values:
tsrec: the value received from A into the tsval field.
tsval: delay received in IP QD option of the datagram encapsulating the segment sent by A.
To identify a timestamp option that carries a value of delay instead of the usual, the side that originates the request must save the tsval value contained in its timestamp option. In this way he is able to match the response with his request.
The receiver side must not use such timestamp option to calculate the RTO. In addition, he must have provisions to avoid generating two segments with the same tsval.

5 Implementation

The processing carried out by TCP is executed at the beginning of the connection and then by each segment that carries the QD option. Despite that these processes are not critical, its implementation should be simple and efficient.
The process of QD option in routers (at IP level) is critical because the code must be executed in real time with the arrival of each packet containing that option.
We considered the following aspects in order to achieve efficiency: amount of instructions, places where kernel code needs to be modified, and state that should be kept for each datagram, from its arrival until it is forwarded.
In our case, the implementation was done in Linux [12], kernel 3.14.5. In this context, the following alternatives were considered:
1-Record the send and receipt times of a datagram. Sum the difference of these values to the QD option value, and
2-Estimate queueing delay based on the number of packets to be transmitted before the datagram be sent (number of packets in the transmission queue). Sum this value to that of the QD option.
Despite the first alternative is more simple, we choose the second due the following reasons:
The first choice requires maintain state for each datagram: when a datagram is received, its arrival time has to be registered and kept in the associated sk_buffer. If we want to avoid storing state (sk_buffer), it would be necessary to subtract the arrival time to the QD option value; then add the departure time. In this way, QD value has to be considered signed, because routers are

not synchronized. Moreover, it should be noted that in this way, the packet processing time is included in QD value. Additionally, if mechanisms such as Generic Segmentation Offload (GSO) or TCP Segmentation Offload (TSO) are used, the second step of the process (obtain sending time and modify QD value) must be inserted in the appropriate code (belonging to GSO or TSO), which is not always possible because hardware implementations.
The second option was choosen due to its efficiency and simplicity. It requires only one point of modification in kernel code: just before the datagram is queued to be transmited. The process consists in estimate the time that transmission of previous packets will demand, and adding this value to that of QD option. The estimated value is obtained multiplying the number of packets queued by the estimated packet transmission time.
The advantages of the last choice over the first are the following: 1-only one modification to kernel code is needed, 2-do not require keep state relative to each datagram, 3-do not include processing time in the estimated queueing delay, 4-does not depend on the accuracy of each computer's clock.
One aspect to consider is how to estimate the transmission time of the previously queued datagrams. The proposed solution is to calculate that value as a function of the number of bytes (or datagrams) queued, the bandwidth, and the MTU of the transmission link. In our case, we consider datagrams of same length as the MTU. As in the first case, we must consider the use of GSO or TSO. If these Mechanisms are active, the amount of bytes or datagrams previously queued must be requested to them.

6 Simulation

Given the impossibility of checking the behavior of our proposal with real machines (physical or virtual) because it is necessary to involve a considerable number of routers, simulations were performed. We use Network Simulator 2 (ns-2) [13], version 2.3.5 running on Linux with kernel 13.14.5.
In a first step the simulation was used to check how is affected the measurement of delay queueing, by the fact that some of the routers do not implement the QD option. At a later stage, the developed code will be used to compare the queuing delay that we obtain, with other OWD estimations carried out by some algorithms of TCP congestion control (eg Vegas or Fast).
The simulated topology is composed of thirty routers and hosts running applications which generate network load. Considering topology, paths between hosts used to estimate delay are symmetric; asymmetry is obtained varying the load generated in the network. This load is originated by multiple UDP flows with CBR applications. In all cases the location of applications and their sending rates and duration were generated randomly, with periods of asymmetry in both directions.
Results are displayed in Figure 2. The left side shows the relationship between the round trip delay (RTT) taken by the source host, the queuing delay perceived by the destination host, and the OWD from origin host toward destination host. The value of RTT is taken by the source host in standard way, queuing delay is taken according to what is proposed in this paper, and the OWD is taken as a function of simulation time and serves only as a reference
It can be seen the correspondence between the value of OWD and the value

of the queuing delay, regardless of the asymmetry in both directions, which is observed between 27 and 29 seconds of simulation.
An important aspect to evaluate, is how queueing-delay estimation is affected by the fact that some routers are not able to process the QD option; this can be seen on the right side of the figure, where queuing-delay is shown for cases in which 100%, 75% and 50% of the routers are capable to process the QD option. Estimation accuracy decreases with decreasing the number of routers, but remains proportional, even in periods of asymmetry.This is because the resulting delay is the aggregate of those taken by all routers.
On the other hand, when the number of routers capable of measuring the delay increases, the value obtained tends to the actual value (OWD).
Moreover, it is observed that during periods of asymmetry caused by variations in load, the estimated value is stable, which does not happen with OWD estimates based on RTT.

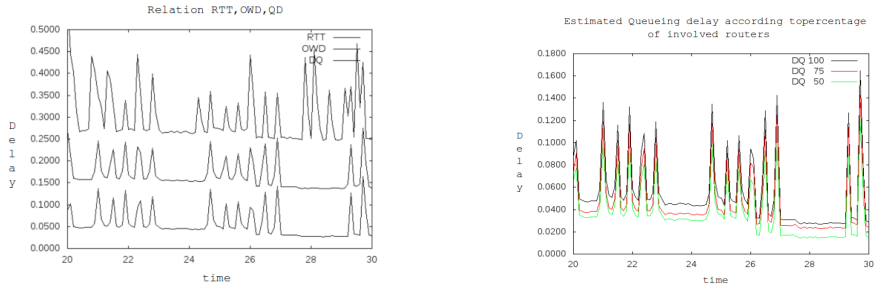


Figure 2: Relation RTT-OWD-QD (left). QD values (right).

7 Conclusions

Prevent Internet overloading depends largely on transport level protocols. A key role is performed by TCP, which is the most used protocol and therefore which generates most of network traffic.
TCP congestion control algorithms are responsible for avoiding network congestion, and so they have been under constant research and improvement. In recent years, it has begun to use the delay between transmitter and receiver as a parameter to adjust the load introduced into the network. It has been used alone or in combination with packet loss. Similarly, IP level mechanisms working in collaboration with TCP (ECN, AQM, etc.) have been improved to allow a more effective reaction against losses.
This paper has proposed a mechanism at IP level, that helps TCP assess more accurately the transmitter receiver delay caused by network load, and hence react timely to changes in network load, avoiding packet loss. To carry out the proposed functions both IP and TCP have been modified. In both cases the changes were minimal, and code efficiency was taken into account. Simulations were performed that demonstrate that our approach works adequately in asymmetrical networks and in situations where a number of routers have not implemented the QD option.
The next steps to complete the validation of the proposal are the use of queuing

delay in different congestion control algorithms employed by TCP (including Vegas and FAST). In order to materialize this step we will implement the congestion control algorithms as Linux kernel modules; this approach allows separation of congestion control of the rest of the TCP code, simplifying the code. For the simulation, the characteristic of ns-2, that allows a quick port of Linux kernel modules into the simulator, will be exploited.

References

[1] Postel, J., "Transmission Control Protocol - DARPA Internet Program Protocol Specification", RFC 793, DARPA, September 1981.

[2] Jain, R., "A Delay Based Approach for Congestion Avoidance in Interconnected Heterogeneous Computer Networks", Computer Communications Review, ACM SIGCOMM, pp. 56-71, October 1989.

[3] Paxson, V; Allman, M.; Chu, J.; M. Sargent, "Computing TCP's Retransmission Timer", RFC 6298, June 2011.

[4] Pathak1, Himabindu Pucha1, Ying Zhang2, Y. Charlie Hu1, Z. Morley Mao, "A Measurement Study of Internet Delay Asymmetry", (https://web.eecs.umich.edu/-zmao/Papers/pam08-owd.pdf)

[5] "Internet Protocol - DARPA Internet Program Protocol Specification", RFC 793, DARPA, September 1981.

[6] S. Deering R. Hinden, "Internet Protocol, Version 6 (IPv6):Specification", Network Working Group, IETF, RFC 2460, December 1998

[7] K. Sundaresan et al., ATP: A Reliable TransportProtocol for ad hoc Networks , ACM Intl. Symp. Mobile Ad Hoc Net. and Comp. (MOBI HOC), 2003, pp. 64 75.

[8] Antonio.de.Rochaet al. " A Non-cooperative Active Measurement Technique for Estimating the Average and Variance of the One-Way Delay", Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet Lecture Notes in Computer Science Volume 4479, 2007, pp 1084-1095

[9] D. Borman, B. Braden, V. Jacobson, R. Scheffenegger, Ed."TCP Extensions for High Performance",RFC 7323,September 2014

[10] Jacobson, V.;Braden, B.; D. Borman, "TCP Extensions for High Performance", RFC 1323, May 1992

[11] Allman, M.; V. Paxson, "On Estimating End-to-End Network Path Properties", Proceedings of the ACM SIGCOMM, Technical Symposium, Cambridge, September 1999

[12] http://www.kernel.org

[13] S. McCanne; S. Floyd, "ns-LBNL Network Simulator 2", http://www.isi.edu/nsnam/ns/

BATMAN Adv. Mesh network emulator

José Daniel Britos[1], Silvia Arias[1], Nicolás Echániz[3], Guido Iribarren[3], Lucas Aimaretto[1], Gisela Hirschfeld[2]

[1] Laboratorio de Redes y Comunicaciones (LaRyC) Facultad de Ciencias Exactas, Físicas y Naturales, Universidad Nacional de Córdoba Av. Vélez Sarsfield 1611 - Ciudad Universitaria - CP: X5016GCA Córdoba.
[2] Departamento Universitario de Informática, Universidad Nacional de Córdoba, Valparaíso s/n - Ciudad Universitaria - CP: X5016GCA
[3] Asociación Civil AlterMundi, José de la Quintana, Córdoba.

Resumen We introduce a new network emulator environment, developed by our research group, called the BAMNE. Our emulator is designed specifically to allow working with BATMAN Adv. mesh protocols. This mesh network emulator facilitates doing tests with BATMAN Adv. protocol and evaluate and debug mesh networks. The emulated wireless equipment runs in virtual machines using VirtualBox, and the wireless links are simulated with Vde-switch. Vde-switch allows simulating impediments in the link transmission like loss of bits, packet loss, delay. To construct the emulation environment, python language was used.

1. Introduction

The purpose of this network emulator is to test, evaluate and debug mesh network protocols like the BATMAN Adv. protocol. This network emulator is a front end for VirtualBox OpenWrt machines connected through a Vde-switch and Wirefilter, emulating a wireless link. This is able to emulate delays, noise factors, channel bandwidth and packet loss on virtual wireless channel. The front end is written in python with "pygtk" graphics user interface (GUI). The python program doesn't introduce latency overhead in emulation because the only purpose of this python software is to set up the emulation environment, parameters and monitor the OpenWrt machines with Simple Network Management Protocol (SNMP), showing in the main screen transmitted packets for each interface, and originators details for BATMAN Adv. protocol. The Vde-switches have tap interfaces connected with the host machine, this allows to monitor the packet traffic with the Wireshark program [13]. The *eth0* interface of the OpenWrt machines are connected to the host via the *Vboxnet* interface of the host, in this way it is possible to access the OpenWrt console for management purpose. This paper is structured as follows: we start in Section 2 with a background on the BATMAN Adv. protocols. Section 3 describes the architecture of the emulation system. This is followed by Section 4 which describes the use of the emulator. Finally conclusions are drawn in Section 5.

2. BATMAN Adv.

BATMAN Adv. [12] is a proactive routing protocol for Wireless Mesh Networks (WMNs). It uses a distance-vector approach and a routing metric which incorporates the reliability of the radio links. Despite being developed and publicly available since 2006, BATMAN, especially its newer BATMAN Adv. variant, has received sparse attention in the scientific community. BATMAN is a simple and robust algorithm for establishing multi-hop routes in ad-hoc networks. As explained by Johnson, D., et al. [15] BATMAN does not maintain tables with full routes to the destination, instead each node along the route only maintains the information about the next link through which the node can find the best route. The objective is to maximize the probability of delivering a message. BATMAN does not check the quality of each link, it checks its existence. The protocol does these checks by periodically broadcasting *hello* packets to its neighbours, these packets are known as originator messages (OGM). Broadcasting is when a single source sends messages to all available nodes in the broadcast domain/network. This is in contrast to unicast where a node sends messages to one specific node in the network. The structure of the OGM packet periodically sent is presented here:

- Originator address
- Sending node address: This is changed by receiving nodes and then the packet is re-broadcasted
- Unique sequence number: The sequence number is used to check the concurrency of the message
- Bidirectional link flag: Used when the OGM packet received is its own and the sender is someone else
- Time to live (TTL)

When a node receives an OGM packet there are two possibilities: either the originator is, or is not already in its routing table. If the originator is not in the routing table, then a new entry is made for it and the sender node is added as a one hop neighbour to it and its count incremented. If the originator is already in the routing table but the sender is new, the sender is added as a one hop neighbour to the originator and count incremented. If the originator is in the routing table and the sender is not new, the sender's count is incremented. The count is the amount of received OGMs packets of an originator through a specific one hop neighbour. The links are compared in terms of the number of originator messages that have been received within the current sliding window; this value is called the transmission quality (TQ) and is the routing metric used by BATMAN. The sliding window is a fixed value that defines a range of the unique sequence numbers seen in each OGM packet sent by a node [11]. BATMAN advanced (BATMAN Adv.) [2] only uses the MAC address for addressing it neighbours. The result of working in layer 2 is that BATMAN Adv. is able to emulate an Ethernet bridge, so that all nodes appear to be connected by a direct link. This causes all protocols above layer two be unaware of multi hop links. BATMAN's routing technique causes low processing and traffic cost. This

makes it an attractive option for use on devices with scarce resources. In this work we focus on BATMAN Adv. routers that have small processors such as the MP.

3. Architecture

The emulator architecture is basically composed of two elements: Nodes (OpenWrt) and Links (Wirefilter) [3] as shown in the figure 1. Each node is composed as shown in figure 2, a complex diagram that has the following elements:

- OpenWrt, trunk version for x86 with minimal modifications (see below)
- VirtualBox (unmodified), the version must support Vde-switch [5].
- Vde_switch must be able to run two instances, in order to support 2.4 GHz and 5.0 GHz networks interface. Thus, Vde-switch is patched with *colourful*, see below [14].

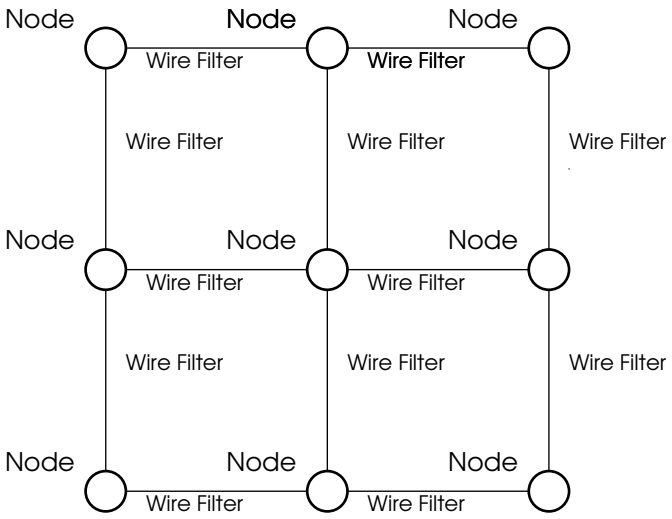


Figura 1. Network architecture

3.1. OpenWrt

A standard OpenWrt can be downloaded and configured for X86 [1]. Once that the virtual machine is running some packages must be downloaded (ip, snmpd, tcpdump, netcat, kmod-batman-adv, batctl). For an automatic configuration of the network interfaces and devices, a boot script must be used, such as the following script (saved as */files/etc/rc.local*) in your OpenWrt.

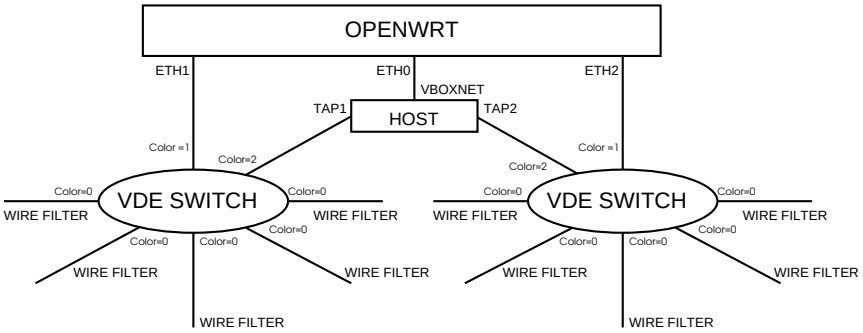


Figura 2. Node Diagram

```
#!/bin/sh
# pass IP config trough ethernet mac address
RED=$(ifconfig eth1 | sed '1,1!d' | sed 's/.*HWaddr //'| sed
's/.\{11\}://'| sed 's/.\{5\}$//')
NUM=$(ifconfig eth1 | sed '1,1!d' | sed 's/.*HWaddr //'| sed
's/.*://'| sed 's/[\n\ ].*//')
ip link delete eth0
ip addr add 192.168.100.$NUM/24 dev eth0
ip link set dev eth1 mtu 1532 up
ip link set dev eth2 mtu 1532 up
batctl -m bat0 interface add eth1
batctl -m bat0 interface add eth2
ip addr add 192.168.$RED.$NUM/24 dev bat0
ip link set dev bat0 address 90:$NUM:$NUM:$NUM:$NUM:$NUM
ip link set dev bat0 up
batctl -m bat0 originators
```

In VirtualBox it is difficult to learn the IP address for the interfaces of the virtual machines; this is solved by setting the MAC address in VirtualBox and in the *"rc.local"* script read the MAC address to set an according IP address for the interfaces.

3.2. SNMPD

After the "SNMP" is installed in the OpenWrt machine, we proceed to setup the MIB for BATMAN Adv.. To add custom records to the BATMAN Adv. MIB a series of shell scripts are run and return their output to *stdout*, which is captured by SNMP [6]. To request originators table the next script was made: Script name *batctl_o.sh* (for originator list)

```
#!/bin/sh
BAT=$(batctl o | sed -n 's/^\(..:..:..:..:..:..\).*/\1/p')
echo $BAT
```

Then add entries for the SNMPD configuration file */etc/snmp/snmpd.conf* using the *uci* command in a terminal.

```
uci add snmpd exec
uci set snmpd.@exec[-1].name=.1.3.6.1.4.1.32.1.1
uci set snmpd.@exec[-1].prog=batctlo
uci set snmpd.@exec[-1].args=/batctl_o.sh
uci commit snmpd
/etc/init.d/snmpd restart
```

And append the following line to the file */etc/snmp/snmp.conf*

```
exec  .1.3.6.1.4.1.32.1.1 batctl_o /batctl_o.sh
```

From the host the snmp can be tested with the followings command.

```
$ snmpget -v 1 -c public 192.168.100.11 iso.3.6.1.4.1.32.1.1.101
.1iso.3.6.1.4.1.32.1.1.101.1 = STRING: "80:03:00:00:07:41 80:03:
00:00:07:31 80:02:00:00:07:31 80:02:00:00:07:21 80:03:00:00:07:21"
```

We repeat the same for the next SNMP commands.
In the GitHub *(https://github.com/dbritos/Network-mesh-emulator/blob/ master/openwrt.ova)* repository there is a fully configured virtual machine. Download openwrt.ova in VirtualBox go to:

```
File menu -> Import Appliance
```

3.3. Ip assignations in OpenWrt

For assigning the IP address to the VM, first a MAC address is assigned to the VM. Each VM has three interfaces: *nic1*, *nic2* and *nic3*. This interfaces in OpenWrt appear as *eth0*, *eth1* and *eth2*.

```
nic1 (eth0) mac 80:01:00:00:07 + nodenumber(nn)
nic2 (eth1) mac 80:02:00:00:07 + nodenumber(nn) the 2 for 2.4GHz)
nic3 (eth2) mac 80:05:00:00:07 + nodenumber(nn) the 5 for 5.0GHz)
```

For configuring the VM with this MAC address the following commands are used:

```
VBoxManage modifyvm openwrtnn --nic1 generic --nicgenericdrv1 VDE
--nicproperty1 network=/tmp/c24GHznn[2] --macaddress1 8001000007nn
VBoxManage modifyvm openwrtnn --nic2 generic --nicgenericdrv2 VDE
--nicproperty2 network=/tmp/c24GHznn[2] --macaddress2 8001000007nn
VBoxManage modifyvm openwrtnn --nic3 generic --nicgenericdrv3 VDE
--nicproperty3 network=/tmp/c24GHznn[2] --macaddress3 8001000007nn
```

Where: *nn* is the Node number.
The script in OpenWrt in */etc/rc.local* reads the MAC address of the interface *eth1* and configures the IP of the interfaces:

```
IP Address eth0  = 192.168.100.nn
IP Address bat0  = 192.168.7.nn
MAC Address bat0 = 90:nn:nn:nn:nn:nn
```

Whit this convention of IP Address and MAC address it's easy to follow the packets trough the nodes. With the *eth0* interface it's possible to access the nodes via ssh to the IP address 192.168.100.nn. The host has the *vboxnet0* interface with the IP address 192.168.100.1. Each Vde-switch has a tap interface, and through "Wireshark" we can sniff the packets that traverse the Vde-switch.

3.4. VirtualBox

The VirtualBox version must by 4.3 or higher [4]. To verify VDE-Switch support, go to the Network configuration window, select Attached to: "Generic Driver", and then verify in the Name: box that VDE is listed. In VirtualBox it is difficult to learn the IP address for the interfaces of the virtual machines; this is solved by setting the MAC address in VirtualBox and in the "rc.local" script read the MAC address to set an according IP address for the interfaces.

3.5. Vde-switch

The main advantage of Vde-switch [8,14] over UML switch is that any clients can be attached to this virtual switch: VirtualBox, UML, tap interfaces, virtual interconnections, and not just UML instances. If the Vde-switches were just connected with Wirefilter [10] "patch cables" without modification, we would end up creating a single broadcast domain and unwanted switch loops: The goal is to allow the packets to travel only from one host to it's neighborhood, not farther. To accomplish this, the Vde-switch needs to be modified to have "coloured" ports. The idea is that each port has a "colour" (an integer number) and packets are only passed from ports to others with different colours. Packets are dropped on outgoing ports if it has the same colour (same number) as the incoming port. In this concept, the host port can have colour 1, the TAP port colour 2, while the interconnection ports have colour 0. In this way, packets can only travel from the host to (all of) the interconnection ports, or from one interconnection port to the host port. However packets can not travel between the interconnection ports, thus only allowing "one hop" connections and avoiding switch loops and shared broadcast domains. The concept is illustrated in figure 2.

The patch against vde2-2.3.2 (current latest stable version) to add this colour patch can be found here:

```
(http://www.open-mesh.org/attachments/download/152/vde2-2.3.2_
colour.patch).
```

The patched vde-switch can be download from here:

`(https://github.com/dbritos/Network-mesh-emulator/blob/master/vde2-2.3.2-patch.tar).`

3.6. Wirefilter

The Wirefilter program [7] is a tool where it's possible to simulate various link defects and limits as example: packet loss, burst loss, delay, duplicates, bandwidth, Interface speed, Channel capacity, Noise (damage to packets) and MTU

However as the links are only set up bidirectional, interferences can unfortunately not be simulated with this system. For advanced testing it might be necessary to apply the aforementioned link defects to some packets only, whereas other packets are able to traverse the emulated environment unharmed. Once you applied the "ethertype" patch you can specify an ethertype which Wirefilter will simply forward. To apply a packet loss of 50 % to all packets except BATMAN adv packets, run:

```
wirefilter --ether 0x4305 -l 50
```

This patch also allows to filter BATMAN Adv. packet types. To apply a packet loss of 50 % to all packets except BATMAN Adv. ICMP packets, run:

```
wirefilter --ether 0x4305:02 -l 50
```

You can specify up to 10 packet types (separated by colon). The patch against vde2-2.3.1 (current latest stable version) can be found here:

`http://www.open-mesh.org/attachments/download/106`

4. Using BAMNE

In this section, we present a simple example of using BAMNE to create a topology of five groups of nine nodes each. This example of BAMNE emulation is shown in figure 3, and discussed in detail. In this emulation, there are 45 nodes running BATMAN Adv. protocol in a computer with an Intel i7 CPU and $16GB$ of RAM.

The software have two modes of operation, when we run the program it begins in edition mode. In this mode we can build the network topology. To switch to emulation mode, in the main menu we can choose "RUN" mode and the emulation runs until we choose "STOP" from main menu to finish the emulation. In Execution mode we can't create nodes, but we can destroy nodes or links to see how the BATMAN protocol reacts to the case of nodes or links failures.

The information shown in the main window is the following: In the first line of the screen we can see the transmitted and received packets of each interfaces of the highlighted node: lo, *eth0*, *eth1*, eth2 and *bat0* interface. The second and third

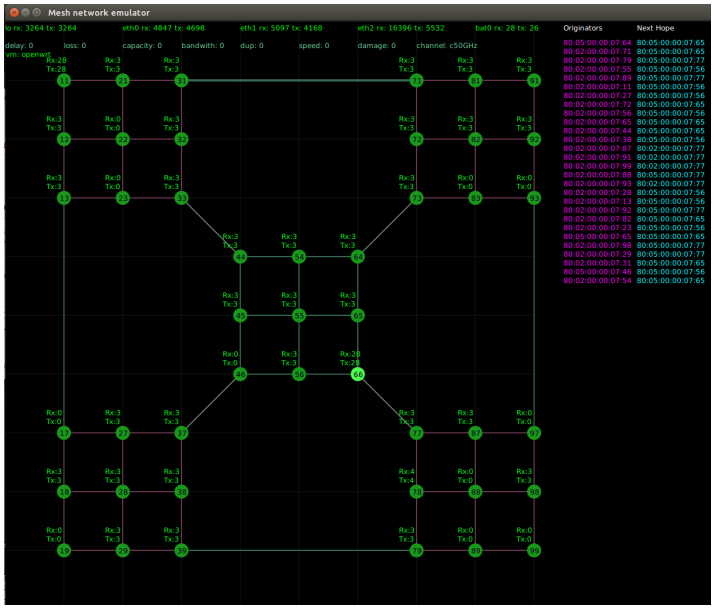


Figura 3. Emulator screen

line shows the properties of the links wire filter, packets loss, channel capacity, damage packets, delay, bandwidth of the channel, duplicate packets and channel frequency for each frequency 2,4 and $5GHz$ respectively . Above each node it's shown the number of transmitted and received packets. Inside the green circle of each node is shown the node number. The links between nodes in red are 2,4GHz links and the links in green are in $5GHz$.

In the top right of the screen there are the originators and next hop list for the node number of the highlighted node, 66 in this example.

When the virtual machine of the OpenWrt is created, a console of this machine appears in the main window. From there, a run test can be started, for example in the figure 4 the console of VM node 66 is making a ping to VM node number 71, with five hops the delay is acceptable, below $150ms$ and the jitter is below $30ms$ showing that the network can support VoIP as the values fall well within the boundaries recommended by the ITU-Recommendation G. 114 [9]. In the figure 5 is shown the CPU and memory usage for 45 nodes running all together, in this chart it is possible to see that the CPU usage is less than 35 % and the memory usage is less than a 35 % of $16GB$.

5. Conclusions

In this paper we propose an emulator for BATMAN Adv. protocol, with the capacity to evaluate the performance and the convergence in building the next

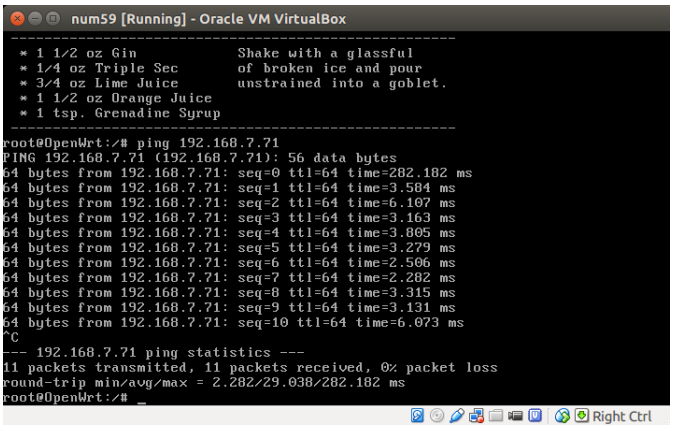


Figura 4. Delay with ping

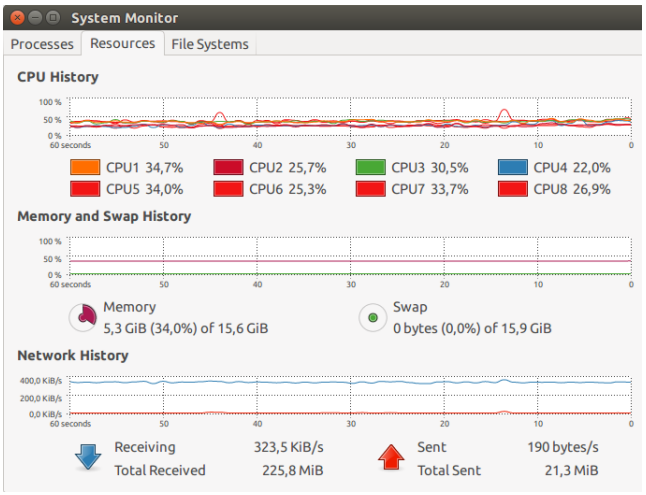


Figura 5. CPU usage

hop table of the protocol for many topology of the network. One of the improvements introduced by this emulator is the graphical interface allowing easy and fast deployment of network topology and interact with that, building and destroying links or changing their properties with rapid movements of the mouse and see the results in graphical form.The program fulfilled the expectations proposals, it can emulate various impediments on the transmissions links such as lost packets, delay, bandwidth, showing good performance with up to 90 nodes running on an Intel i7 processor with 16 GBs. of RAM. The principal goal to add to this program is to show graphically the trace route at level two of the OSI model.

Referencias

1. Batman-adv-openwrt-config - batman-adv - Open Mesh, http://www.open-mesh.org/projects/batman-adv/wiki/Batman-adv-openwrt-config
2. Doc-overview - batman-adv - Open Mesh, http://www.open-mesh.org/projects/batman-adv/wiki/Doc-overview
3. Emulation - Open-mesh - Open Mesh, http://www.open-mesh.org/projects/open-mesh/wiki/Emulation
4. OpenWrt in VirtualBox [OpenWrt Wiki], http://wiki.openwrt.org/doc/howto/virtualbox
5. Oracle VM VirtualBox, https://www.virtualbox.org/
6. SNMPD [OpenWrt Wiki], http://wiki.openwrt.org/doc/howto/snmp.server
7. Ubuntu Manpage: wirefilter - Wire packet filter for Virtual Distributed Ethernet, http://manpages.ubuntu.com/manpages/natty/man1/wirefilter.1.html
8. VDE - Virtualsquare, http://wiki.virtualsquare.org/wiki/index.php/VDE
9. ITU T.: One-way transmission time. recommendation G.114 (Feb 1996)
10. Caini, C., Firrincieli, R., Davoli, R., Lacamera, D.: Virtual Integrated TCP Testbed (VITT). In: Proceedings of the 4th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities. pp. 36:1–36:6. TridentCom '08, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium (2008), http://dl.acm.org/citation.cfm?id=1390576.1390620
11. Cerda-Alabern, L., Neumann, A., Escrich, P.: Experimental Evaluation of a Wireless Community Mesh Network. In: Proceedings of the 16th ACM International Conference on Modeling, Analysis & Simulation of Wireless and Mobile Systems. pp. 23–30. MSWiM '13, ACM, New York, NY, USA (2013), http://doi.acm.org/10.1145/2507924.2507960
12. Chissungo, E., Blake, E., Le, H.: Investigation into Batman-adv Protocol Performance in an Indoor Mesh Potato Testbed. In: 2011 Third International Conference on Intelligent Networking and Collaborative Systems (INCoS). pp. 8–13 (Nov 2011)
13. Combs, G., et al.: Wireshark-network protocol analyzer. Version 0.99 5 (2008)
14. Davoli, R.: VDE: Virtual distributed Ethernet. In: First International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005. pp. 213–220 (Feb 2005)
15. Johnson, D., Ntlatlapa, N., Aichele, C.: Simple pragmatic approach to mesh routing using BATMAN (Oct 2008), http://researchspace.csir.co.za/dspace/handle/10204/3035

Un Enfoque Multiobjetivo en Redes Periódicas de Tiempo Continuo

Diego Bosch[1], Benjamín Barán[1,2], and Uri Yael[1]

[1] Universidad Católica "Nuestra Señora de la Asunción". Asunción, Paraguay
{diegojbosch,ing.uriyael}@gmail.com
[2] Universidad Nacional del Este. Ciudad del Este, Paraguay
bbaran@cba.com.py

Resumen Una red dinámica es conocida como red periódica cuando se compone de enlaces que se conectan y desconectan periódicamente. Trabajos anteriores han tratado el problema teniendo en cuenta sólo tiempos discretos. En contrapartida, el presente trabajo utiliza las mismas funciones objetivo que estos trabajos, considerando por primera vez el tiempo continuo, además de una variabilidad en el tiempo que dura cada era (lapso de tiempo entre dos cambios de topología). El trabajo modela el problema en un contexto multi-objetivo y propone el uso de un Algoritmo Evolutivo Multi-Objetivo (el *Strength Pareto Evolutionary Algorithm*), implementado y luego comparado con respecto a los principales algoritmos del estado del arte como: DSDV, DSR, AODV, EG Shortest y EG Foremost, demostrando en las simulaciones que sería el preferido en un contexto puramente multi-objetivo que considera siete funciones objetivo: tiempo de viaje y tiempo de vida (promedio, varianza y peor caso), así como cantidad de saltos promedio.

Keywords: Grafo evolutivo, redes dinámicas, redes periódicas, optimización multi-objetivo, solución preferida.

1. Introducción

Una red dinámica es conocida como red periódica cuando se compone de enlaces que se conectan y desconectan con un intervalo regular de tiempo, conocido como periodo (T) [1]. Para transmitir un paquete de datos se necesita un camino desde el nodo fuente (remitente) hasta el nodo destino (destinatario). En el camino, existen nodos intermedios que actúan reenviando el paquete hasta llegar al destino. Si un nodo intermedio está ausente en un momento dado, el nodo predecesor debe almacenar el paquete hasta que el enlace entre ellos se active, o tomar la decisión de cambiar el camino, según sea la política de encaminamiento.

Las redes arriba citadas se analizan usando un formalismo para representar la conectividad de una red en cualquier instante, conocido como *Evolving Graph* (EG) [10], [11]. Un EG puede entenderse como una secuencia indexada de subgrafos que representan la configuración de la red en cada era e, como ilustra la Figura 1, donde por ejemplo, el enlace entre los nodos A y B solo está presente en la tercera y la décima era de cada ciclo (o periodo) que dura 15 eras.

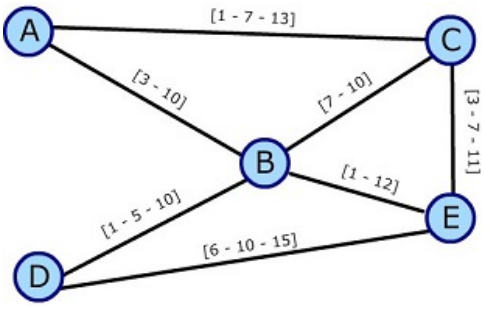


Figura 1. Grafo evolutivo que representa la topología de una red pequeña, donde en cada enlace se identifica las eras en la que se encuentra activo

Las redes periódicas han llegado a ser muy útiles en el modelado de redes de comunicación. Por citar algunos ejemplos, se las utiliza en el estudio de Satélites de Órbita Baja (*Low Earth Orbit* - LEO), redes de sensores, redes móviles del tipo *ad-hoc* (MANET) [8], así como en el proyecto *One Laptop Per Child* (OLPC) que utiliza este modelado con un fin educativo [4], entre otras aplicaciones posibles.

2. Contexto del trabajo

Este trabajo sigue la línea de investigación propuesta por Yael y Barán [16], [17], donde se aborda el problema de encaminamiento en redes dinámicas periódicas, modelándolo como un problema multi-objetivo. En dichos trabajos, Yael y Barán proponen un algoritmo evolutivo multi-objetivo, capaz de hallar un conjunto de soluciones Pareto óptimas [12], que no sólo encuentra individuos competitivos con respecto a las soluciones calculadas por otros algoritmos del estado del arte, sino que además logra calcular individuos que dominan (en el sentido Pareto [5]) a las soluciones calificadas como óptimas, cuando una sola función objetivo es considerada. Aquí corresponde recordar que una solución multi-objetivo es mejor que otra (o la domina) si no es peor en ningún objetivo y es estrictamente mejor en al menos un objetivo [5]. Trabajos anteriores [10], [11], [16] solo consideran la duración de las eras como una variable discreta t y no asignan un valor específico en segundos a dicha variable. Esto resulta en que las soluciones son comparadas, entre otras formas, por la cantidad de eras transcurridas y, por lo tanto, dichos trabajos asumen implícitamente que la duración en segundos de dos eras cualesquiera, es siempre la misma.

Los trabajos [10], [11], [16] utilizan Grafos Evolutivos que permiten implementar el concepto de camino (o viaje) compuesto por pares ordenados $< n, e >$, donde n es un nodo y e una era en la que el nodo se encuentra conectado a la red. El presente trabajo propone por primera vez asignar a cada era una duración en segundos, considerando para ello un proceso aleatorio pues en la mayoría de los casos prácticos este valor no es necesariamente constante [3]. De esta forma,

la misma era en ciclos distintos puede tener una duración ligeramente diferente al ser considerada como una variable aleatoria. Esto permite simular diferencias en la duración de las eras, debido por ejemplo a márgenes de error en los cronogramas o fenómenos atmosféricos que impactan sobre la conectividad real de la red al momento efectivo de su utilización.

Al realizar simulaciones considerando eras cuyas duración es aleatoria, se espera que las soluciones del Algoritmo Evolutivo Multi-objetivo sean mejores en comparación a las soluciones tradicionales utilizando algoritmos mono-objetivo presentados en los trabajos de referencia [16], [17].

3. Simulaciones

3.1. Simulador

Siguiendo la línea propuesta en [16], [17], este trabajo modifica el simulador desarrollado por los autores en dichos trabajos de forma a considerar tiempos físicos (medidos en segundos) para cada era. Recordando que el tiempo de duración de una era es una variable aleatoria, esta aleatoriedad es representada mediante el uso de diferentes escenarios que reflejen el azar. Esto es, un escenario es una realización concreta de valores que puede ir tomando cada era durante el tiempo que dure una simulación. Estos escenarios pueden ser generados en forma aleatoria a partir de una función de distribución de probabilidad, como se hace en este trabajo, u obtenerse a partir de datos históricos o incluso ser propuestos por un experto.

Cabe destacar que en las simulaciones que se presentarán se considera que los paquetes siempre llegan a destino dado que si el enlace deja de estar disponible en una era, los paquetes todavía no enviados pueden ser calendarizados para ser transmitidos en el siguiente ciclo, dado que la red tiene una periodicidad T asegurada y no se consideran restricciones en los nodos.

3.2. Funciones objetivo

Trabajos anteriores [16], [17] han calculado un conjunto de soluciones óptimas teniendo en cuenta la optimización de varias funciones objetivos como: la cantidad promedio de saltos efectuados por los paquetes para llegar al nodo destinatario, la cantidad promedio de eras vividas por los paquetes hasta que todos ellos culminan su camino y la cantidad promedio de eras durante las cuales los paquetes viajan a través de la red hasta llegar a destino [16]. Este trabajo utilizará las mismas funciones objetivo de modo a comparar las simulaciones y los resultados obtenidos. La *cantidad de saltos* mide cuántos enlaces fueron atravesados por un paquete para llegar desde el nodo remitente hasta el nodo destino; el conjunto de caminos más cortos es el calculado con el algoritmo de Dijkstra [10]. El *tiempo de vida* es el tiempo transcurrido desde que se crea un paquete para ser enviado, hasta que llega a su destino. El *tiempo de viaje* es el tiempo transcurrido desde que un paquete efectivamente sale del remitente y hasta que

llega al destinatario. Es decir, no se considera el tiempo transcurrido entre que se crea un paquete y se inicia su transmisión efectiva. Notar que el tiempo de vida es igual al tiempo de viaje más el tiempo de espera.

Debido a que cada algoritmo se simulará sobre N escenarios que caracterizarán la distribución de probabilidad de las variables aleatorias del problema considerado, surgen nuevas alternativas al definir las funciones objetivo a partir de las arriba descritas. Para este trabajo se ha decidido trabajar con el promedio, varianza y peor caso para las funciones objetivo que consideran tiempos, dando como resultado siete funciones objetivo: f_1: promedio de la cantidad de saltos, f_2: peor caso del tiempo de viaje, f_3: promedio del tiempo de viaje, f_4: varianza del tiempo de viaje, f_5: peor caso del tiempo de vida, f_6: promedio del tiempo de vida y f_7: varianza del tiempo de vida.

3.3. Entorno de simulación

Los algoritmos evaluados en las simulaciones que siguen son los utilizados en [16], [17]: *Destination-Sequenced Distance-Vector* (DSDV) [13], *Dynamic Source Routing* (DSR) [9], *Ad-hoc On-Demand Distance Vector* (AODV) [14], *Evolving Graph Shortest* (EG Shortest), [10] y *Evolving Graph Foremost* (EG Foremost) [10], además del *Strength Pareto Evolutionary Algorithm* (SPEA) [18], utilizando los siguiente parámetros en las simulaciones: cantidad de individuos por generación = 10, cantidad máxima de individuos en la población externa = 1000, probabilidad de cruzamiento = 0.5 y probabilidad de mutación = 0.25.

El simulador originalmente programado en Ruby [16] fue adaptado para simular el funcionamiento de los algoritmos en la red periódica de la Figura 1. Las simulaciones fueron ejecutadas en una máquina con un procesador Intel Core i5, 2.27 GHz y memoria RAM de 6.00 GB. La duración de las simulaciones, dependiendo del conjunto de escenarios utilizados fueron variando desde algunas horas, varios días y hasta un poco más de una semana. Cabe destacar que en general los algoritmos mono-objetivo necesitaron un tiempo de simulación considerablemente menor al SPEA.

Las pruebas que se presentan a continuación se realizaron utilizando la red periódica de la Figura 1. Al comenzar una simulación, lo primero que se hace es generar la cantidad de escenarios que se necesite. En la Tabla 1 se puede ver un ejemplo de una matriz generada para un escenario correspondiente a la red antes mencionada (Figura 1), donde cada fila representa un ciclo (letra C), cada columna representa una era (letra e) y cada casilla representa la duración en segundos, recordando que en una red periódica cada ciclo tiene un número constante de eras; por ejemplo, en la Tabla 1, los 5 ciclos analizados están compuestos de 15 eras, cada uno.

Todas las pruebas fueron realizadas sobre cuatro escenarios distintos que fueron generados al comenzar la simulación. Al considerar cuatro escenarios ya se tiene una razonable variedad de datos distintos, haciendo que la simulación pueda ser viable en cuanto a recursos y tiempo consumido. Una mayor cantidad de escenarios es lógicamente preferible pero requeriría un tiempo de ejecución mayor, como así también una cantidad de recursos muy grande para la realidad

Tabla 1. Ejemplo de escenario, donde E representa una era y C representa un ciclo

	e1	e2	e3	e4	e5	e6	e7	e8	e9	e10	e11	e12	e13	e14	e15
C1	0,15	0,14	0,22	0,18	0,27	0,15	0,14	0,22	0,18	0,27	0,15	0,14	0,22	0,18	0,75
C2	0,16	0,18	0,20	0,12	0.17	0,16	0,19	0,20	0,12	0.17	0,16	0,18	0,20	0,16	0.71
C3	0,44	0,29	0,26	0,27	0.15	0,24	0,29	0,26	0,27	0.18	0,24	0,29	0,26	0,27	0.15
C4	0,10	0,16	0,16	0,10	0.23	0,10	0,16	0,16	0,10	0.23	0,10	0,16	0,19	0,10	0.23
C5	0,29	0,20	0,23	0,26	0.29	0,29	0,20	0,23	0,24	0.29	0,29	0,20	0,23	0,26	0.29

concreta en que se realizó este trabajo. Para determinar la duración de cada era se utilizó una distribución de probabilidad triangular, escogida empíricamente de forma a lograr que los paquetes lleguen razonablemente a destino.

En lo que sigue, se asume que el tiempo que tarda en viajar un paquete de un nodo a otro es de 100 milisegundos (es decir 0,1 segundos). Las pruebas se realizaron con un tráfico que contenía 31 paquetes que se van creando dentro de la red ilustrada en la Figura 1. Para una red de este tamaño se puede asumir que se trata de un tráfico bastante congestionado. Entonces, de manera a que varios paquetes no puedan ser enviados dentro de una misma era por falta de tiempo, se definió que las eras duraban entre 0,1 y 0,3 segundos. De este modo se puede ver como se comportan los algoritmos simulados a medida que las eras y por ende los ciclos, van avanzando en el problema ejemplo presentado. En el caso de que un paquete no pueda ser enviado en una era debido a que el tiempo de la era finaliza, se lo calendariza para ser enviado en la misma era pero en el siguiente ciclo. De esta manera, todos los paquetes en algún momento llegarán necesariamente a su destino final para la red dinámica de la Figura 1.

3.4. Esquema de simulación

En la Figura 2 se puede apreciar el esquema a seguir en las simulaciones, donde las siglas utilizadas son: (N) número máximo de escenarios, (E_n) un escenario de los N posibles, (AB) enlace del nodo A al nodo B, (AC) enlace del nodo A al nodo C, (S_m) una solución SPEA de las m posibles, (FO) función objetivo, (t) tiempo medido en segundos, además de utilizar las siglas correspondientes a cada algoritmo arriba mencionado.

En primer lugar se utiliza la distribución de probabilidad para los escenarios, con la cual se van generando los N escenarios que se necesitan para empezar la simulación (pasos 1 y 2 de la Figura 2). Luego, el esquema de simulación se divide de acuerdo al tipo de algoritmo a simular (pasos 3 y 4 de la Figura 2). Para los algoritmos mono objetivo primeramente se tiene como entrada un escenario para luego probar la solución encontrada sobre cada uno de los N escenarios considerados. Como resultado de las simulaciones sobre los N escenarios se pueden calcular las 7 funciones objetivo para cada algoritmo, las que luego son comparadas para conformar un conjunto de soluciones no dominadas (Pareto óptimas). El algoritmo multi-objetivo SPEA realiza cada iteración sobre los N escenarios,

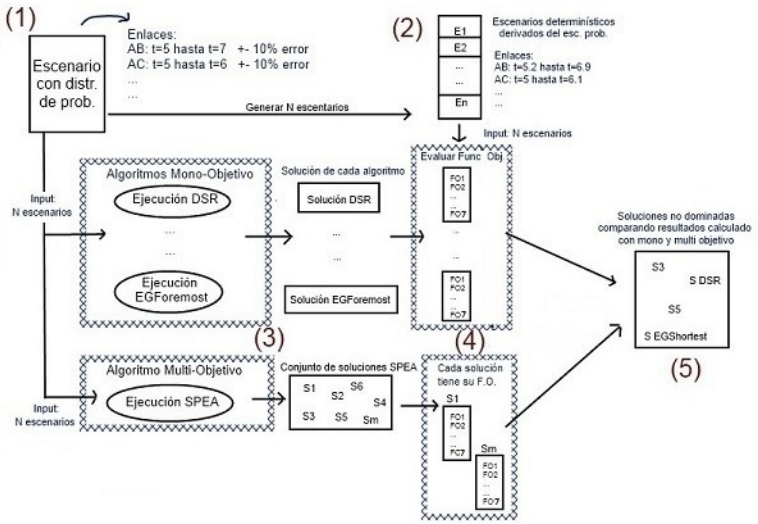


Figura 2. Esquema de simulación

para dar como resultado un conjunto de soluciones no dominadas en una sola corrida del algoritmo. Cada solución lógicamente tiene sus correspondientes funciones objetivo debidamente calculadas.

Por último, las soluciones de los algoritmos mono-objetivo y las soluciones del SPEA son comparadas entre sí para evaluar si forman parte del conjunto final de soluciones no dominadas (paso 5 de la Figura 2). De esta forma, se obtiene un conjunto de soluciones no dominadas que permite realizar un análisis final, el cual se explica en la siguiente sección.

4. Resultados experimentales

Las pruebas fueron realizadas de manera a ir comparando el algoritmo SPEA con los demás algoritmos mono-objetivo. Se hicieron pruebas del SPEA con 5, 20, 40, 100, 300 y 1000 iteraciones ya que el SPEA, al ser un algoritmo evolutivo, itera hasta que converja a un razonable conjunto de soluciones no dominadas.

Una vez terminada una simulación, se comparan los resultados de los algoritmos mono-objetivo para tener un conjunto de soluciones no dominadas. Debido a la gran cantidad de funciones objetivo que se calculan para cada solución, resulta muy difícil que una solución de un algoritmo en particular domine a todas las otras soluciones [15]. Entonces, para estos casos donde se tienen varias funciones objetivo, se puede utilizar la Relación de Preferencia introducida por Drechsler et al. en [7] quienes utilizaron un algoritmo previamente propuesto en [6] para resolver un problema de múltiples objetivos. Dicho algoritmo utiliza una relación de preferencia para comparar soluciones no dominadas entre sí [15]. La relación de preferencia define que, dadas dos soluciones x y x', se dice que x es

Tabla 2. Conjunto de soluciones no dominadas encontradas al resolver el problema ejemplo de la Figura 1

Algoritmo	f_1	f_2	f_3	f_4	f_5	f_6	f_7
1.DSDV	1,677	8,935	10,387	0,819	25,613	28,758	4,389
3.DSDV	1,677	8,935	10,266	1,098	25,613	28,637	4,024
17.EGs	1,677	5,516	9,266	5,780	21,968	22,935	0,936
20.EGs	1,677	6	8,661	2,985	20,516	22,330	2,619
22.SPEA	1,935	11,258	11,379	0,043	24,516	26,330	1,565
32.SPEA	2,097	10,677	11,645	0,351	21,71	25,097	6,204
34.SPEA	2,129	12,387	13,113	0,292	21,806	27,008	11,869
37.SPEA	1,839	7,226	10,250	3,321	19,29	22,677	6,205

preferida a x' si y sólo si x tiene una mayor cantidad de objetivos mejores que x'.

Dentro de todas las simulaciones realizadas [2], a continuación sólo se presenta una simulación específica tomada como ejemplo para analizar los resultados obtenidos. En esta simulación en particular, el algoritmo SPEA sólo fue iterado 20 veces. Una vez que el simulador ejecutó todos los algoritmos antes mencionados sobre los cuatro escenarios, resulta un conjunto de soluciones no dominadas que se pueden apreciar en la Tabla 2. En dicha tabla cada columna f representa una función objetivo conforme fuera definido al final de la sección 3.2.

Tabla 3. Comparación del número de funciones objetivo en que una solución es mejor que la otra, considerando sólo las soluciones no dominadas.

	1.DSDV	3.DSDV	17.EGs	20.EGs	22.SPEA	32.SPEA	34.SPEA	37.SPEA	**S**
1.DSDV	**	1(N)	2(N)	2(N)	2(N)	3(N)	2(N)	3(N)	0
3.DSDV	3(P)	**	2(N)	2(N)	2(N)	3(N)	2(N)	3(N)	1
17.EGs	4(P)	4(P)	**	2(N)	4(P)	3(N)	3(N)	3(N)	3
20.EGs	4(P)	4(P)	4(P)	**	3(N)	4(P)	3(N)	5(P)	5
22.SPEA	5(P)	5(P)	3(N)	4(P)	**	4(P)	6(P)	3(N)	5
32.SPEA	4(P)	4(P)	4(P)	3(N)	3(N)	**	5(P)	3(N)	4
34.SPEA	5(P)	5(P)	4(P)	4(P)	1(N)	2(N)	**	3(N)	4
37.SPEA	4(P)	4(P)	4(P)	2(N)	4(P)	4(P)	4(P)	**	6

Luego, con las soluciones no dominadas de la Tabla 2 se crea una matriz de comparación de las funciones objetivo de cada solución (ver Tabla 3), donde se compara en cuantas funciones objetivo la solución de una fila es mejor que la solución de una columna. Así, al aplicar la Relación de Preferencia se asigna el valor (P) si la solución de la fila es preferida a la solución de la columna y el valor (N) si la solución de la columna es preferible a la solución de la fila. Para identificar cual sería la solución preferida entre todas las soluciones no dominadas, se puede por ejemplo contar las veces que una solución ha sido la

preferida sobre otras; es decir, los valores P para cada fila, lo que se muestra en la columna '**S**' de la Tabla 3. Finalmente, se puede notar que la solución preferida bajo este criterio sería la solución '37.SPEA' por tener el valor más alto (6) en la columna '**S**'.

Tabla 4. Relación de preferencia entre las soluciones no dominadas elegidas para realizar el grafo de la Figura 3.

	2.DSDV	3.DSDV	17.EGs	18.EGs	19.EGs	20.EGs	138.SPEA	234.SPEA
2.DSDV	**	(P)	(N)	(N)	(N)	(N)	(N)	(P)
3.DSDV	(N)	**	(N)	(N)	(N)	(N)	(N)	(N)
17.EGs	(P)	(P)	**	(P)	0	(P)	(N)	(N)
18.EGs	(P)	(P)	(N)	**	(N)	0	(N)	(N)
19.EGs	(P)	(P)	0	(P)	**	(P)	(N)	(N)
20.EGs	(P)	(P)	(N)	0	(N)	**	(N)	(P)
138.SPEA	(P)	(P)	(P)	(P)	(P)	(P)	**	0
234.SPEA	(N)	(P)	(P)	(P)	(P)	(N)	0	**

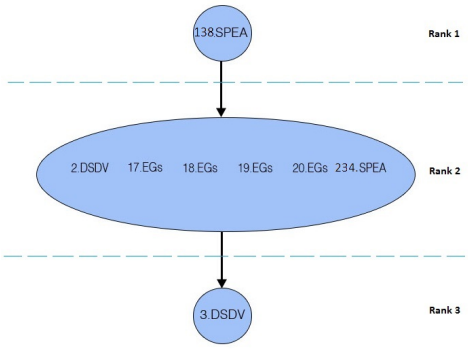


Figura 3. Grafo de relación de preferencia correspondiente a la Tabla 4, donde la solución no dominada '138.SPEA' se encuentra en el primer nivel de la clasificación, quedando así demostrado que es la solución preferida.

Alternativamente, la selección de la solución preferida puede hacerse utilizando una técnica más formal como un *Grafo de Relación de Preferencia* [15]. Para ilustrar este concepto, la Figura 3 presenta el *Grafo de Relación de Preferencia* correspondiente a la Tabla 4, similar a la Tabla 3 que resulta de otra simulación que obtuvo 110 soluciones no dominadas que por un tema de espacio y didáctica solo se presenta parcialmente. En efecto, la Figura 3 muestra que la solución '138.SPEA' es preferida a todas las demás soluciones por lo que se encuentra en el primer lugar de la clasificación (Rank 1); luego se observa que seis soluciones

colapsan en un solo nodo (Rank 2) debido a que todas ellas resultan preferidas a al menos una solución de este conjunto, existiendo al menos una solución del conjunto que es preferida a la solución considerada; es decir, al graficar las relaciones de preferencia entre estas seis soluciones se encontrará al menos un ciclo. Finalmente, se puede observar que la solución '3.DSDV' en ningún caso es preferida a pesar de ser no dominada, por lo que queda última en la clasificación (Rank 3). Cabe destacar que la elección de una solución calculada por el algoritmo SPEA como preferida en el ejemplo no es una simple casualidad dado que consistentemente la soluciones preferidas siempre fueron calculadas por el SPEA corriendo un número suficientemente grande de iteraciones.

5. Conclusión

La primera contribución de este trabajo es el simulador implementado que permite incluir al tiempo continuo en las simulaciones y que incluso cuenta con la posibilidad de utilizar diversos escenarios para modelar la incertidumbre.

Experimentalmente se comprobó que algunos algoritmos como el AODV, DSR y EG Foremost no son buenos para resolver el problema en cuestión, dado que no han sido capaces de encontrar soluciones no dominadas. En contrapartida, se demostró experimentalmente que algunos algoritmos como el DSDV y el EG Shortest son razonablemente buenos (aún siendo algoritmos mono-objetivo) dado que lograron encontrar soluciones no dominadas, aunque estas soluciones no lleguen a ser las soluciones preferidas. Por lo tanto, si no se tiene acceso a suficientes recursos computacionales como para utilizar un algoritmo multi-objetivo, estos dos algoritmos serían los recomendables.

Cabe destacar que por primera vez se propone la utilización de la Relación de Preferencia [15] para comparar el gran número de soluciones no dominadas encontradas en el problema estudiado. Así, se demostró que mediante esta Relación de Preferencia se logró identificar la solución no dominada preferida del conjunto de soluciones Pareto óptimas, incluso al considerar un número considerable de funciones objetivo, siete en este caso.

Como principal aporte de este trabajo, se aportan datos experimentales que concluyentemente indican que un algoritmo de ruteo que considere un contexto multi-objetivo (en este caso el SPEA) es claramente superior a los algoritmos del estado del arte, típicamente mono-objetivo, para redes periódicas de tiempo continuo en contextos de incertidumbre. Como trabajo futuro los autores se encuentran experimentando con otros algoritmos multi-objetivos, diversas redes periódicas y variados conjuntos de escenarios.

Referencias

1. Ash, G. R. (1995). Dynamic network evolution, with examples from AT&T's evolving dynamic network. *Communications Magazine, IEEE*, 33(7), 26-39.
2. Bosch, D. (2015) Ingeniería de tráfico en redes dinámicas periódicas en tiempo continuo utilizando el algoritmo SPEA. Tesis de grado. Universidad Católica "Nuestra Señora de la Asunción".

3. Bosch, D., Barán, B., & Yael, U. (2012) Ingeniería de tráfico en redes dinámicas periódicas en tiempo continuo con eras de duración estocástica. *IEEE ARANDU-CON*.
4. Carrano, R. C., Bletsas, M., & Magalhães, L. C. (2007). Mesh networks for digital inclusion-testing OLPC's XO mesh implementation. In *8o Forum Internacional de Software Livre, 2007, Porto Alegre.*
5. Coello, C. A., Lamort, G.B., & Van Veldhuizen, D.A. (2007). Evolutionary Algorithms for Solving Multi-Objective Problems. 2da.Edición, Editorial Springer.
6. Drechsler, N., Drechsler, R., & Becker, B. (1999). Multi-objected optimisation in evolutionary algorithms using satisfyability classes. In: *Reusch, B. (ed.) International Conference on Computational Intelligence, Theory and Applications, 6th Fuzzy Days. Lecture notes in Computer Science*, vol. 1625, pp 108-117. Springer, Dortmund.
7. Drechsler, N., Drechsler, R., & Becker, B. (2001). Multi-objected optimisation based on relation favour. In: *First International Conference on Evolutionary Multi-Criterion Optimization. Lecture Notes in Computer Science*, vol. 1993, pp. 154-166. Springer, Berlin.
8. Jarry, A., & Lotker, Z. (2004). Connectivity in evolving graph with geometric properties. In *Proceedings of the 2004 joint workshop on Foundations of mobile computing* (pp. 24-30). ACM.
9. Johnson, D. B., & Maltz, D. A. (1996). Dynamic source routing in ad hoc wireless networks. In *Mobile computing* (pp. 153-181). Springer US.
10. Monteiro, J. (2007). Uso de Grafos Evolutivos no Roteamento em Redes Dinâmicas: algoritmos, fluxos e limites. Tesis de maestría. Instituto de Matemática y Estadística, Universidad de Säo Paulo. 2007.
11. Monteiro, J., Goldman, A., & Ferreira, A. (2007, June). Using Evolving Graphs Foremost Journey to Evaluate Ad-Hoc Routing Protocols. In *Procedings of 25th Brazilian Symposium on Computer Networks (SBRC'07)*, Blem, Brazil.
12. Pareto, V. (1896). Cours D'Economie Politique,Volume 1. Lausanne: F. Rouge.
13. Perkins, C. E., & Bhagwat, P. (1994). Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *ACM SIGCOMM Computer Communication Review* (Vol. 24, No. 4, pp. 234-244). ACM.
14. Perkins, C., Belding-Royer, E., & Das, S. (2003). Ad hoc On-Demand Distance Vector (AODV) Routing. IETF. RFC 3561.
15. von Lücken, C., Barán, B., & Brizuela, C. (2014). A survey on multi-objective evolutionary algorithms for many-objective problems. *Computational Optimization and Applications*, 1-50.
16. Yael, U. (2009). Ingeniería de Tráfico en Redes Dinámicas Periódicas Utilizando el Algoritmo SPEA. Tesis de grado, Universidad Católica "Ntra. Sra. de la Asunción".
17. Yael, U., & Barán, B. (2010) Routing in Periodic Dynamic Networks using a Multi-Objective Evolutionary Algorithm. *ALIO-INFORMS Joint International Meeting.*
18. Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. E*volutionary Computation, IEEE on*, 3(4), 257-271.

Localidad estructural, criterio de división para la ejecución de redes de Petri no autónomas en IP-Core

Dr. Ing. Orlando Micolini[1], Geol. Marcelo Cebollada y Verdaguer[1], Ing. Ventre, Luis Orlando[1].

[1]Laboratorio de Arquitectura de Computadoras - FCEFyN - Universidad Nacional de Córdoba
omicolini@compuar.com, {mcebollada, lventre}@gmail.com

Abstract. — Este trabajo propone un nuevo concepto, localidad estructural, para dividir y representar redes de Petri no autónomas con el fin de reducir de forma significativa los recursos de hardware de los IP-Core que las ejecutan. Con la consecuente ventaja de lograr asi abordar problemas de mayor tamaño.

Las redes así expresadas dan origen a un algoritmo de ejecución que preserva el modelo original y facilita el paralelismo.

En este trabajo se expone un caso de aplicación donde se manifiestan las ventajas de la localidad estructural aplicada a una red de Petri con diferentes semánticas temporales y tipos de brazos, obteniendo una disminución de recursos en la FPGA que implementa el IP-Core.

Abstract. — This paper proposes a new concept, structural locality, to divide and represent no autonomous Petri nets, with the aim of significantly reduce the hardware resources of the IP-cores that run them. With the consequent advantage of addressing in this way larger problems.

The Petri nets represented by this way raise an algorithm of execution which preserves the original model and facilitates the parallelism.

Finally in this paper we expose a real case of application where shows the advantages of the structural locality applied to a Petri net with different temporal semantics and types of arms, achieving an important decrease in resources of the FPGA that implements the IP-Core.

Keywords: Procesador de Petri, Redes de Petri jerárquicas, localidad estructural, IP-Core.

1 Introducción

Para mejorar el rendimiento de los sistemas informáticos y la capacidad de cómputo, las aplicaciones actuales implementan hilos [1] que se ejecutan en sistemas multicore. Estos hilos cooperan y se ejecutan concurrentemente. Las aplicaciones diseñadas de esta manera generan una complejidad adicional con respecto a los programas secuenciales. Esta complejidad se manifiesta en el diseño, detección de errores, testing, validación y mantenimiento [2]. Además es necesario introducir mecanismos de control, como los semáforos, que generan penalización en los tiempos de ejecución.

Por todo esto es importante resolver el sistema formalmente, lo que puede realizarse haciendo un modelo que facilite la programación.

Investigaciones recientes mostraron que los modelos obtenidos con redes de Petri (RdP) pueden facilitar la implementación de sistemas de manera directa, utilizando procesadores o IP-Core que ejecutan RdP [3-6].

Los principales problemas de estas soluciones son: el tamaño de las matrices que representan los modelos, las distintas semánticas temporales, tipos de brazos y tipos de eventos en las transiciones. Los que limitan el tamaño de los problemas que se pueden abordar, debido a los recursos de hardware disponibles.

Los procesadores de RdP (PP) han sido implementados como IP-Cores, en una FPGAs Spartan-6, como se muestra en [7, 8]. Debido a las limitaciones de hardware existentes, se pueden sintetizar con facilidad PP capaces de contener hasta 50 plazas y 50 transiciones. Para resolver sistemas de mayor envergadura se requiere modelos de RdP con mayores números de componentes o elementos, como: plazas, transiciones, tipos de brazos y distintas semánticas temporales. Hay que notar que los recursos demandados a la FPGA, para implementar los IP-Core, aumentan en proporción al producto de la cantidad de plazas por la cantidad de transiciones y por la cantidad de distintos tipos de brazos.

Por cada tipo de brazo existe una matriz de dimensiones Plazas por Transición y además por cada transición existe una cola de eventos de entrada y una de salida. También según sea el tipo de semántica temporal pueden existir registros de tiempo y un contador, ambos de 32 bit.

En el presente trabajo se amplía el uso de los mecanismos de división de las RdP, introducido en [8], haciendo uso del concepto de localidad estructural, para obtener RdP jerárquicas (HPN) que aprovechan los recurso de manera más efectiva, y mantienen todas las propiedades de la ecuación de estado de las RdP no autónomas.

2 Redes de Petri jerárquicas

En las HPN [9], cada subred que compone el sistema posee un estado particular en un momento dado, esto da lugar a que haya distintas transiciones sensibilizadas por cada subred del sistema en un instante determinado. Estas transiciones pueden disparse arbitrariamente si no hay conflicto entre ellas. En caso de conflicto se emplea un esquema de prioridades para mantener el determinismo del sistema.

Existen dos tipos de transiciones en las subredes, las transiciones internas y las transiciones de borde [9]. Para que una transición de borde pueda ser disparada es necesario que todas las transiciones de borde que representan a esa única transición en el sistema original estén sensibilizadas, esto conserva la semántica original de las transiciones en las RdP.

Las transiciones de borde de cada subred, son transiciones que han sido divididas y pertenecen a distintas subredes. Como consecuencia de la división una transición queda distribuida y pertenece a subredes distintas.

$$[T_i] \xrightarrow{Ri^{|Ti|x|Tb|}} [Transiciones\ de\ Borde\ de\ la\ subred\ [b]]$$

Donde T_i son las transiciones de la subred i.

Todas las transiciones de borde que no tienen transición interna en la subred i deben estar permanentemente sensibilizada en la transición de borde de la subred i, para esto se realiza una máscara de la subred i. La máscara tiene un uno por cada transición de borde que no está en la subred.

$$sensibilizados_de_borde_de_la_subred[i] = Transiciones\ de\ Borde\ de\ la\ subred\ [i]\ \ OR\ mascara_subred[i]$$

$$sensibilizados_distribuidos = \bigwedge_{j=0}^{r-1} sensibilizados_de_borde_de_la_subred[i]$$

Donde r es la cantidad de subredes en que se ha dividido el sistema.

La Figura 1 muestra las transiciones divididas, distribuidas y de borde de una RdP que modela un productor consumidor.

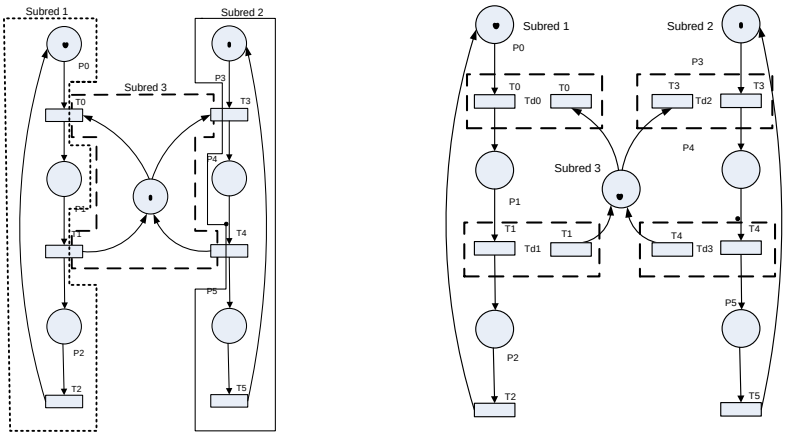


Figura 1. Red de productor consumidor dividida en tres subredes

Las transiciones distribuidas, en la figura son: Td0, Td1, Td2 y Td3. Mientras que las transiciones internas de la subred 1 son T0, T1 y T2, para la subred 2 son T3, T4 y T5 y para la subred 3 son T0, T1, T3 y T4.

El algoritmo de disparo de HPN [9] mantiene la ejecución de la RdP original, y es implementado en hardware mediante un circuito combinacional, que ejecutan la lógica de las ecuaciones del apartado anterior. Esto ha sido realizado con compuertas simples que permiten mantener los beneficios temporales alcanzados por el PP [8].

Interpretación del rango de la matriz de incidencia

De la semántica de disparo de una RdP podemos interpretar a la matriz de incidencia considerando a las columnas (transiciones) como la evaluación conjuntiva de las

restricciones que imponen las filas (plazas). Es decir que en una matriz de incidencia de dimensión mxn se evalúan n combinaciones de m variables lógicas, como se expresa en la siguiente ecuación:

$$if_{h=0}^{n-1}(\bigwedge_{k=0}^{m-1} p_i \geq w_{i,h})$$

Donde $w_{i,h}$ son los elementos de la matriz I [3] que resulta de la resta entre las matrices I_i^+ e I_i^-, y representan el peso y la dirección de los arcos de la RdP. Donde p_i es la marca en la plaza i.

3 División de RdP

La investigación bibliográfica del análisis de división de RdP ha sido realizada en trabajo [6].

Si consideramos que las RdP están constituidas por plazas, transiciones y brazos, a los cuales llamamos elementos. Donde, estos elementos participan de la matriz de incidencia, de la siguiente manera: las transiciones como la cantidad de columnas, las plazas como la cantidad de filas y los brazos son los valores que relaciona una plaza con una transición, según un peso y una dirección.

En las RdP vemos que estos elementos se agrupan según subconjuntos, caracterizados por estar fuertemente relacionados entre sí. Es decir, que hay brazos relacionando plazas con transiciones y viceversa. A su vez, estos subconjuntos se relacionan con otros subconjuntos de una manera débil, es decir que hay pocos brazos que relaciona a ambos sub conjuntos. Por esto vemos a la matriz de incidencia, como un caso de "matriz rala" [10] [11] [12].

Como podemos ver en la bibliografía, los algoritmos para resolver estas matrices ralas no son aplicables a implementaciones del PP con FPGA. Los algoritmos evaluados usan técnicas de compactación y/o punteros para direccionar los elementos, no usan operaciones lógicas simples; todo esto demanda recursos y ciclos de máquina, lo que se traduce en excesiva sobrecarga.

Con el fin de obtener una reducción de recursos se propone dividir la red en subredes. Para explicar esto, ahora suponemos que una red con M plazas y N transiciones es posible dividirla en dos subredes, donde cada una tienen la mitad de plazas (M/2) y transiciones (N/2), lo que resulta en:

- Para el sistema sin dividir $\boldsymbol{M} * \boldsymbol{N}$ elementos en la matriz.
- Para el sistema dividido $\boldsymbol{2}((\boldsymbol{M}/\boldsymbol{2}) * (\boldsymbol{N}/\boldsymbol{2})) = (\boldsymbol{1}/\boldsymbol{2})\boldsymbol{M} * \boldsymbol{N}$, más una sobrecarga que especifica las relaciones entre las subredes.

Hacemos notar que la cantidad total de plazas y transiciones del sistema se mantiene, por lo que en principio es posible mantener su representación, siempre y cuando seamos capaces de determinar cómo relacionar las subredes. Además, la sobrecarga que especifica las relaciones entre las subredes tiene que ser menor a $(1/2)M * N$. Esta diferencia se traducirá en la reducción de recursos. Para este caso, en que la red

ha sido dividida en dos subredes, la ganancia máxima es de dos (sin considerar la sobrecarga).

Podemos considerar que si se divide la red en J partes, la ganancia máxima teórica sería función de J, menos el costo de interconexión. La determinación de la ganancia máxima teórica sin tener en cuenta la localidad estructural ha sido tratada en [9].

Teniendo en cuenta lo dicho, proponemos reducir aún más la cantidad de recursos en el HPP, donde se realizara la división logrando que cada PP que conforma al HPP tenga sólo los recursos necesarios para ejecutar la subred.

3.1 Consideraciones para División RdP

Para realizar la disminución de recursos que nos hemos propuesto, dividiendo una RdP de la forma y el modo anteriormente propuesto, es necesario contestar los siguientes interrogantes:

- ¿Cuánto y cuál es la ganancia de dividir las RdP? Esto ha sido contestado, sin considerar la localidad estructural, en el trabajo [9].
- ¿Las RdP resultante son más simples? Las subredes son más simples, dado que es posible realizar la división de las transiciones en cualquier número de partes, lo que nos permite elegir subredes simples.
- ¿Existe un criterio de división de la RdP original con el que se obtenga una mejora respecto a una división en subredes equilibradas en cuanto a la cantidad de elementos? Esta es la respuesta que buscamos en este trabajo.

La RdP dividida tiene que expresar y preservar el comportamiento de la RdPnA. Esto tiene dos aspectos: la regla de disparo de las transiciones y la relación de la red con los eventos.

El primer aspecto ha sido mostrado en[9] . En cuanto al segundo, es menester que se mantengan las relaciones entre las transiciones y los eventos que disparan las transiciones. Además las comunicaciones que generan los disparos de las transiciones deben poder ser informadas. Es decir que la división no introduzca ambigüedad en los eventos generados por el disparo de una transición, esto ha sido considerado y resuelto en [6].

Finalmente el esquema de prioridades tiene que ser posible de implementar y facilitar la ejecución en paralelo. Para esto, las transiciones en conflicto en cada subred, son resueltas por un esquema de prioridad local. Mientras, las transiciones de borde tienen más prioridad que las locales de cada subred y además un esquema de prioridades propio.

La Figura 1, muestra una RdP dividida por las transiciones. Las transiciones divididas se encontrarán presentes en cada una de las subredes resultantes, la comunicación se realiza por las transiciones de borde, que son los límites de las subredes. Podemos entonces decir que existe una relación entre todas las partes de cada transición cortada, que originalmente constituía una única transición. Estas partes están comunicadas y se evalúan y disparan de forma simultánea.

Hay que notar que el hecho de dividir las redes de esta forma, genera una jerarquía explícita de redes, con un concepto más amplio, que llamamos ¨Redes de Petri Jerár-

quicas Divididas por Transiciones¨. De esta división resulta una reducción de recursos con respecto a la red original.

Por esto consideramos al sistema resultante como una HPN, lo que facilita la interpretación y la implementación en distintos dispositivos intercomunicados como se realiza en [13].

3.2 Consideraciones para obtener Redes Jerárquicas Divididas por Transiciones

Lo primero que observamos, es que sólo existe una red. No hay una red global y subredes, sino que todas las partes de la red o subredes tienen la misma jerarquía y se relacionan entre ellas mediante transiciones de borde. No obstante a cada parte de la red obtenida por la división la llamaremos subred.

1. Las transiciones distribuidas están constituidas por el conjunto de todas las transiciones de borde, que son el resultado de la división de una misma transición de la red total.
2. Para obtener el estado global del sistema consideramos todas las plazas en conjunto, es decir que hay que tener el marcado de cada subred.
3. Cuando se ejecuta una transición distribuida, se deben ejecutar todas las transiciones de borde de todas las subredes a las que pertenece, con el fin de garantizar la semántica del disparo. Esto es, cuando todas las transiciones distribuidas que pertenecen a una misma transición están sensibilizadas, esta transición está en condiciones de dispararse.
4. Con el fin de que un disparo de una transición distribuida no genere conflicto en una subred (con otras transiciones en conflicto), solo se permite la ejecución de una sola transición distribuida por ciclo de disparo.
5. La red se divide cortando las transiciones, de modo que las transiciones elegidas como límite entre dos o más subredes quedan partidas, constituyendo una transición de borde para cada una de las subredes. Estas transiciones de borde, en cada subred estarán relacionadas, constituyendo una transición distribuida.
6. Como cada subred mantiene su funcionamiento por separado, se aumenta el nivel de paralelismo en la ejecución del sistema, pudiendo darse como máximo que todas las subredes ejecuten un disparo (en transiciones que no son de borde) en el mismo ciclo.
7. La matriz binaria de relación de transiciones distribuidas, indica como las transiciones distribuidas se reagrupan para constituir la transición original en cada subred.
8. Para evitar excesiva comunicación entre las subredes, ante un conflicto, las transiciones distribuidas tienen mayor prioridad de disparo que cualquier transición interna de la subred.
9. Si es necesario que una transición interna sea de mayor prioridad que una transición de borde en conflicto, se la puede definir como una transición distribuida que solo está relacionada con la subred a la que pertenece.

Ahora introducimos el concepto de localidad estructural para guiar la división de la red, esto es: "*dividir las redes por las transiciones agrupando elementos en los subconjuntos con características comunes en cada subred*". Estos elementos son: arcos con peso uno, arcos con pesos mayor a uno, transiciones con tiempo, transiciones temporizadas y otros tipos de arcos.

4 Un ejemplo: División de una red con N lectores y un escritor en tres subredes

En la Figura 1. Red de productor consumidor dividida en tres subredes, se muestra la RdP que modela a N lectores y un escritor, que ha sido dividida en tres subredes.

Originalmente se parte de una RdP de 7 plazas (m) y 6 transiciones (n) más la matriz de prioridades (nxn), para esta RdP el PP requiere 372 bits.

$$PN = (mxn)8 + (nxn) = 372$$

Dividiendo ésta RdP, como lo muestra la Figura 1. Red de productor consumidor dividida en tres subredes , en tres sub redes los recursos necesarios son 290 bits. Lo que muestra en la Tabla 1.

Tabla 1: Recursos RdP con N lectores y un escritor, dividida en tres subredes

	Plazas	Transiciones	Matriz de P.	Recursos
Red0 y Red1	3 c/u	3 c/u	3x3	81 c/u = 162
Red2	1	4	4x4	48
Matriz de relación	Son 8 las transiciones de borde, con cuatro sub redes 4x4x3			48
Matriz de P. de T. de borde.	4*4			16
Total de recursos de la red divida- Reducción de recursos 79%				290/79%

Por lo que la división tiene una ganancia de recursos es del 21%.

Si además observamos que las subredes uno y dos solo necesitan brazos con peso uno y menos uno en la matriz, por lo que se requiere dos bit, mientras que en la subred tres se requiere de brazos con peso N, para lo cual se usan 8 bit. Los recursos necesarios son 55,9%, esta ganancia adicional es resultado de aplicar la localidad estructural

5 Caso de Aplicación

A continuación aplicamos el concepto de localidad estructural a un caso de modelado con RdP para el análisis de desempeño de aplicaciones de búsqueda WEB.

El caso, se ha tomado de [14] (A Modeling Technique for the Performance Analysis of Web Searching Applications). En este trabajo se investiga el comportamiento de un sistema cliente /servidor, haciendo uso de un modelo no-Markoviano realizado

con RdPnA para obtener los índices de rendimiento. También se hizo uso de un entorno experimental, con el fin de obtener mediciones reales que son utilizadas para validar el modelo analítico.

5.1 Modelo de RdP para el cálculo de tiempo de respuesta del sistema Cliente Servidor

La Figura 2 muestra la RdP, presentada por el autor, que ha sido dividida en 5 subredes. La división se ha realizado con el criterio de localidad estructura, agrupando las transiciones del mismo tipo y además respetando los bloques de procesos de la red original.

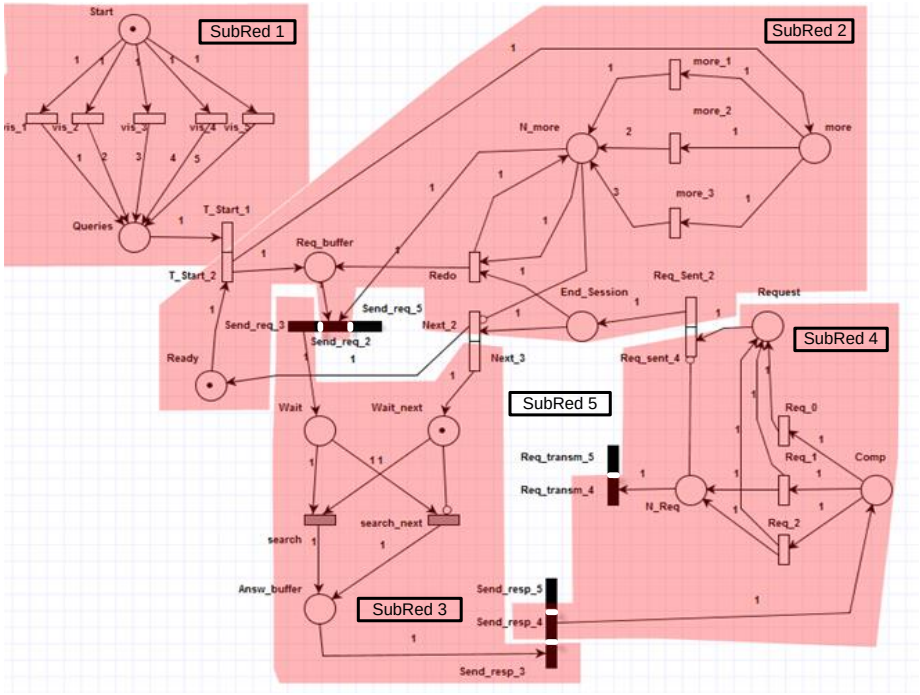


Figura 2: Modelo con RdP dividida para el análisis de búsqueda WEB

En la Figura 2 podemos ver tres tipos de transiciones: transiciones inmediatas representadas por un rectángulo vacío, transiciones temporizadas representadas por un rectángulo gris y las con retardo que son representadas por un rectángulo negro.

Hay que hacer notar que el PP con tiempo [5, 15] original no es capaz de soportar los dos tipos de transiciones con tiempo simultáneamente, mientras que el HPP soporta los dos tipos de transiciones con tiempo en distintas subredes. En concordancia con el concepto de localidad estructural, desarrollado en este trabajo, se ha dividido la red agrupando los distintos tipos de transiciones temporales en una misma subred,

Los recursos empleados para implementar el procesador de Petri temporal (PPT) sin dividir son: 3220.

Plazas= 13, transiciones= 20, comparadores de tiempo 2 x 20 y contadores de tiempo= 1 x 20. Cada comparador y contador de tiempo requiere de 32 bit.

Cantidad de bit para soportar el peso máximo de los brazos 4 bit (incluyendo el signo).

Para simplificar el cálculo de recursos se ha tomado como unidad un FF por bit, por lo que se requiere de 3220 bit para implementar el PP sin dividir, mientras que para implementar el PP jerárquico (HPNP) se requiere de 1384 bit, por lo que se obtiene una ganancia de 232,6%.

Tabla 2: Recursos para la RdP de la figura 2 sin dividir y dividida

	PPT sin dividir	Recursos empleados para implementar las 5 subredes en el HPP				
		Transiciones de la red divididas 6				
		Subred 1	Subred 2	Subred 3	Subred 4	Subred 5
Plazas	13	2	5	3	3	0
Transiciones	20	11	10	8	9	6
PxT (4bit de peso)	13x20x4	2x11x4	5x10x4	3x8x1	3x9x1	1x8x1
Arcos inhibidores	13x20x1	no	5x10x1	3x8x1	3x9x1	no
Transiciones divididas en la subred	NC	1	3	3	2	6
Transiciones no divididas	NC	5	4	2	3	0
Contadores de Tiempo, 32bit	20x32	0	0	8x32	0	6x32
Comparadores de Tiempo 32 bit	2x20x32	0	0	8x32x2	0	0
Recursos necesarios	3220	88	250	792	54	200
subtotales	3220		1384			
Ganancia	232,6%					

6 Conclusión y trabajos futuros

Como conclusión del presente trabajo podemos destacar que: la aplicación del concepto de localidad estructural y división de la red impactan directamente en la reducción de los recursos de hardware empleado para instanciar el IP-Core. Si bien no es posible hacer una generalización, para el caso de análisis que no es el óptimo se ha obtenido un ahorro del 57% en recursos. Esto con respecto a un procesador que instancie todas las posibilidades para todas las transiciones.

La división en subredes, atendiendo a que cada subred sea de una semántica temporal común, permite instanciar RdP heterogéneas haciendo uso de menos recursos de hardware puesto que cada subred tiene los recursos estrictamente necesarios.

Podemos inferir que mientras más heterogéneas sea la semántica de las transiciones, más rectangulares sean las matrices de subredes y menor la cantidad de transiciones divididas mayor es el beneficio.

Según los resultados obtenidos, en las pruebas realizadas usando el IP Core HPNP para sistemas concurrentes y paralelos con tiempo, la mejora en tiempo de respuesta con respecto al uso de semáforos, se traduce en mejoras entre el 40% y el 60%. Esto son los mismos resultados obtenido en los PP.

Finalmente mencionamos, que el uso de la localidad estructural mejora la versatilidad de las RdP para modelar sistemas concurrentes y paralelos, lo que hace que el sistema resultante sea flexible ante cambios en la implementación.

Como trabajo futuro se está desarrollando un algoritmo para la división automática de la red.

REFERENCIAS

1. R. A. B. David R. Martinez, M. Michael Vai, *High Performance Embedded Computing Handbook A Systems Perspective*. Massachusetts Institute of Technology, Lincoln Laboratory, Lexington, Massachusetts, U.S.A.: CRC Press, 2008.
2. M. Domeika, *Software Development for Embedded Multi-core Systems*. 30 Corporate Drive, Suite 400, Burlington, MA 01803, USA Linacre House, Jordan Hill, Oxford OX2 8DP, UK, 2008.
3. M. Diaz, *Petri Nets Fundamental Models, Verification and Applications*. NJ USA: John Wiley & Sons, Inc, 2009.
4. M. Pereyra, M. A. N. Gallia, and O. Micolini, "Heterogeneous Multi-Core System, synchronized by a Petri Processor on FPGA," in *IEEE LATIN AMERICA TRANSACTIONS*, 2013, pp. 218-223.
5. J. N. y. C. R. P. Ornaldo Micolini, "IP Core Para Redes de Petri con Tiempo," *CASIC 2013*, pp. 1097-110, 2013.
6. O. Micolini, "ARQUITECTURA ASIMÉTRICA MULTI CORE CON PROCESADOR DE PETRI," Doctor, Informatica, UNLaP, La Plata, Argentina, 2015.
7. O. Micolini, J. Nonino, and C. R. Pisetta, "IP Core Para Redes de Petri con Tiempo," in *CASIC 2013*, 2013, pp. 1097-110.
8. N. G. M. Pereyra, M. Alasia and O. Micolini, "Heterogeneous Multi-Core System, synchronized by a Petri Processor on FPGA," *IEEE LATIN AMERICA TRANSACTIONS*, vol. 11, pp. 218-223, 2013.
9. O. Micolini, E. Arlettaz, S. H. B. Baudino, and M. Cebollada, "Reducción de recursos para implementar procesadores de redes de Petri," in *en Jaiio 2014*, 2014.
10. S. Pissanetzky, *Sparse Matrix Technology*. Bariloche, Argentina: Centro Atómico Bariloche, 1984.
11. S. Kestury, J. D. Davisz, and E. S. Chungz, "Towards a Universal FPGA Matrix-Vector Multiplication Architecture," *Field-Programmable Custom Computing Machines (FCCM), IEEE 20th Annual International Symposium on*, 2012.
12. G. H. Golub and C. F. V. Loan, *Matrix Computations* Johns Hopkins, 2012.
13. R. Pais, Barros, J.P. ; Gomes, L., "From Petri net models to C implementation of digital controllers " *Emerging Technologies and Factory Automation. ETFA 2005. 10th IEEE Conference on*, 2010.
14. M. Scarpa, A. Puliafito, M. Villari, and A. Zaia, "A modeling technique for the performance analysis of Web searching applications," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 16, pp. 1339-1356, 2004.
15. M. Orlando, J. Nonino, and C. R. Pisetta, "IP Core for Timed Petri Nets," in *CASECongreso Argentino de Sistemas Embebidos*, 2013, pp. 3-8.

Módulo gráfico de un Visualizador de Estructuras de un Sistema Operativo Educativo a través de GDB-Stub

Graciela De Luca[1], Martín Cortina[1], Nicanor Casas[1], Esteban Carnuccio[1], Sebastián Barillaro[1], Daniel Giulianelli[1], Pablo Barboza Carvalho[1], Ezequiel Calaz[1], Gabriela Medina[1]

[1] Universidad Nacional de La Matanza,
San Justo, Buenos Aires Argentina
{gdeluca, mcortina , ncasas, ecarnuccio, sbarillaro, dgiunlian}@ing.unlam.edu.ar
pbarbozacarvalho@hotmail.com, gmedina190@gmail.com, aecalaz@gmail.com

Abstract. En este documento se exponen los mecanismos desarrollados para la construcción de un visualizador de estructuras internas de un sistema operativo didáctico, tanto para la comunicación entre el sistema operativo y el visualizador, como así también los módulos gráficos que permiten ver las estructuras internas durante la ejecución. Estos han sido de gran utilidad para obtener como resultado el estado actual de los módulos de la aplicación, que se encargarán de representar gráficamente los componentes del sistema. Se detallan las metodologías empleadas para que las interfaces GUI del graficador puedan representar la información obtenida de SODIUM a través de GDB y scripts desarrollados en el lenguaje Python. El objetivo final es que el visualizador sea de utilidad en el ámbito académico, facilitando el aprendizaje a los estudiantes de materias afines.

Keywords: GDB, GDB-Stub, Python, Front-End, GDM/MI, Estructuras de un Sistema Operativo

1 Introducción

El proyecto de investigación tiene como objetivo final la construcción de una herramienta que facilite el aprendizaje y enseñanza sobre del funcionamiento interno de un sistema operativo convencional.

Esta aplicación está siendo primeramente desarrollada en base al Sistema Operativo SODIUM [7] debido a que el equipo de trabajo ya se encuentra familiarizado con el mismo, pero se espera que el resultado final sea compatible o adaptable a otros sistemas.

Es importante señalar que la premisa fundamental de este programa es la de permitirle al usuario visualizar gráficamente las estructuras lógicas utilizadas por el sistema operativo durante su ejecución a través de una conexión serial.

Seguidamente se describen los mecanismos que se han empleado hasta la fecha a lo largo del proyecto para intentar alcanzar la meta establecida.

Este documento se ha dividido en dos secciones. En la primera parte, se explica cómo se ha implementado en su totalidad el componente GDB-STUB [9] en SODIUM, gracias al cual ahora es posible conectar al mismo el depurador remoto GDB. También se describe el uso que estamos haciendo de una característica muy

interesante de este depurador, que consiste en ser altamente automatizable por medio de scripts. Con esto, logramos obtener el estado de las estructuras internas del sistema operativo desde otra terminal.

En la segunda sección se detalla el estado actual de los módulos gráficos que conformarán el visualizador, y de qué manera empleamos GDB y Python para su construcción.

2 GDB-STUB y scripts de Python

El pilar principal de este proyecto consiste en utilizar GDB para poder acceder al estado de las estructuras del sistema Operativo desde otra terminal. GDB ofrece dos herramientas para debuguear en forma remota: GDB Server y GDB-Stub. Estos mecanismos son utilizados con mucha frecuencia en los desarrollos de sistemas embebidos, donde la única forma de analizar la ejecución de un programa sobre un dispositivo físico es en forma remota. Como SODIUM no posee las características necesarias para implementar GDB-Server dado que carece de muchas bibliotecas que utiliza ese programa para poder funcionar, se decidió desarrollar el módulo GDB-Stub en este Sistema Operativo [2]. Debido a la complejidad de la implementación, ésta fue llevada a cabo en forma gradual. Las etapas iniciales de dicha adaptación se describen en [7] y [8]. Sin la implementación de este módulo en SODIUM la obtención de los estados de las estructuras del sistema operativo durante su ejecución habría sido mucho más extensa y dificultosa, dado que habríamos tenido que desarrollar nosotros mismos el trabajo que realiza el debugger para poder conocer el valor de los datos en tiempo real. Adicionalmente, el resultado final habría carecido de la flexibilidad que esta solución brinda.

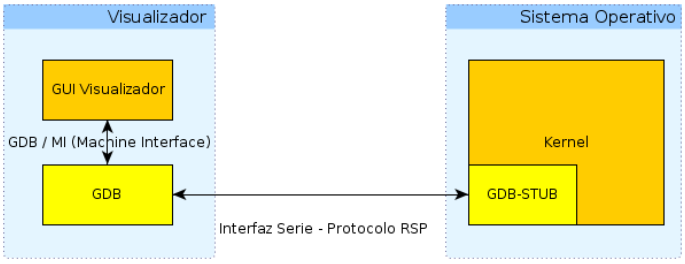


Fig. 1. Relación entre los componentes involucrados.

Se puede observar que se sigue una arquitectura cliente-servidor, donde el lado cliente representa al visualizador y el lado servidor representa al sistema operativo.

Del lado servidor se puede ver que el módulo GDB-STUB que se encuentra integrado en el kernel del sistema operativo. En el lado cliente, se encuentra GDB junto al Visualizador y a los script de Python [5]. Más adelante se hablará acerca del módulo de GDB/MI del lado cliente.

Seguidamente se describen brevemente las principales adaptaciones que se realizaron en el código de SODIUM para poder integrar el módulo de GDB-stub.

Funciones agregadas en Sodium para implementar el módulo GDB-Stub.

En el código de SODIUM se comenzó construyendo un módulo *stub* en forma incremental, que incorporaba las funcionalidades esenciales que ofrece el archivo original *i386-stub.c* de GDB. Inicialmente no fueron desarrolladas en su totalidad, debido a conflictos con la implementación preliminar de driver de puerto serie que poseía nuestro sistema operativo.

Luego de varias modificaciones en el código de SODIUM, se consiguió implementar en su totalidad el módulo GDB-Stub en ese sistema. Una de las adaptaciones especiales que se debieron realizar, consistió en la creación de funciones especiales en diversos sitios dentro de los archivos fuentes de SODIUM.

- **set_debug_traps():** Esta función es usada para configurar los handlers que capturan la excepción 3 [4] [7], que ocurre cuando se ejecuta un breakpoint, y la excepción 1, qué sucede cuando se debuggea paso a paso. Dicho de otra forma, dentro de *set_debug_traps()* se invoca a la función encargada de modificar los descriptores dentro de la IDT que están asociados a las excepciones antes mencionadas. De manera que cuando ocurran de dichos eventos, el sistema automáticamente ejecuta los handlers que provee GDB para manejar esas excepciones. Es importante mencionar que fue necesario invocar a *set_debug_traps* al iniciar la función *main()* de SODIUM. Este hecho ocurre después de haber establecido la CPU en modo Protegido y de haber inicializado el driver de puerto Serie, condición fundamental para la comunicación con GDB.
- **exceptionHandler ():** Función intermedia que provee GDB-Stub, cuya tarea es hacer de intermediario entre *set_debug_traps()* y la función que modifica la IDT.
- **breakpoint ():** Internamente, esta función al ser invocada ejecuta un *"int 3"*. Esta línea de código assembler produce una excepción 3, y principalmente es utilizada para conectar GDB y con el sistema operativo SODIUM. Por dicho motivo fue necesario invocarla en el main luego *set_debug_traps.*
- **putDebugChar() y getDebugChar():** Se debió adaptar el driver de puerto Serie desarrollado en SODIUM, generando dos funciones para transmitir de a un carácter a la a vez entre la terminal cliente y Servidor, en lugar de una palabra completa, Esta adaptación fue necesaria dado que el módulo GDB-stub se encuentra diseñada para utilizar de esa manera el protocolo RSP [3].

Además, para lograr el funcionamiento del módulo GDB-Stub en su totalidad en SODIUM, fue necesario realizar un mecanismo híbrido, para poder utilizar el puerto serie por medio de su driver. El cual consistió en permitir a través de interrupciones la detención del sistema operativo por medio de la consola de GDB, De forma tal, que cuando SODIUM reciba en código ASCII un valor en hexadecimal "03" (carácter de control ctrl+c), el kernel invoque automáticamente a la función principal *handle_exception()* [7] para que controle la ejecución del sistema. Posteriormente, una vez que el stub toma el control, la transferencia de datos con GDB se lleva a cabo en modo polling.

Establecer la conexión entre GDB con el Stub de SODIUM en forma remota.

Para poder conectar GDB con SODIUM primeramente es necesario que el usuario inicie el sistema operativo, ya que funciona como un servidor. Una vez que el kernel inicializa los drivers del puerto serie e invoca a la función set_debug_traps(), ejecutará la función breakpoint() que le dará el control al módulo GDB-Stub, así el sistema quedará en una espera activa hasta que se establezca un enlace con GDB.

Por otra parte, el usuario desde el del lado cliente deberá ejecutar un script, que finalmente establecerá la conexión entre SODIUM y el debugger. Por consiguiente, en la siguiente figura se muestran los argumentos utilizados en el script, con los que se ejecuta el programa GDB para que se conecte en forma automática con el sistema operativo.

```
gdb 'target remote localhost:12345' -b $Baudios --
symbols='kernel/main.ld' --command='comandos_nuevos.gdb'
```

Figura 2. Ejecución de GDB para establecer conexión con SODIUM

El primer argumento 'target *remote localhost:12345'* establece la conexión entre GDB y el sistema operativo, cuando este último se encuentra esperando activamente en su stub. A este parámetro se le debe indicar la IP y el puerto en donde estará escuchando el servidor. El segundo argumento, '*-b'* indica la velocidad en baudios que se va a utilizar durante la transferencia de los datos. El tercer argumento '*--symbols'* es utilizado para señalar cuál es el archivo que contiene la tabla de símbolos del kernel del sistema operativo. Finalmente el último argumento *--command*, indica en qué archivo se encuentran los comandos especializados de GDB que son creados particularmente para el visualizador. Más adelante en este documento se describe de qué manera ha sido utilizado en SODIUM.

Extensión de comandos GDB para el visualizador de estructuras

Una utilidad importante que ofrece GDB es la de permitirle al programador generar nuevos comandos a partir de los ya existentes en el debugger. Esta característica es esencial para poder desarrollar el visualizador, ya que nos permite generar el enlace entre GDB y el graficador a través de la concepción de nuevas instrucciones según nuestras necesidades. GDB ofrece distintos formas de extender su set de comandos[2]. Pero en este documento se mencionan únicamente los mecanismos que se han utilizado hasta la fecha durante el desarrollo del proyecto. Por ese motivo, a continuación se describen los métodos empleados en el transcurso de la investigación:

- **Archivo de Script de Comandos de GDB**

GDB le permite al usuario definir una secuencia de comandos específicos como una unidad y que luego este conjunto pueda ser ejecutado bajo un nuevo nombre de comando. Para su programación el debugger posee su propio lenguaje de scripting, con sus propias estructuras de control, como por ejemplo if, while, for y funciones. De esta forma el usuario tiene la posibilidad de generar bibliotecas con sus propios comandos de GDB. En la figura 2, se puede observar el argumento *--command* que se

le pasa a GDB, al ser ejecutado desde la consola de Linux. Este parámetro se utiliza para indicarle al debugger el nombre del archivo que contendrá los comandos que fueron personalizados por el usuario.

- **Archivo de Script de Python en GDB**

Una de las características más importantes que ofrece GDB es la de poder ejecutar scripts de Python desde su consola, pudiendo extender el conjunto de comandos del debugger utilizando este lenguaje. Para poder aprovechar dicho beneficio, fue necesario recompilar el código fuente de GDB, utilizando el flag --with-python durante su configuración. La forma de utilizar código Python dentro de GDB se puede realizar de dos maneras distintas. La primera forma es invocando al intérprete de Python desde el prompt de GDB. El segundo método es ejecutando el comando de GDB ***source*** junto al nombre del archivo del script de Python, por ejemplo "*source script_python.py"*

Cabe destacar, que GDB ofrece diferentes APIs para invocarlas en Python. Las cuales permiten ejecutar un comando específico de GDB dentro de un script desarrollado en ese lenguaje. De esta manera por ejemplo, se puede conocer desde Python el valor de una estructura del Sistema Operativo SODIUM durante su ejecución. Utilizando para ello la función *gdb.execute("print pstuPCB" ,0,1)*

Por consiguiente en base a las herramientas que ofrece GDB para poder extender sus comandos, se empezaron a desarrollar los módulos gráficos del Visualizador. Por lo tanto, en la segunda sección de este documento, se describen el estado actual del desarrollo de dicho componentes y de qué manera han sido utilizados estos mecanismos de extensión.

3 Desarrollo de Módulos Gráficos del Visualizador de Estructuras de un Sistema Operativo

Como se mencionó anteriormente, el objetivo de este proyecto es generar una herramienta que le permita al usuario poder observar las distintas estructuras que utiliza el sistema operativo SODIUM durante su ejecución. De forma tal, que le facilite a los estudiantes el aprendizaje teórico práctico del funcionamiento de los sistemas operativos convencionales. Por dicho motivo actualmente el desarrollo del visualizador se centra en dos módulos:

- **Módulo de visualización de estructuras de Sodium**

Este componente del visualizador se encargará de mostrarle al usuario, a través de interfaces GUI, el estado de las siguientes estructuras que utiliza el sistema operativo en un momento determinado: IDT, GDT, TSS y PCB. Además representará gráficamente el mapa de memoria que maneja el sistema operativo. Mostrando para ello la ubicación exacta de los segmentos en un instante específico.

- **Módulo de visualización de Diagrama Temporal**

Este módulo pretende permitirle al usuario poder observar en forma gráfica mediante el visualizador, los diferentes estados que van adquiriendo los procesos que van ejecutándose en SODIUM a lo largo de su vida. De esta se intenta que el

estudiante pueda observar la ejecución de distintos algoritmos de planificación de CPU en forma visual.

Es importante mencionar que en este documento se describe en detalle el desarrollo actual del primer módulo antes mencionado.

Embeber GDB en Python y GDB/MI

Una dificultad encontrada durante el desarrollo, consistió en la imposibilidad de importar la biblioteca que utiliza GDB en un programa desarrollado íntegramente en Python (*"import gdb.py"*). Lo que resultó ser un impedimento para construir un ejecutable del visualizador desarrollado en ese lenguaje, que permita acceder a la información que ofrece SODIUM a través de GDB. Este percance es debido a la existencia de un fallo en el código fuente entre GDB y Python [10], que impide hacer esto, que aún no ha sido solucionado. Por consiguiente, para intentar subsanar dicho obstáculo, se determinó que era conveniente basar el desarrollo de las interfaces gráficas ejecutando los scripts de Python del visualizador desde la consola de GDB. Utilizando para ello el comando *source*, como se había mencionado anteriormente.

Además se descubrió que GDB ejecuta sus comandos en un único hilo de ejecución. Por lo que al ejecutar un script de Python que muestra una interfaz GUI desde la consola del debugger, se imposibilita poder seguir la ejecución normal de SODIUM. Debido a que en ese momento el debugger se encuentra ejecutando el script de Python. Por dicho motivo se decidió analizar los mecanismos que utilizan los Front Ends que usan GDB y extraer su funcionamiento elemental para poder aplicarlo luego al visualizador. Estas herramientas permiten ejecutar una interfaz gráfica y en simultáneo continuar debuggeando una aplicación utilizando GDB/MI [2]. GDB/MI hace de intermediario entre la comunicación de GDB y los Front End, mediante una interfaz basada en texto protocolizado. Por dicho motivo, se está analizando el funcionamiento de un Front End en especial desarrollado en Python, denominado Pyclewn [11]. No obstante paralelamente, se está desarrollando cada módulo gráfico en forma modular como un conjunto de comandos de GDB. En donde cada uno de ellos ejecuta una interfaz GUI que realiza una tarea determinada. Luego una vez finalizado este proceso, se pretende integrar todos los submódulos gráficos aplicando los conceptos que adquiridos sobre el funcionamiento de los Front Ends.

En consecuencia, a continuación se describen el estado actual de las interfaces gráficas que componen el módulo de visualización de estructuras de SODIUM.

Módulo de visualización de estructuras de Sodium

A los fines de poder visualizar gráficamente las estructuras internas del sistema operativo SODIUM, se generó un archivo de comando de script de GDB (llamado "*comandos_nuevos.gdb").* El cual, tiene como objetivo generar un conjunto de comandos personalizados con la intención de extender las utilidades que ofrece el debugger. Como se muestra en la figura 2, el nombre del script debe ser pasado como parámetro a GDB al ser ejecutado desde Linux. Dentro de este archivo, se generaron nuevos comandos utilizando las técnicas de extensión previamente detalladas. En algunas de estas instrucciones, se invocan a archivos con script de Python, que se encargan de mostrar por pantalla la información solicitada por el usuario a través de interfaces GUI. Para ello, el código de las ventanas gráficas fue escrito en PyQT [6],

de acuerdo a como se había señalado en [8]. Además como SODIUM está desarrollado en base a la arquitectura X86, la información es representada en concordancia a lo especificado por INTEL [1]. A partir de lo explicado, en este script se crean las siguientes utilidades:

- **Comandos para visualizar la IDT, GDT, TTS y PCB**

Se creó un set comandos en Python que, al ser invocados mediante la instrucción *guiGDB*, desde el prompt de GDB, mostrarán la siguiente interfaz gráfica al usuario. Permitiéndole así observar la composición de las estructuras internas de SODIUM.

Figura 3. Interfaz GUI principal para visualizar las estructuras de SODIUM

En la figura anterior se pueden observar distintas opciones: PCB, GDT, IDT Y TSS. Si el usuario selecciona una de las alternativas, el visualizador mostrará otra ventana que permitirá al usuario indicar el registro de la estructura que se desea visualizar.

En consecuencia, si el usuario elige el botón GDT podrá observar lo siguiente:

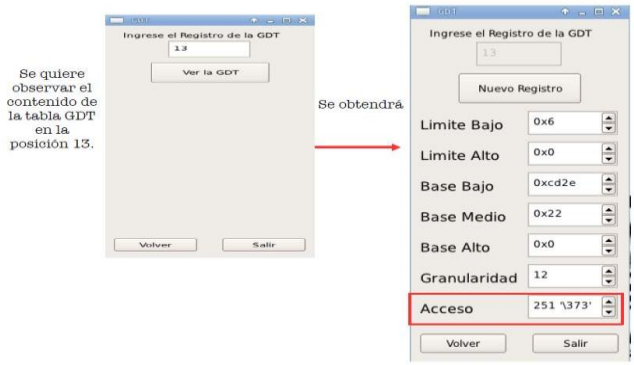


Figura 4. Interfaz GUI con la composición de la GDT

En cambio, si se escoge un descriptor determinado de la IDT se mostrará dicha estructura de la siguiente manera:

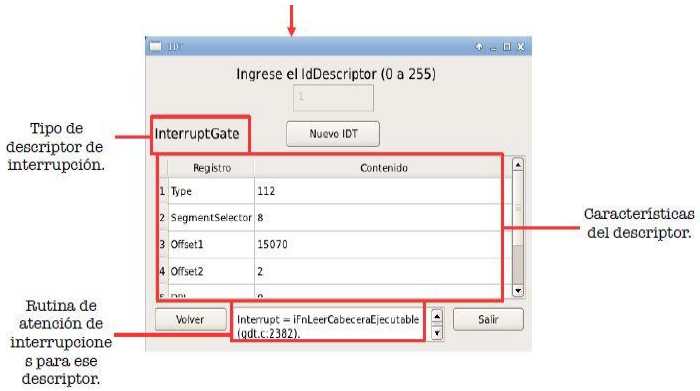


Figura 5. Interfaz GUI con la composición de la IDT

Si por otra parte el usuario desea ver la composición del PCB asociado a un determinado proceso, podrá conocerlo ingresando su número de PID en el campo correspondiente.

Figura 6. Interfaz GUI con la composición del PCB de un proceso

- **Comandos para visualizar el estado del mapa de memoria del sistema operativo en un momento determinado**

Con el objetivo de poder observar la ubicación en memoria de los segmentos que conforman cada componente que administra el sistema operativo, se desarrollaron un set de instrucciones de GDB que permiten ver el estado del mapa en un momento determinado. Por lo tanto, ingresando el comando *mapamemoria* en la consola de GDB, se imprimirá en modo textual el estado del mapa de memoria que tiene la terminal donde se esté ejecutando SODIUM en ese momento. Esto se puede apreciar en la siguiente figura.

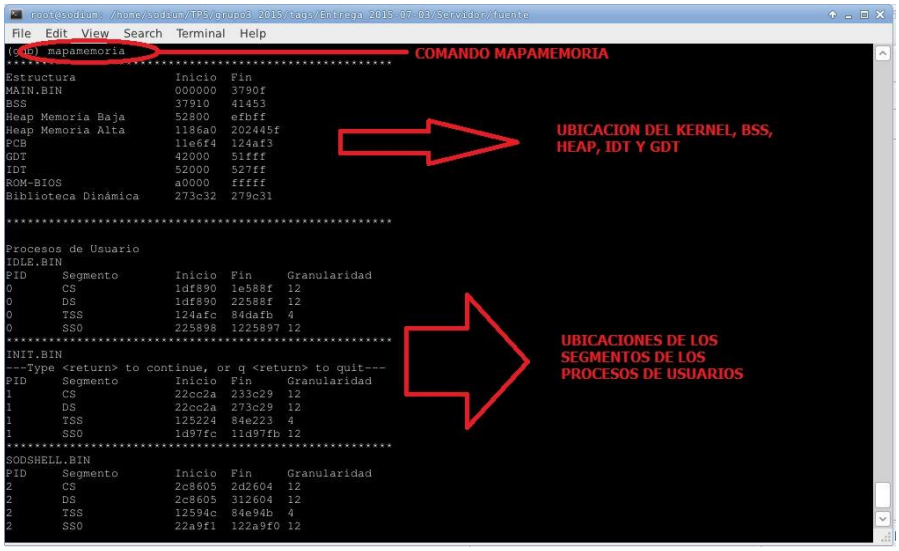


Figura 7. Mapa de memoria de Sodium

Como se observa en el gráfico, por cada proceso se imprimen la dirección inicial y final en donde se ubican sus segmentos de código, datos, TSS y SS0 en la memoria principal. Además simultáneamente, se imprimen las direcciones del Kernel de SODIUM (Main.bin), así como también su BSS, Heap y las tablas GDT e IDT.

- **Comandos para establecer nuevos Puntos de instrumentación**

Fue necesario investigar una nueva forma de implementar los puntos de instrumentación en el código de S.O.D.I.U.M., debido a que se dificultaba poder aplicar correctamente la metodología que se había mencionado en [7], capturando desde GDB los mensajes emitidos por la función *vFnLog* del Sistema Operativo. Por consiguiente, se debió generar otro sets de comandos de GDB. Los cuales definen una cierta cantidad de breakpoints en determinadas partes del código de SODIUM, asociándolos a un evento del sistema operativo.

Luego cuando se ejecuta uno de estos breakpoints durante la ejecución, el debugger captura dicha ocurrencia y determina el evento que ha acontecido notificándoselo al usuario a través de una ventana gráfica. Cabe mencionar que esta técnica aún no se encuentra implementada en su totalidad, pudiendo sufrir cambios en un futuro.

4 Conclusiones

Hasta la fecha en esta investigación se consiguieron grandes avances en el desarrollo de las interfaces gráficas del visualizador. Dado que se pudieron mostrar los datos de las estructuras internas que utiliza SODIUM durante su ejecución utilizando PyQT. Gracias a la implementación del módulo GDB-Stub, en el código

del sistema operativo, se pudieron obtener estos avances. De forma tal, que nos permitió controlar totalmente su ejecución. Si bien por el momento cada interfaz gráfica es visualizada en forma separada, ejecutando un comando personalizados de GDB. Se pretende integrar todas las ventanas GUI en una única aplicación, empleando los mecanismos que utilizan los Front Ends que actualmente se están analizando. Pudiendo así conseguir ejecutar las interfaces del visualizador y GDB en forma simultánea. Otro logro obtenido consistió en poder ver en forma textual desde el debugger la composición del mapa de memoria del sistema operativo durante su ejecución. Con lo cual, el paso siguiente consistiría en representar dicha información mediante interfaces GUI. De esta manera se está desarrollando una aplicación que les permita a los estudiantes observar el funcionamiento interno de un sistema operativo real en forma gráfica.

5 Referencias

1. Intel: "Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3: System Programming Guide": pp 65-6,413 (2011)
2. Richard Stallman, Roland Pesch, Stan Shebs "Debugging with gdb" Free Software Foundation, Tercera Edición: pp 319-449 y 461-539 (2015)
3. Bill Gatliff "Embedding with GNU: the GDB Remote Serial Protocol" revista Embedded Systems Programming, Noviembre 1999
4. Prasad Krishnan: "Hardware Breakpoint (or watchpoint) usage in Linux Kernel", IBM Linux Technology Center, Canada: pp 1-10 (2009)
5. Guido van Rossum,"El tutorial de Python", Editorial: Fred L. Drake Jr. Septiembre 2009
6. Mark Summerfiled, "Rapid Gui Programming with Python and QT", Editorial: Prentice Hall, Año: 2007
7. Graciela De Luca, Martín Cortina, Nicanor Casas, Esteban Carnuccio, Sebastián Barillaro Sergio Martín, Gerardo Puyo, " Visualizador de Estructuras de un Sistema Operativo Educativo", Congreso CACIC (2014)
8. Graciela De Luca, Martín Cortina, Nicanor Casas, Esteban Carnuccio, Sebastián Barillaro Sergio Martín, Gerardo Puyo, "Desarrollo de un prototipo para un visualizador de estructuras de un sistema operativo en ejecución a través de la comunicación serial", Congreso WICC(2015)
9. The GNU Project Debugger,https://sourceware.org/gdb/onlinedocs/gdb/Remote-Stub.html
10. The Cliffs of Inanity, http://tromey.com/blog/?p=806
11. Pyclewn, http://pyclewn.sourceforge.net/

Integración de un periscopio militar a un simulador de realidad virtual para entrenamiento táctico

José A. Marone[1], Marcelo A. Tosini[1]

[1] Grupo de Sistemas Digitales, Instituto INTIA, Facultad de Cs Exactas, UNCPBA, Tandil, Buenos Aries, Argentina
{marone, mtosini}@exa.unicen.edu.ar

Resumen. El presente trabajo describe tareas de actualización recientemente realizadas en la Escuela de Submarinos y Buceo (ESyB) de la Armada Argentina con sede en la Base Naval de Mar del Plata en el adiestrador de operarios de submarinos en el marco del proyecto de transferencia tecnológica (RCS 1112/13 UNCPBA-ESyB). En particular se trabajó en la modernización de un simulador de periscopio, desarrollado originalmente por este mismo grupo de investigadores en el año 2003. Los distintos movimientos y controles asociados a los comandos del periscopio son convertidos a una interfaz clara de alto nivel y enviados al motor de simulación desacoplando así ambos sistemas. Este artículo muestra un panorama general de la arquitectura del sistema de control del periscopio y describe las soluciones implementadas a nivel de hardware y software. Se consiguió implementar un sistema embebido adecuado al presupuesto, con componentes accesibles a nivel nacional y con tiempos de respuesta por debajo de lo exigido por el motor de simulación.

Palabras clave: Sistemas Embebidos, Actualización de Sistemas Militares, Codiseño de hardware/software

1 Introducción

Sin dudas la actualización o incorporación de nuevas tecnologías en sistemas antiguos no es una tarea simple. En muchos casos, y particularmente en sistemas de defensa, se debe interactuar con sistemas de diversas índoles (eléctricos, electrónicos, mecánicos, electromecánicos, ópticos, etc.) haciendo que las tareas de actualización y mantenimiento sean complejas y costosas [1]. En muchos casos los países en vías de desarrollo no pueden afrontar la compra o el desarrollo de equipos de entrenamiento nuevos, por lo que deben modernizar gradualmente los más antiguos [2].

En las últimas décadas se ha producido un vertiginoso crecimiento de los simuladores de realidad virtual para entrenamiento de operarios, en parte, debido al incremento en la potencia de cálculo de las computadoras modernas y sus tarjetas gráficas. Para que el entrenamiento sea efectivo los operarios deben disponer de la funcionalidad equivalente y en lo posible sentirse en el ambiente real, para lo cual se debe recurrir a la simulación de los procesos físicos intervinientes. En el caso de los simuladores de equipos militares es deseable que las interfaces de comando se mantengan tal cual se encuentran en el momento de operación, por lo que se utilizan

los comandos reales que deben ser adaptados por medio de sistemas electrónicos al motor de simulación [3].

En un sentido amplio, "Sistema embebido" es la denominación aplicable a los equipos electrónicos que incluyen procesamiento de datos, pero que, a diferencia de una computadora de propósito general, están diseñados para satisfacer una función específica, como en el caso de un reloj digital, un router, el sistema de control de un automóvil (ECU), un reproductor de mp3, etc. Componentes fundamentales de un sistema embebido son la arquitectura de hardware subyacente, los dispositivos de entrada-salida y el software de control que en general esta optimizado para funcionar en sistemas de tiempos de respuesta acotados, bajo consumo y alta fiabilidad [4].

En este trabajo se presenta el desarrollo de una sistema embebido para la actualización y adaptación de un periscopio militar, y su vinculación al motor de simulación de realidad virtual, desarrollado para el entrenamiento de los alumnos de la Escuela de Submarinos y Buceo de la Armada en la ciudad de mar del plata [5], los requerimientos fueron analizados cuidadosamente, ya que se requería realizar tanto adaptaciones electrónicas como eléctricas y mecánicas, manteniendo el costo de actualización bajo y con suministro de repuestos accesibles en el país.

Por otro lado era prioritario no modificar la dinámica de uso ni la apariencia del equipo ya que el periscopio debe seguir luciendo y funcionando tal como los originales instalados en los submarinos.

2 Descripción de Sistema

La operación de un submarino involucra el manejo de numerosos subsistemas de control tales como el de profundidad, de timón y planos, de potencia de motores y de lastre, por otro lado están los subsistemas tácticos, que incluyen sonares, computadoras de tiro y los periscopios. Este último sistema es usado en tareas de vigilancia, detección de buques y ataque. En la ESyB se desarrolló un simulador de este sistema táctico, llamado ADITACSUB, que en su gran mayoría utiliza el mismo equipamiento que se encuentra en una nave real, (periscopio de ataque, sonar, computadora de cálculo de tiro, mesa de táctica, etc.), como se aprecia en la Figura 1.

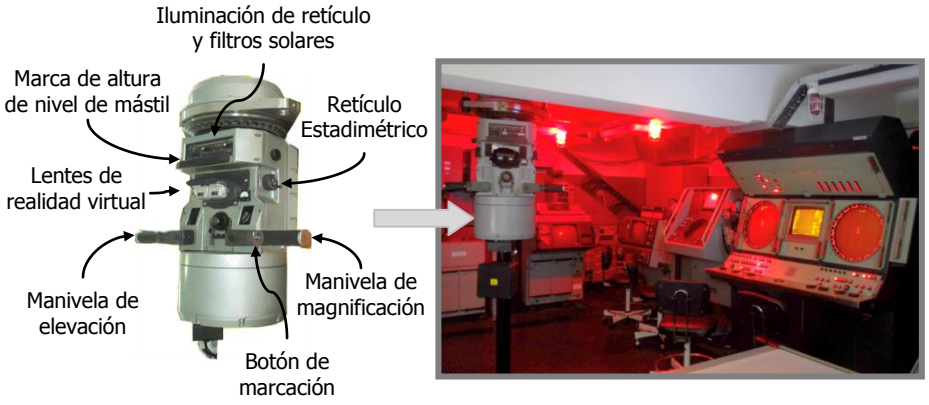


Figura 1: Descripción de las partes del periscopio y sala de simulación

Las señales que recibe cada uno de estos equipos son simuladas con computadoras, lográndose un grado de realismo aceptable. Estas computadoras, en la sala de generación de escenarios, permiten generar blancos tanto navales como aéreos de diferentes tipos y con parámetros de posición, orientación y velocidad determinados; y generar los parámetros dinámicos del propio submarino (posición, rumbo, profundidad, velocidad, etc). En uno de los procesos de creación de este simulador se adaptó un periscopio real de submarino (a la izquierda de la figura 1 y en el lado izquierdo de la sala de simulación) [5], en el que se reemplazó el visor y las ópticas por unos lentes de realidad virtual sobre el cual se proyecta un modelo de visualización tridimensional generado en tiempo real mediante un simulador gráfico de alto desempeño. De este modo, las operaciones y movimientos ejecutados sobre el periscopio son transformados en señales digitales e ingresados al simulador para realimentar el modelo visual. La Figura 2 muestra el esquema general del simulador y el modelo de comunicación.

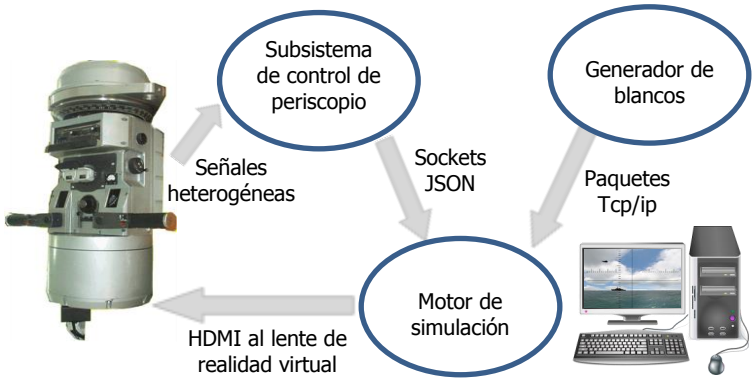


Figura 2: Modelo del simulador ADITAC

Este trabajo se centra en el "subsistema de control de periscopio" este módulo debe encargarse de adecuar la señales eléctricas como así también los movimientos mecánicos provenientes del periscopio y presentarlos de una manera clara ante el motor de simulación. Las operaciones más importantes que deben censarse son: Giro en azimut, selección de ópticas, control de elevación, filtros para disminuir intensidad de luz recibida, control de iluminación del retículo, desfasaje de retículo estadimétrico; y se describen a continuación en la tabla 1.

Tabla 1. Descripción de las operaciones del periscopio implementadas en el simulador ADITAC. Cada función requirió un análisis de rangos de uso y precisión mínima necesaria.

Operación	Descripción
Rotación en Azimut	Permite el giro del periscopio en 360º sobre su eje vertical sin límite de vueltas en ambos sentidos. La posición de 0º es tal que el periscopio mira hacia la proa del buque.
Manivela de elevación	Elevación de la óptica de visualización en el rango -10º a 60º de modo de realizar un paneo vertical del área de visión marítima, terrestre o aérea.

Cambios de aumento	Indicador de tres grados de aumento -½X, 6X y 12X.
Retículo estadimétrico	Desdobla la imagen con el objeto de estimar distancias a partir del uso de trigonometría, desplazando una imagen de un buque sobre si misma de manera que la parte inferior del buque coincida con el extremo superior del mástil mayor; y conociendo la altura de palo de dicho buque.
Panel de altura	Permite introducir al sistema la altura de mástil del objetivo. Se obtienen tres dígitos BCD desde un panel numérico mecánico.
Filtros solares	Cuatro grados de oscurecimiento del sistema óptico.
Rebatido de manivelas	Indicador de posición de las dos manivelas laterales. La posición hacia debajo de las dos manivelas indica a los sistemas que el periscopio está operativo.
Iluminación de retículo	Cinco grados de iluminación de la retícula de medición en pantalla.
Botón de marcación	Permite al operario marcar un blanco informando su posición al sistema de sonar.

3 Diseño e implementación de la solución

A fin de generar un modelo de solución funcional y correcto se utilizó la metodología de co-diseño en "V" para sistemas embebidos en la cual la verificación se desglosa en etapas de nivel de abstracción creciente. En cada etapa de diseño se crea un plan de pruebas que es el que guía la etapa de validación que le corresponde, este trabajo no intenta seguir la metodología como un formalismo sino más bien como una guía para el desarrollo ordenado.

3.1 Diseño de Software

El diseño del software tiene tres partes bien diferenciadas:

1. Diseño del software del colector de datos (Sistema embebido)
2. Diseño del software del controlador del periscopio (Driver)
3. Diseño del protocolo de envío y recepción de datos

La Figura 3 esquematiza la interacción entre el software del *Colector de Datos* y del *Controlador de Periscopio* como así también la conexión con el motor de simulación, en el diagrama se observa también los protocolos e interfaces de comunicación con el resto del entorno.

Diseño del software del controlador

Para el diseño del sistema de control de, se utilizó una arquitectura de software centrada en datos tipo blackboard [7], proponiendo dos threads principales, uno de ellos actualiza y transforma las señales de datos provenientes de las interfaces electrónicas del periscopio, el otro informa a las rutinas correspondientes sobre la actualización de un determinado conjunto de datos. Cada una de estas rutinas

empaqueta los datos y los encola para ser enviados por el canal serie al controlador de periscopio, para evitar inconsistencias en los datos se desarrollaron los mecanismos de sincronización correspondientes.

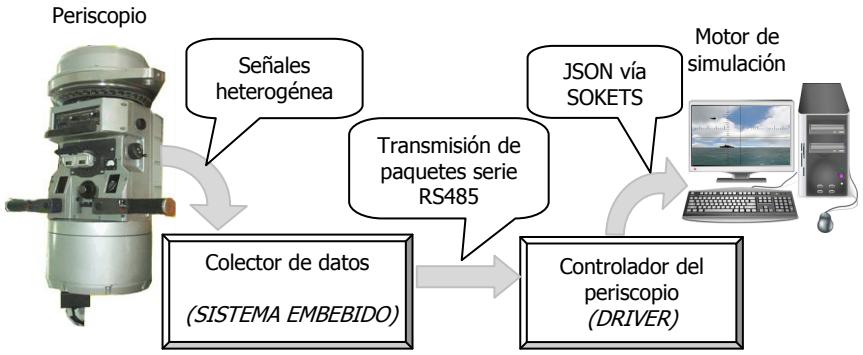


Figura 3: Esquema de interacción de módulos de software

Los datos son mantenidos en la memoria principal y actualizados en un período menor a 1ms, cabe destacar que todas las tareas de procesamiento de datos son determinísticas y sin bloqueos.

Los datos son considerados válidos para ser transmitidos si se cumplen las siguientes reglas:

1. El dato fue actualizado desde la última vez que fue transmitido o en caso de inicio del sistema si estaba como *no_inicializado*.
2. El dato no está siendo actualizado en ese momento.

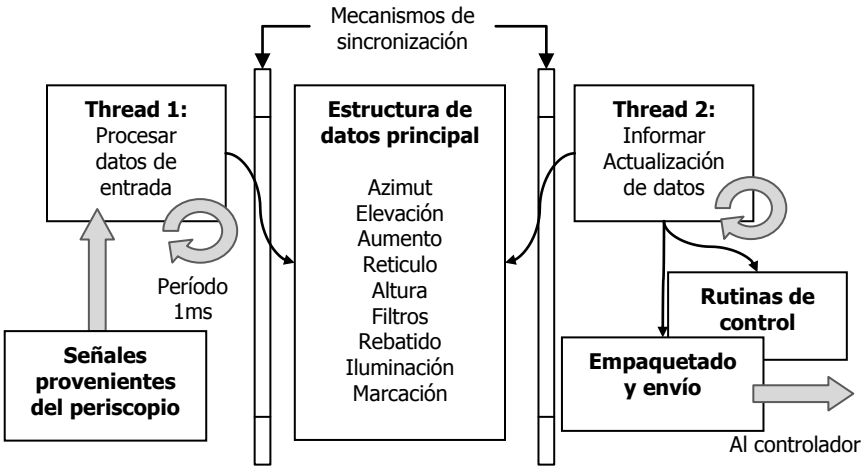


Figura 4: Esquema general de los módulos de software embebido.

Diseño del software de control del periscopio

El software de control (comúnmente denominado driver) es un pequeño sistema (ver **Figura 5**) que se inicia al encender el motor de simulación y permite realizar las siguientes funciones:

1. Chequeo del estado de salud del colector de datos y la comunicación
2. Configuración de los parámetros iniciales
3. Visualización de los datos en formato legible
4. Monitorización del sistema de recepción y despacho de eventos

El diseño es una típica arquitectura MVC (Model View Controller) que intercambia mensajes con el colector de datos vía una comunicación serial.

Para definir una interfaz clara con el equipo que desarrolló el motor de simulación se optó por una arquitectura *Cliente-Servidor* basada en *sockets.* El formato elegido para los datos fue JSON (JavaScript Object Notation), quedando de esta manera totalmente desacoplados ambos desarrollos. Esto impactaría positivamente también en el testeo e implantación del sistema.

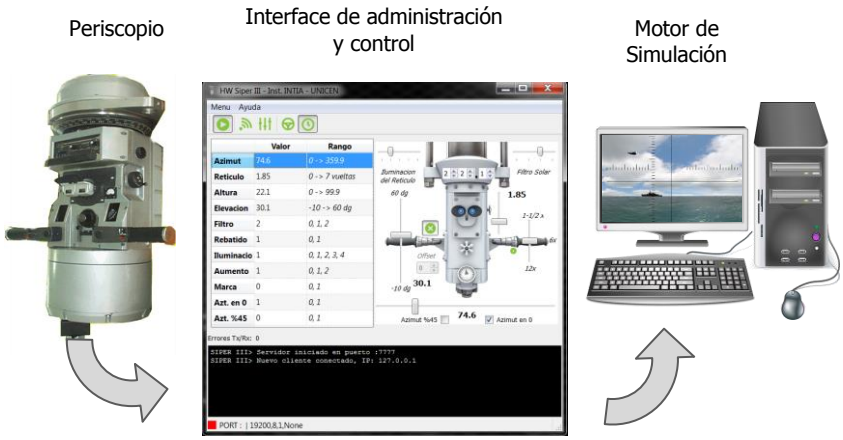


Figura 5: Interfaz gráfica del controlador de Periscopio SIPERIII

Diseño del protocolo de envío y recepción de datos

Luego de analizar varios métodos de transmisión de información, se optó por la implementación de uno propio sobre la capa física que brinda el standard RS485 [8]. El protocolo se basa en MODBUS RTU [9] al que se le han recortado varios campos para acelerar la tasa de transmisión, la definición del paquete de comunicación es compartida tanto por el colector de datos como por el controlador de periscopio.

Cuando ocurre un nuevo evento en los mandos del periscopio, el colector de datos lo decodifica e inmediatamente envía un paquete con dicha información al controlador de periscopio, este verifica su correctitud y procede a desempaquetarlo para encolarlo en la cola de eventos que se enviarán al motor de simulación.

Cabe destacar que cada paquete viene acompañado de un campo de chequeo de correctitud por códigos de redundancia cíclica (CRC). En la **Figura 6** se presenta el paquete de datos típico de comunicación entre el colector de datos y el controlador de periscopio.

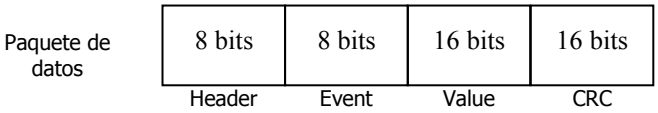


Figura 6: Estructura del paquete de datos básico.

3.2 DISEÑO DE HARDWARE

El diseño del hardware del sistema se dividió en dos etapas. Por un lado, se realizó el análisis de requerimientos y las adaptaciones necesarias a fin de que interactúen con los sensores y actuadores seleccionados. En esta primera etapa, las adaptaciones realizadas para las operaciones fueron las siguientes:

Rotación en azimut: El sistema de giro en 360º del periscopio posee una caja de engranajes en la parte superior que interactúan con varios sincromotores. En esa caja se realizó la Inserción mecánica de los sensores. Se utilizó un encoder relativo de 360º con una sensibilidad de 60 pulsos por grado (error de un minuto de grado) y dos sensores de paso por 0º y por múltiplos de 45º. Estos últimos sensores permiten detectar posiciones absolutas del periscopio.

Manivela de elevación: Para esta función se realizó el reemplazo del sincromotor ubicado dentro de la manivela izquierda por un potenciómetro con mapeo directo de la posición angular de 70 grados a un valor binario de 10 bits vía un conversor analógico-digital. Se utilizó un potenciómetro lineal de altas prestaciones para disminuir el posible ruido eléctrico en la conversión. Los requerimientos originales planteaban errores máximos de medio grado. Mantener bajo el ruido de digitalización en este componente es importante cuando se usa el aumento de 6x ó 12x

Cambios de aumento: En este caso se reutilizó el componente original consistente en un selector rotativo de tres posiciones con tres salidas digitales de 0 ó 5 voltios asociadas a cada uno de los tres aumentos.

Retículo estadimétrico: Esta función opera mediante una rueda lateral con capacidad de giro de siete vueltas. En este caso se realizó la unión mecánica entre la cremallera interna asociada a la rueda de control mediante una unión cardánica a un potenciómetro multivueltas de altas prestaciones y su posterior digitalización mediante un ADC de 10 bits. La exactitud debida a errores inducidos por ruido eléctrico es de 8 bits.

Panel de altura*:* Se reutilizo el panel original consistente de tres selectores mecánicos de dígitos decimales. Puesto que la salida de cada selector es un valor BCD de 4 bits, se envían los tres valores directamente a la placa de lectura de datos.

Filtros solares: Se adaptó en este caso un selector rotativo de cuatro posiciones que devuelve los valores codificados en binario en dos bits.

***Rebatido de manivelas**:* El rebatido de las manivelas laterales del periscopio aporta información al resto de los sistemas tácticos indicándoles que deben recibir datos de posición y tipo de los blancos detectados. La adaptación en este caso consistió en la detección de la posición individual de cada manivela en sus dos posiciones extremas (rebatida o extendida). Para la detección se utilizaron sendos conmutadores eléctricos de presión conectados en serie.

Iluminación de retículo: Para esta funcionalidad se adaptó un selector rotativo de cinco posiciones al cuerpo del periscopio que devuelve las intensidades seleccionadas codificadas en binario en tres bits.

***Botón de marcación**:* Se utilizó el componente original consistente de un pulsador que devuelve 5 voltios en estado de reposo y 0 voltios al ser oprimido (lógica negativa).

***Encendido/apagado de los lentes de visualización y sistema de ventilación**:* Esta funcionalidad no existe en el periscopio real puesto que tiene un sistema de visualización óptico. Se implementa en el simulador a fin de encender los distintos subsistemas cuando se inicia una sesión de entrenamiento.

Una segunda etapa del diseño de hardware consistió en la selección del controlador apropiado para el manejo de las señales descritas anteriormente y su procesamiento y empaquetado para transmitirlas al driver en la PC. Se optó por un sistema de desarrollo basado en un microcontrolador con arquitectura AVR [10] de la firma ATMEL más una placa de diseño propio para soporte de potencia y comunicación serial vía protocolo rs485 que aumenta la inmunidad al ruido eléctrico en un ambiente donde existen tensiones de alimentación alternas.

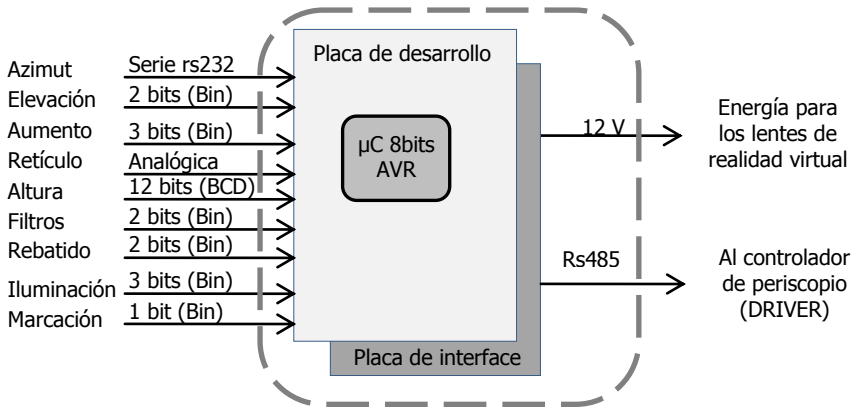


Figura 7: Esquemático del hardware del colector de datos

Los requerimientos originales de implementación del sistema con determinadas restricciones presupuestarias llevaron al uso de un encoder relativo para la adquisición del ángulo de azimut. Debido a este tipo de encoder, no puede conocerse la posición absoluta del periscopio. Una solución parcial es la incorporación de uno o más sensores de posición para detectar ángulos determinados (sensor de 0º y sensor de

múltiplos de 45º). Esta solución, aunque efectiva, es falible si se usa el periscopio durante cierto tiempo sin pasar por algún sensor absoluto. Para solucionar el problema se implementó un filtro de corrección que disminuye gradualmente el error a fin de evitar efectos de saltos en la visualización.

4 Experimentos

El subsistema de control de periscopio debe operar en un ambiente con dos desventajas importantes. Por un lado, la computadora donde se encuentra el controlador de periscopio y el sistema gráfico (en la sala de control de ejercicio) está ubicada a aproximadamente diez metros del colector de datos (dentro del periscopio en la sala de adiestramiento) y, los demás sistemas en esa sala funcionan con tensiones alternas altas. En este ambiente es necesario asegurar que el ruido eléctrico no perjudique la calidad de la comunicación. A fin de probar el desempeño de la etapa de comunicaciones, y dado que el periscopio real se encuentra ya instalado en la base naval, se desarrolló un emulador en hardware del mismo para la realización de pruebas de laboratorio sin necesidad de conectarse físicamente al sistema.

Los experimentos realizados permitieron determinar la cantidad de errores de transmisión de paquetes con distancias superiores a diez metros. En estas pruebas se determinó que los errores fueron mayormente de sincronización durante el encendido o apagado del periscopio y desaparecían cuando las máquinas de estado de transmisión y recepción se sincronizaban.

En pruebas de stress se sometió al sistema a sesiones de funcionamiento de hasta una semana completa para comprobar la eficiencia de las comunicaciones y los componentes de hardware con resultados satisfactorios.

En los ensayos de campo, en el ADITAC, se detectaron errores relacionados con ruidos de conversión en los ADC del colector de datos debidos a las perturbaciones eléctricas de los sistemas. Esto obligó al uso de un sistema de comunicación optoacoplado entre el colector de datos y el controlador del periscopio.

5 Conclusiones y trabajos futuros

Se consiguió implementar un sistema embebido adecuado al presupuesto, con componentes accesibles a nivel nacional y con tiempos de respuesta por debajo de lo exigido por el motor de simulación.

La adecuada división entre las etapas de hardware y software, sumado a un modelo de comunicación integral permitió un proceso de desarrollo casi en paralelo de ambas etapas y, al mismo tiempo, el protocolo de comunicación desarrollado puede ser ampliado para su aplicación en otras aplicaciones.

Terminada la experiencia de implantación de este sistema de entrenamiento táctico se comenzó con las primeras etapas para el desarrollo de un nuevo simulador de operaciones de manejo integral de submarinos para entrenamiento en tareas de gobierno y control de inmersión.

Este emprendimiento busca afianzar lazos de trabajo interdisciplinario e interinstitucional; y permitirá aplicar metodologías de diseño más robustas para afrontar proyectos de mayores dimensiones.

Agradecimientos

Este trabajo se desarrolló dentro del marco del proyecto de investigación "Metodologías de Diseño para Sistemas Embebidos" 03-C238 del programa de incentivos a la investigación y el PICT-2009-0041: Desarrollo y Verificación de Sistemas Digitales Complejos. Además, fue financiado en parte por el proyecto de transferencia tecnológica (RCS 1112/13 UNCPBA-ESyB).

Referencias

1. Loffler, T., Nielson, J.: International HARM precision navigation upgrade. A GPS/INS missile upgrade that improves effectiveness and minimizes friendly-fire accidents. IEEE Aerospace and Electronic Systems Magazine, 18(5):26-31, (2003).
2. Aguierre, L., Ramirez, D., Leiva, L., Marone, J., Vazquez, M.: AHRS R-001: Actualización de Sistemas Inerciales de Navegación en Aeronaves Supersónicas. II Congreso de Microelectrónica Aplicada. (2011).
3. BORONI G., VÉNERE, M. "Un simulador distribuido para entrenamiento de operarios". Proceedings VIII Congreso Argentino de Ciencias de la Computación. 2002. ISBN Nº: 987-96- 288-6-1. pp. 727-738
4. Tosini, M., Todorovich, E., Vázquez, M., Leiva, L., Aciti, C., Marone, J., Goñi, O., Pantaleone, L., Acosta, N., Curti, H., Toloza, J., Kornuta, C: Metodologías de Diseño para Sistemas Embebidos. XV Workshop de Investigadores en Ciencias de la Computación. (2013)
5. Boroni, G., Vagliati, P., Venere, M., Marone, J., Tosini, M., Avila, E., Grasso, O., Lagar, D., Leisamon, R.: Realidad Virtual y Simulación Computacional Aplicada al Entrenamiento de Operarios. Mecánica Computacional, Vol XXII, Bahia Blanca. Noviembre 2003
6. Pérez, A., Berreteaga, O., Ruiz de Olano, A., Urkidi, A., Perez, J.: Una metodología para el desarrollo de hardware y software embebidos en sistemas críticos de seguridad. Sistemas, Cibernética e informática. Vol 3, Nro 2, ISSN: 1690-8627. (2006).
7. Daniel D. Corkill. 1988. Design Alternatives for Parallel and Distributed Blackboard Systems. Technical Report. University of Massachusetts, Amherst, MA, USA.
8. Soltero, M., Zhang, J., Cockril, C., Zhang, K., Kinnaird, C., & Kugelstadt, T. RS-422 and RS-485 Standards Overview and System Configurations, Application Report, (2002).
9. MODBUS.ORG, "MODBUS Application Protocol Specification V1.1," www.modbus.org, Access Date, Jan, 2003.
10. Myklebust, G: The AVR Microcontroller and C Compiler Co-Design. ATMEL Development Center, Trondheim, Norway, (1996).

Evaluación de la calidad de la experiencia para servicios de datos móviles

Brenda R. Quiñónez, Luis F. Fernández, Diego P. Pinto Roa, Carlos Núñez y María E. García

Facultad Politécnica, Universidad Nacional de Asunción, Paraguay
{bquinonez, luis.fernanez, dpinto, cnunez, mgarcia}@pol.una.py

Abstract. La calidad de la experiencia (QoE) percibida por los usuarios móviles juega un papel importante en el posicionamiento de las empresas de telecomunicaciones y en la fidelidad de los usuarios. Además, puede ser considerada como una métrica clave en la gestión de recursos de red. Por esta razón, este trabajo se enfoca en el establecimiento de un modelo estimación del QoE considerando parámetros a nivel de red, dispositivo y aplicación para el contexto de uso de video streaming. Para este fin, se implementan una plataforma de transmisión de datos móviles y una aplicación Android que son utilizadas durante el proceso de recolección de datos. Los datos obtenidos en dicho proceso se emplean como entrada de distintos algoritmos de aprendizaje automático, cuyos desempeños son comparados con el fin de identificar a los más adecuados para la creación de un modelo que permita estimar el QoE en el contexto de uso mencionado.

Keywords: QoS, QoE, Datos móviles, Redes móviles, Video streaming, Aprendizaje Automático.

1 Introducción

La gran proliferación de dispositivos móviles, la facilidad de acceso a Internet proporcionada a los usuarios y los avances tecnológicos en la infraestructura de las redes móviles han generado una gran demanda por los servicios de datos móviles. Como resultado de ello, actualmente se experimenta un importante aumento del tráfico de datos a través de las redes móviles, lo que puede ocasionar la saturación de las infraestructuras de las redes móviles y en consecuencia, la degradación de la calidad percibida por los usuarios [2].

Conocer cómo los usuarios de los servicios móviles perciben la calidad de los mismos permitirá a las empresas de telecomunicaciones mejorar de manera eficaz y rentable los servicios brindados y de esta forma mantener la fidelidad de los clientes y la competitividad en el mercado.

En base a la importancia de conocer la percepción del usuario acerca del servicio prestado, este trabajo propone el diseño de un modelo de estimación de la calidad de la experiencia para servicios de datos móviles, y en particular de video streaming, considerando parámetros a nivel de red, dispositivo y aplicación.

Lo que resta de este trabajo se organiza de la siguiente manera: en la sección 2 se describe brevemente el concepto de la Calidad de Servicio (QoS) y Calidad de la Experiencia (QoE) y se presenta una clasificación de los métodos de evaluación del QoE. En la sección 3 se realiza una revisión literaria de los métodos utilizados en la evaluación y medición de QoE. En la sección 4se presentan los objetivos de este trabajo, el método propuesto y los experimentos llevados a cabo. Finalmente, en la sección 5, presentamos las conclusiones y los trabajos futuros.

2 Calidad de la experiencia y calidad de servicio

La Calidad de la Experiencia (Quality of Experience - QoE) es el término utilizado para describir las percepciones del usuario acerca del rendimiento de un servicio en particular [9]. Está expresada en sensaciones humanas tales como: 'bueno', 'excelente', 'pobre', entre otros.

Por otra parte, la Calidad de Servicio (Quality of Service - QoS) es definida formalmente como la habilidad de una red o porción de una red de proporcionar un servicio a un nivel garantizado [10]. Puede ser medida, expresada y comprendida en términos de red y elementos de red, que por lo general tienen poco significado para el usuario.

Con el fin de proporcionar el mejor QoE de una forma rentable, competitiva y eficiente, los proveedores de servicios deben manejar el QoS de la red y la provisión de servicios de forma eficiente y efectiva [9].

2.1 Clasificación de los métodos de evaluación de QoE

De acuerdo a [10], los métodos de evaluación de QoE se pueden clasificar en: pruebas subjetivas y métodos objetivos.

Pruebas Subjetivas: la idea es que las personas, que utilizan el servicio en un entorno real, evalúen el servicio completando encuestas. Las preguntas y respuestas de los cuestionarios deben reflejar diversas características del servicio. Es considerado el método m\'as fiable para la evaluación del QoE. Sin embargo, es costoso. La medida más utilizada en las pruebas subjetivas se denomina MOS (Mean Opinion Score), esta medida permite obtener la calidad percibida por el usuario en base a una escala de valores predefinidos [1].

Métodos objetivos: proporcionan una evaluación del QoE basada en la medición de varios parámetros relacionados con la prestación de servicios particulares o indicadores de calidad de servicio en la señal de salida del canal de transmisión, es decir, la señal que será percibida por el usuario real. La desventaja de los métodos objetivos es que pueden ser poco informativos y dar resultados inexactos. Por lo tanto, son desarrollados cuidadosamente y verificados por pruebas subjetivas.

Pseudo-Subjective Quality Assessment - PSQA: esta metodología propone un método hibrido entre las evaluaciones subjetivas y objetivas de la calidad de la experiencia [8]. La metodología PSQA nos permite obtener un valor acerca del QoE cercano

al valor promedio que nos daría un observador humano. PSQA está basado en el aprendizaje de cómo los observadores humanos cuantifican la calidad de un flujo de datos (audio, video, entre otros) bajo condiciones experimentales estandarizadas. El proceso de aprendizaje consiste en el entrenamiento de un algoritmo de aprendizaje automático con el fin de capturar la relación entre un conjunto de factores que tienen un fuerte impacto en la calidad percibida.

3 Trabajos relacionados

En esta sección haremos una breve revisión de la literatura existente con respecto a la evaluación del QoE en servicios móviles.

En [5], los autores se enfocan en la medición del QoE para video streaming sobre redes inalámbricas. Los autores hacen uso de enfoques de evaluación subjetivo, objetivo e híbrido y miden sus rendimientos. Específicamente, se centran en la metodología PSQA, y demuestran que por medio de dicho enfoque se obtienen resultados similares a los obtenidos en las pruebas subjetivas realizadas por los observadores humanos.

En [11], los autores presentan una aplicación móvil cuyo objetivo es obtener las métricas de QoS y evaluar el QoE en un entorno móvil. La aplicación móvil, desarrollada para la plataforma Android, actúa como un agente móvil que realiza las mediciones de las métricas específicas de un servicio. Para clasificar el QoE del usuario, los autores han considerado una escala MOS de cinco puntos [1]. Así, por medio de un modelo de cálculo del MOS y los parámetros recabados por el agente móvil, se obtiene un valor de MOS, el cual representa una estimación del QoE de un usuario.

Un modelo de correlación QoS-QoE para servicios de Internet móvil que utiliza el enfoque de Redes Neuronales es presentado en [7]. En la metodología propuesta, se hace uso de pruebas subjetivas para tres tipos de servicios de Internet m\'ovil (navegación web, video streaming, descarga de archivos), además del uso de agentes m\'oviles como medio para obtener parámetros de red. A partir de los datos recabados, la Red Neuronal (*Perceptrón Multicapa*) es entrenada con el fin obtener una correlación entre los parámetros de QoS y QoE.

En vista del estado del arte relacionado a la evaluación de QoE expuesto en este apartado, hemos identificado las siguientes limitaciones:

- Los reportes presentados no consideran todos los factores que inciden en el QoE. Por lo tanto, es necesario, diseñar un sistema de monitoreo que, además de monitorear los parámetros de QoS a nivel de red, realice el monitoreo a nivel de aplicación y también considere el monitoreo de los parámetros de estado del dispositivo móvil (memoria disponible, uso de CPU, entre otros).
- Comparar distintas técnicas de aprendizaje automático para el proceso de estimación del QoE y seleccionar la que arroja resultados más cercanos a lo que sería la percepción del usuario.

4 Método propuesto

En base a la problemática del monitoreo y evaluación de la calidad percibida por el usuario final, este trabajo tiene como objetivo general y aporte principal: el diseño de un modelo de estimación de la calidad de la experiencia para servicios de datos m\'oviles, en particular en el streaming de video, considerando parámetros a nivel de red, dispositivo y aplicación.

4.1 Objetivos específicos

1. Desarrollo de una plataforma de transmisión de datos móviles que emule una red de transmisión de datos móviles real.
2. Desarrollo de una aplicación m\'ovil (para la plataforma Android) que permita la recolección de parámetros a nivel de red, dispositivo y aplicación y la realización de pruebas subjetivas para la evaluación del QoE en el contexto de video streaming.
3. Recolección de datos que servirán de base para el objetivo de esta propuesta utilizando la plataforma y la aplicación m\'ovil desarrollada.
4. Comparación e identificación de los algoritmos de aprendizaje automático m\'as adecuados para la estimación del QoE en base a los parámetros obtenidos con la aplicación móvil.
5. Creación de un modelo de estimación del QoE para el contexto de video streaming, considerando parámetros a nivel de red, dispositivo y aplicación.

4.2 Metodología de solución

Este trabajo utiliza una adaptación de la metodología PSQA con el fin de generar un modelo de correlación entre la calidad experimentada por el usuario y los parámetros de calidad de servicio a nivel de red, dispositivo y aplicación. En la figura 1 se puede observar el *esquema propuesto en este trabajo*, que es llevado a cabo para la obtención del modelo de correlación mencionado. A continuación se describen brevemente cada uno de los componentes del esquema.

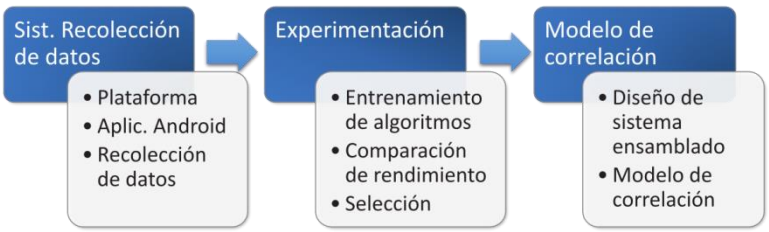


Fig. 1. Vista del proceso de obtención del modelo de correlación.

4.3 Sistema de recolección de datos

1. **Plataforma de transmisión de datos móviles:** el proceso de recolección de datos consiste en obtener la información necesaria acerca de cómo los usuarios evalúan la calidad de distintos servicios móviles ofrecidos bajo diversas condiciones de red y utilizando diferentes dispositivos móviles. Para este fin se optó por construir un entorno de laboratorio conformado por una plataforma de transmisión de datos móviles que nos permite emular una red de transmisión de datos móviles real. En la figura 2 se puede observar la composición general de la plataforma de transmisión.

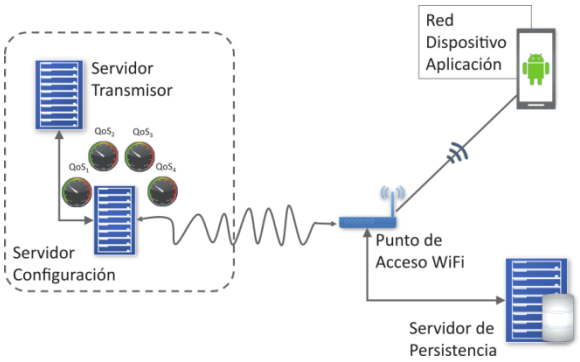


Fig. 2. Plataforma de transmisión de datos móviles.

La plataforma de transmisión cuenta con cinco componentes principales, a continuación se describirá brevemente las funciones de los mismos dentro del entorno:

- *Servidor Transmisor*: este servidor es el encargado de almacenar el contenido del servicio móvil (videos) y luego transmitirlo a los diversos dispositivos conectados a la plataforma.
- *Servidor de Configuración*: es el encargado de realizar las variaciones de la condiciones de red, de forma a generar tráfico imperfecto, emulando los distintos escenarios de tráfico que reciben los usuarios. Para lograr lo anterior, el servidor modifica cuatro parámetros básicos de calidad del servicio: *ancho de banda, delay, jitter y paquetes perdidos.*
- *Punto de Acceso WiFi:* es el punto de acceso de los dispositivos móviles a la red conformada por el servidor de configuración y de transmisión.
- *Dispositivos Móviles:* son los encargados de conectarse al punto de acceso WiFi y ejecutar una aplicación Android de encuestas (pruebas subjetivas).
- *Servidor de Persistencia:* es el encargado de servir como repositorio de información de los datos enviados por los dispositivos móviles.

2. **Aplicación móvil Android:** para la realización de las pruebas subjetivas se utilizó la plataforma de transmisión de datos móviles descripta anteriormente y se desarrolló una aplicación Android cuyo propósito es realizar una encuesta a los usuarios

acerca de la calidad experimentada luego de la visualización del video, recolectar métricas a nivel de aplicación, recolectar los parámetros de red y obtener las características y el estado del dispositivo móvil que la está ejecutando.

3. **Recolección de datos:** el proceso de recolección de datos consiste en la realización de pruebas subjetivas con usuarios reales utilizando la plataforma de transmisión de datos móviles expuesta en la figura 2 y la aplicación Android desarrollada. Cada prueba subjetiva se realiza bajo condiciones de red distintas. Para generar las distintas condiciones de red utilizadas en este experimento escogimos diversos valores para los parámetros de red seleccionados que pueden ser observados en el cuadro \ref{tab:param}.

Table 1. Parámetros de red emulados

Parámetros	Valores	Unidades
Ancho de Banda	350, 750, 1200, 1600	kbps.
Paquetes perdidos	0, 2.5, 5, 10	%
Delay	100, 300, 5000	ms.
Jitter	0, 25, 50	ms.

Como parte del proceso de pruebas subjetivas, fueron encuestadas un total de 70 personas, cada una de ellas realizó 5 pruebas con distintas configuraciones de red. En cuanto a los dispositivos móviles, se contabilizó un total de 34 dispositivos diferentes catalogados en tres gamas (baja, media o alta) de acuerdo a las prestaciones del procesador y el procesador gráfico, la memoria RAM y las propiedades de la pantalla (resolución y densidad de píxeles por pulgada). Posterior al proceso de recolección se realizó un proceso de análisis y limpieza de datos (*data cleaning*) sobre el total de las 350 muestras obtenidas. Como resultado del procesamiento de datos anterior, se obtuvieron 295 muestras de pruebas subjetivas.

4.4 Experimentación

El proceso de experimentación consiste en el entrenamiento, comparación y selección de algoritmos de aprendizaje automático, que para el problema de la estimación del QoE, obtienen los mejores resultados de acuerdo a las métricas de evaluación, descritas posteriormente. Así, este proceso consta de lo siguiente:

1. **Entrenamiento de algoritmos de aprendizaje automático:** a continuación se describirá el proceso de entrenamiento de los distintos algoritmos de aprendizaje ofrecidos por la herramienta de aprendizaje automático WEKA [3].

   Definición del conjunto de datos: el conjunto de datos de entrada al proceso de entrenamiento constituye el conjunto de ejemplos utilizados para el entrenamiento y evaluación de los algoritmos de aprendizaje. En el presente caso, cada una de las pruebas subjetivas corresponde a un ejemplo o instancia del conjunto, mientras que cada uno de los datos recolectados por la misma corresponde a un atributo de la

instancia. Se ha decidido analizar los atributos de las instancias en tres categorías diferentes: correspondientes al dispositivo, al estado de la red y aquellos correspondientes a métricas a nivel de aplicación. En la tabla 2 se detallan los atributos que forman parte de las instancias del conjunto de datos, separados en las categorías correspondientes.

Table 2. Atributos de las instancias del conjunto de datos.

Categoría	Atributo	Descripción
Dispositivo	Uso de CPU	Porcentaje de uso de la CPU del dispositivo.
	Uso de RAM	Porcentaje de uso de la memoria RAM del dispositivo.
	RAM	Cantidad total de memoria disponible.
	Nivel de batería	Porcentaje de batería disponible.
	Gama	Categorización del dispositivo de acuerdo a sus prestaciones
Red	Ancho de Banda	Ancho de banda efectivo.
	Paquetes Perdidos	Porcentaje de pérdida de paquetes de la red subyacente.
	Delay (RTT)	Tiempo de viaje de ida y vuelta de un paquete.
	Jitter	Variación del delay.
Aplicación	Tiempo de carga inicial	Cantidad de tiempo transcurrido durante el proceso de carga inicial del video.
	Tiempo de buffering	Duración promedio de los eventos de rebuffering.
	Frecuencia de buffering	Cantidad de ocurrencias de eventos de rebuffering.
	Cancelado	Atributo binario que indica la cancelación o no de la reproducción del video.

Algoritmos de aprendizaje automático: luego de definición de los datos, se realiza el proceso de entrenamiento y comparación de distintos algoritmos de aprendizaje automático con el objetivo de identificar aquellos que presentan mejor desempeño para el problema de la estimación del QoE. Los datos citados anteriormente se utilizaron como entrada de 18 algoritmos de aprendizaje automático. Se realizó un total de 10 ejecuciones de 10 *cross validation* para cada uno de los algoritmos configurados con los parámetros por defecto de WEKA 3.7.

2. **Comparación de algoritmos de aprendizaje:** para realizar un análisis comparativo de los distintos algoritmos de aprendizaje automático sobre el conjunto de video streaming utilizamos las métricas *porcentaje de aciertos*, *coeficiente kappa* y la *medida F* [4]. En la figura 3, se puede observar el diagrama de caja correspondiente a la métrica de porcentaje de aciertos.

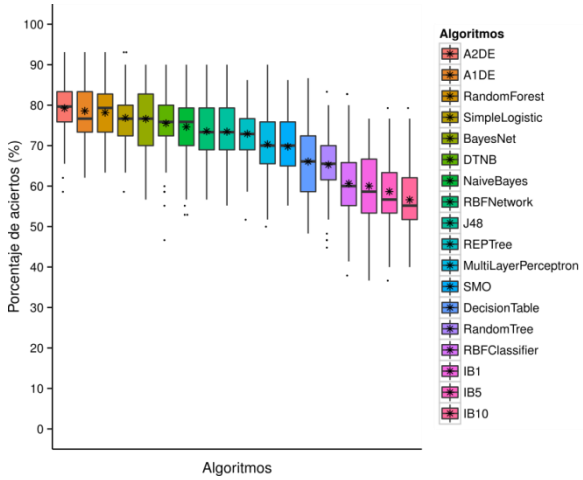


Fig. 3. Diagrama de cajas con respecto al porcentaje de aciertos.

Para las métricas kappa y medida F se obtuvieron resultados similares, en donde los algoritmos A2DE, A1DE, RandomForest y SimpleLogistic presentaron los mejores resultados con valores promedio iguales a: (0.74 y 0.79), (0.73 y 0.78), (0.72 y 0.77) y (0.7 y 0.76), respectivamente.

Como resultado de los experimentos podemos concluir que para el conjunto de datos analizado los clasificadores A2DE y A1DE obtienen, en promedio, los mejores resultados para la mayoría de las métricas. Los algoritmos tales como: IBk, RandomTree, SMO y RFBClassifier obtuvieron valores inferiores en la mayoría de las métricas analizadas.

3. **Selección de algoritmos de aprendizaje:** de esta manera, obtuvimos el ranking de los algoritmos que se desempeñaron mejor para el conjunto de datos de video streaming: *A2DE, A1DE, RandomForest y SimpleLogistic*, en ese orden.

4.5 Modelo de Correlación

1. **Diseño del sistema ensamblado:** para culminar el análisis iniciado en la sección anterior, se ha creado un sistema ensamblado [6] formado por los algoritmos con mejores desempeños: A1DE, A2DE, SimpleLogistic y RandomForest. La operación de combinación de los resultados de los algoritmos base del sistema ensamblado formado es sencilla: se toma como clase aquella votada por la mayoría de los

algoritmos base. En las figuras 4(a), 4 (b) y 4(c) se pueden observar los resultados obtenidos por el sistema ensamblado para el conjunto de datos analizado.

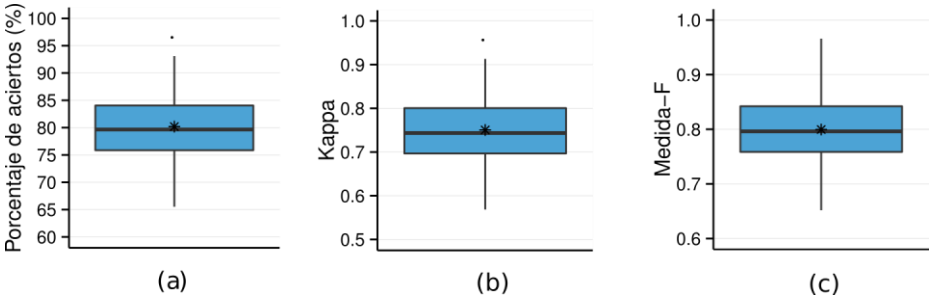


Fig. 4. Diagramas de cajas correspondientes a las métricas analizadas para el algoritmo ensamblado.

Como conclusión, observando los diagramas de cajas de las figuras 4(a), 4 (b) y 4(c), en promedio, el sistema ensamblado presenta mejores resultados en todas las métricas y una menor variabilidad en los resultados que los algoritmos base, notándose que la mayor parte de los resultados están centrados en la media.

2. Modelo de estimación del QoE: WEKA permite guardar los modelos obtenidos por los algoritmos entrenados como objetos Java serializados en un archivo binario. A partir de ello, se creó un archivo binario que representa al modelo (sistema ensamblado), el cual podrá ser nuevamente utilizado por medio de la librería Java del software WEKA. De esta manera, se obtiene como resultado una herramienta que permite estimar la calidad percibida por los usuarios (valor MOS) para el contexto de video streaming en base a los parámetros analizados.

5 Conclusión

Este trabajo presenta una herramienta que permite estimar el QoE para el contexto de video streming en base al monitoreo de un conjunto de parámetros a nivel de red, dispositivo y aplicación. Esta herramienta puede ser de gran utilidad para las operadoras de datos móviles, debido a que otorga visibilidad acerca de las condiciones que afectan a la calidad de la experiencia. Con esta información, las operadoras pueden tomar acciones correctivas que permitan maximizar la calidad de la experiencia mediante la optimización del uso de recursos de la infraestructura de la red.

Como líneas de trabajo futuras, se propone realizar la validación del modelo de correlación obtenido con datos de un entorno operativo real. Además, este trabajo podría ser extendido incluyendo otros contextos de uso sobre transmisiones de datos móviles (live streaming, videoconferencias, entre otros). Igualmente se podrían incluir otros

factores que influyen en el QoE como parámetros de entrada del modelo de correlación, tales como: políticas de precios, el contexto, el perfil del usuario, entre otros.

Agradecimientos: este trabajo fue apoyado por el Consejo Nacional de Ciencia y Tecnología (CONACyT). Agradecemos a los voluntarios de las pruebas.

Referencias

1. ITU-T P.800. Methods for subjective determination of transmission quality. Tech. rep. (Aug 1996), https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-P.800-199608-I!!PDF-E&type=items
2. Mobile crowd sensing and data offloading in collaborative networks (2013), http://www-public.it-sudparis.eu/~maag/MOSAIC/index.html
3. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. ACM SIGKDD explorations newsletter 11(1), 10-18 (2009)
4. Japkowicz, N., Shah, M.: Evaluating learning algorithms: a classification perspective. Cambridge University Press (2011)
5. Piamrat, K., Viho, C., Bonnin, J., Ksentini, A.: Quality of experience measurements for video streaming over wireless networks. In: Information Technology: New Generations, 2009. ITNG'09. Sixth International Conference on. pp. 1184-1189. IEEE (2009)
6. Polikar, R.: Ensemble based systems in decision making. Circuits and Systems Magazine, IEEE 6(3), 21-45 (2006)
7. Rivera, S., Riveros, H., Ariza-Porras, C., Lozano-Garzon, C., Donoso, Y.: QoS-QoE correlation neural network modeling for mobile internet services. In: Computing, Management and Telecommunications (ComManTel), 2013 International Conference on. pp. 75-80. IEEE (2013)
8. Rubino, Gerardo: Quantifying the quality of audio and video transmissions over the internet: the psqa approach. Design and operations of communication networks: a review of wired and wireless modeling and management challenges. Imperial College Press, London (2005)
9. Soldani, D., Li, M., Cuny, R.: QoS and QoE management in UMTS cellular systems. John Wiley & Sons (2007)
10. Stankiewicz, R., Cholda, P., Jajszczyk, A.: Qox: what is it really? Communications Magazine, IEEE 49(4), 148-158 (2011)
11. Tsiaras, C., Sehgal, A., Seeber, S.: Towards evaluating type of service related quality-of-experience on mobile networks (2014)

Diseño e Implementación de Red Académica Experimental aplicando IPv6

Gustavo Mercado[*], Javier Gitto[*], Rubén Soria[§], Carlos Taffernaberry[*], Cristian Pérez Monte[*], Marcela Orbiscay[§], Sebastián Tobar[*], Ana Diedrichs[*],

[*]GridTICS – Grupo UTN de I&D en Tecnologías de la Información y las Comunicaciones / Departamento de Electrónica / Facultad Reg. Mendoza / UTN Rodríguez 273, Capital – Mendoza,
{gustavo.mercado}@gridtics.frm.utn.edu.ar

[§]Centro Científico Tecnológico de Mendoza CONICET,
Av. Dr Ruiz Leal S/N, Capital - Mendoza
{rsoria, morbis}@mendoza-conicet.gob.ar

Resumen: Las redes académicas científicas regionales y metropolitanas están poco desarrolladas en Latinoamérica. Generalmente cada institución académica o científica tiene su propio acceso a la red institucional de su organización o a un proveedor de Internet privado y toda comunicación entre distintas instituciones se desarrolla a través de este esquema. Este trabajo contiene los lineamientos generales del diseño y despliegue de una red metropolitana experimental de avanzada, (Red Académica Científica y Tecnológica), que permite la conexión en forma confiable, con altas prestaciones y bajo costo a instituciones académicas y científicas de la ciudad de Mendoza - Argentina. Esta red se basa en una infraestructura de telecomunicaciones heterogénea, con enlaces inalámbricos y fibra óptica sobre la que se montan nuevas tecnologías tales como movilidad IPv6, multicast, gamestreaming e IOT; además de explorar otras tecnologías y aplicaciones emergentes. La experiencia obtenida sirve en la capacitación, la difusión y la diseminación del diseño e implementación de redes interinstitucionales fomentando los proyectos científicos y tecnológicos en ambientes colaborativos regionales basados en la infraestructura de red de telecomunicaciones propuesta.

Palabras Clave: Redes de Avanzada, Entornos colaborativos, IPv6, QoS, VoIP, Movilidad IP, Virtualización, IOT

1 Introducción

La revolución del siglo XX produjo la aparición de Internet, la cual introdujo cambios en todos los estados de nuestras vidas. Sin embargo, a poco tiempo de su incorporación en nuestro quehacer, se fue sintiendo un vacío e insatisfacción en la comunidad científica y educativa, en la academia y en la investigación [1]. La solución casi no tardó en aparecer, y fueron los avances en infraestructura tecnológica, desarrollados por las redes avanzadas, lo que reinstaló la esperanza en las comunidades de investigación, científicas y académicas, pues pusieron a su disposición a través de una gran infraestructura tecnológica condiciones de uso casi

en exclusividad de herramientas y aplicaciones que les permitieron el desarrollo e incremento de sus actividades, lo que era insostenible a partir de la Internet comercial.

Hoy estas redes son conocidas como redes académicas avanzadas o redes de investigación [2], y su característica principal es que ofrecen una red de prueba segura dirigida al ambiente de investigación y desarrollo, permitiendo a la comunidad de investigadores trabajar con mecanismos (herramientas o aplicaciones) de colaboración, para compartir información o el uso de recursos a través de esta serie de redes interconectadas [3].

Las redes académicas avanzadas, propiciaron el desarrollo de herramientas y aplicaciones tales como IPv6, IP Multicast y Calidad de Servicio (QoS), dando un fuerte impulso a la investigación y a la educación. Este recurso tecnológico y la rápida evolución de las tecnologías de telecomunicación, particularmente, las de intercambio y comunicación de datos, posibilita a los investigadores y académicos llevar cabo un sin número de proyectos afines que están en la frontera del conocimiento.

2 Identificación del Problema y Justificación

Las redes académicas y científicas, regionales y metropolitanas están poco desarrolladas en Argentina. Generalmente cada institución académica o científica tiene su propio acceso a la red institucional de su organización o a un proveedor de Internet privado y toda comunicación entre distintas instituciones se desarrolla a través de este esquema. Además la Argentina sufre de un fuerte centralismo de la ciudad de Buenos Aires, de manera que la mayoría de las comunicaciones (públicas y privadas) de alguna manera se encaminan hacia nodos ubicados en Buenos Aires. Esto significa que las comunicaciones entre instituciones de una misma ciudad del interior, distantes algunos cientos de metros, deben pasar obligatoriamente por nodos centrales (NAP) ubicados en la ciudad de Buenos Aires, en el mejor de los casos, o en nodos del exterior, en otros casos. El diseño de las redes centradas en Buenos Aires no es privativo de las comunicaciones de datos, sino también se puede apreciar en el tendido de carreteras y de líneas férreas.

Además y debido a este esquema, los anchos de banda de acceso a Internet de las instituciones son en general bajos o muy bajos en comparación de instituciones similares características de países vecinos, dificultando la implementación de tecnologías de avanzada. Al ser topologías WAN en estrella, las redes académicas argentinas actuales resultan en estructuras ineficientes, costosas y no fomentan la cooperación entre instituciones vecinas.

Algunas iniciativas de redes académicas se han realizado tales como la RIU (Red de Interconexión Universitaria - www.riu.edu.ar) que nuclea a la mayoría de las universidades nacionales públicas y la RUT (Red Universitaria Tecnológica – www.utn.edu.ar/secretarias/tic/rut.utn) que conecta a las Facultades Regionales de la Universidad Tecnológica Nacional dispersas en el territorio nacional. Pero estas redes, que son de carácter nacional, sólo relacionan a instituciones de la misma naturaleza y presentan también el mismo problema de centralismo, es decir sus nodos principales están ubicados en la ciudad de Buenos Aires.

El proyecto, tiende a subsanar los problemas planteados implementando una red experimental metropolitana con la participación de instituciones académicas y científicas de la ciudad de Mendoza, y que entre sus objetivos, también sirva de ejemplo para que otras ciudades/regiones adopten soluciones similares. Esta red se basa tanto en infraestructura inalámbrica como de fibra óptica. Inicialmente se ha previsto implementar todas las características esperadas compatibles con el protocolo IPv4 y gradualmente el montaje del protocolo IPv6 nodo a nodo, obteniendo así, una infraestructura que permite el ensayo de nuevas tecnologías y aplicaciones emergentes basadas en IPv6 inclusive. En su diseño se privilegian características de robustez, confiabilidad y bajo costo operativo, destacando la facilidad de instalación inicial.

3 Antecedentes

Hay varios ejemplos en el mundo de Redes Avanzadas centradas en IPv6 desde la red de avanzada por excelencia como Internet 2 (www.internet2.edu) en Estados Unidos, pasando por la red europea Geant2 (www.geant.net) y la asiática APAN (Asia-Pacific Advanced Network Consortium-apan.net). También en Latinoamérica han sido creadas tales redes, como: Clara (Cooperación Latinoamericana de Redes Avanzadas - www.redclara.net), RNP de Brasil (RNP – Rede Nacional de Ensino e Pesquisa - www.rnp.br), CUDI de México (Corporación Universitaria para el Desarrollo de Internet – www.cudi.edu.mx), Reuna de Chile (Red Universitaria Nacional - www.reuna.cl) y RENATA (www.renata.edu.co) en Colombia, de notable crecimiento. También esta el caso de la red Innova (www.innova-red.net) en Argentina, que da acceso a redes avanzadas a las instituciones nacionales por intermedio de Clara.

Un ejemplo significativo de redes es la red Redecomep (www.redecomep.rnp), que es una iniciativa del Ministerio de Ciencia y Tecnología de Brasil (MCT), coordinada por Rede Nacional de Ensino e Pesquisa (RNP), que tiene como objetivo implementar redes de alta velocidad en regiones metropolitanas del país servidas por los Puntos de Presencia de RNP. Ya en el año 2004, Redecomp implementó una red piloto en el área metropolitana de Belém, capital del estado de Pará. Allí se montó infraestructura óptica para 12 instituciones logrando mejorar la conectividad y bajando significativamente los costos.

En Argentina, en el ámbito de la UTN (Universidad Tecnológica Nacional) existe la red RUT2 (Red Universitaria Tecnológica 2) que conecta a las Facultades Regionales pero tiene el inconveniente antes mencionado, a saber; el de basar su infraestructura en una topología WAN en estrella, con nodo central en Bs. As. y enlaces institucionales de ancho de banda extremadamente bajo (2/4Mbps en promedio por institución), además, no se ha desplegado el protocolo IPv6 sobre esta red hasta la fecha y no se desplegaría antes del 2017.

Es así que tanto en Argentina como en la mayor parte de Latinoamérica, no están desarrolladas las redes metropolitanas/provinciales como en Brasil.

3.1 Acerca de Red Experimental

La Red Experimental [4] pretende ser la red de interconexión de avanzada de las instituciones metropolitanas de Mendoza. Actualmente está siendo impulsada por una serie de proyectos de aplicación activos; desde conectividad alternativa (red de transporte) entre dependencias de instituciones, o bien, la instalación y evaluación de aplicaciones competitivas y cooperativas, tales como IPv6, Movilidad, Clustering, Virtualización, VoIP, Computación de Alto Desempeño, IOT entre otras.

Características:

- Ámbito colaborativo interinstitucional
- Sinergia y recursos compartidos
- Grandes anchos de banda y calidad de servicio
- Disponibilidad de nuevo protocolo IPv6
- No distribuye "default gateway"
- Proyectos de aplicación interinstitucionales
- Políticas de acceso a la red normalizadas
- Enlaces propios de las instituciones

3.2 Topología proyectada

Se trata una topología MAN/WAN tipo malla, como se aprecia en la figura 1, en donde cada institución miembro se conecta a la red mediante dos encaminadores de base; uno de borde denominado ANR (ACyT.net Network Router), que forma parte de la estructura troncal o de núcleo y un segundo tipo de encaminador denominado rName, de contacto directo con las redes de la institución.

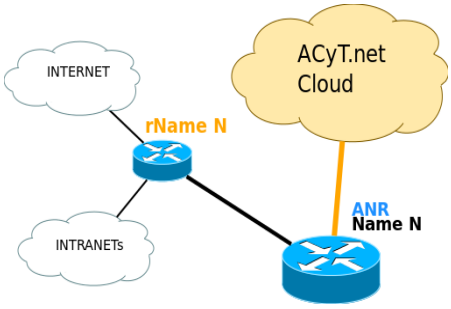


Fig. 1. Diseño conceptual y arquitectura global

Los encaminadores ANR aseguran la conectividad entre otros ANR y conforman el troncal de una red orientada al transporte de datos transparente; una virtual nube de ACyT.net. Por su parte los encaminadores de la institución rName se constituyen en el nexo que vincula las redes de cada institución con la nube de ACyT.net, siendo su función primaria la de declarar y propagar las redes públicas concernientes a cada institución (IPv4/IPv6) y que serán conocidas a través de toda la nube ACyT.net.

ACyT.net se conforma mediante un esquema de topología MAN/WAN tipo malla para mayor robustez y confiabilidad de la infraestructura base. Se pretende que cada uno de los nodos ANR sean conectados con un mínimo de dos enlaces hacia ANR distintos para asegurar los caminos alternativos ante fallas y cortes parciales de enlaces. Se considera obligatorio y como parte de la política de acceso a la red el uso de los protocolos de encaminamiento dinámico OSPF/OSPF6 [5] entre distintos nodos ANR y RIP2/RIPng [6] entre los encaminadores de la institución rName y los ANR, como se denota en la figura 2.

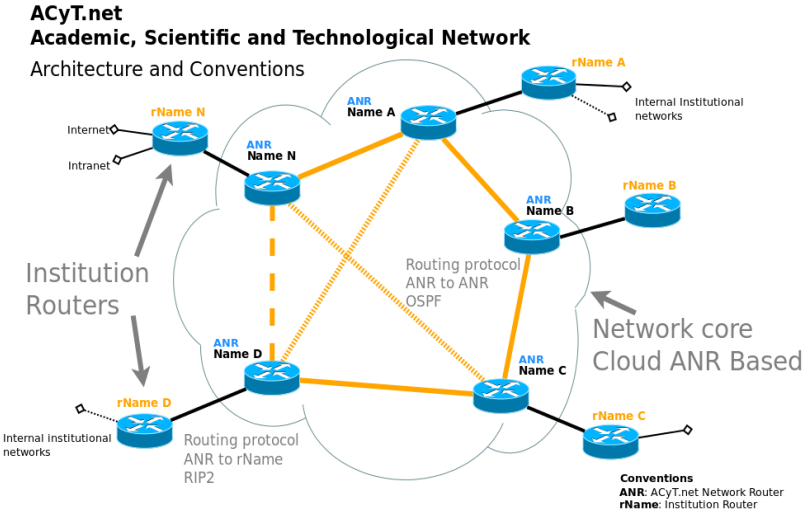


Fig. 2. Arquitectura y convenciones

Cada par de encaminadores rName y ANR se dispondrán en un mismo lugar físico, preferentemente dentro de los centros de datos de las instituciones respectivas.

4 Objetivos y Resultados

4.1 Objetivo Principal

Estudiar, diseñar e implementar una red metropolitana experimental de avanzada, denominada ACyT.net, que permita conectar en forma confiable, con altas prestaciones y bajo costo a instituciones académicas y científicas de la ciudad de Mendoza; y que sea útil para capacitar, difundir y diseminar en el medio local y nacional, el estudio, diseño e implementación de redes interinstitucionales de tecnología avanzada.

4.2 Objetivos Secundarios

- Punto de presencia POP IPv6 único para todas las instituciones participantes.

- Convergencia de multimedia y Qos sobre redes de avanzada con IPv6.
- Movilidad en Redes de Avanzado con IPv6: Organizar, implementar y evaluar PMIP (Proxy Mobile IP), a través de unos prototipos en red de avanzada con IPv6.
- Computación de Alto Desempeño y GPU aplicado a seguridad en redes de avanzada e IPv6
- IOT: Diseñar y permitir acceso a Redes de Sensores en IPv6, en ambientes distribuidos.
- Difusión y Capacitación: Participar en el fortalecimiento y difusión de las redes de avanzada y sus aplicaciones, incentivando su uso e implementación en el medio regional.

4.3 Avances y resultados preliminares

ACyT.net actualmente está en fase de desarrollo y con algunas implementaciones en marcha. Al momento la red une a 4 instituciones locales; el Centro Científico Tecnológico de Mendoza (Centro Científico y Tecnológico Mendoza - www.mendoza-conicet.gob.ar), la Regional Mendoza del Instituto Nacional del Agua (INA - www.ina.gov.ar), el Grupo de Investigación y Desarrollos en TICs (GridTICS - www.gridtics.frm.utn.edu.ar) de la Universidad Tecnológica, Facultad Regional Mendoza y Facultad de Ciencias Agrarias de la Universidad Nacional de Cuyo (www.fca.uncu.edu.ar)

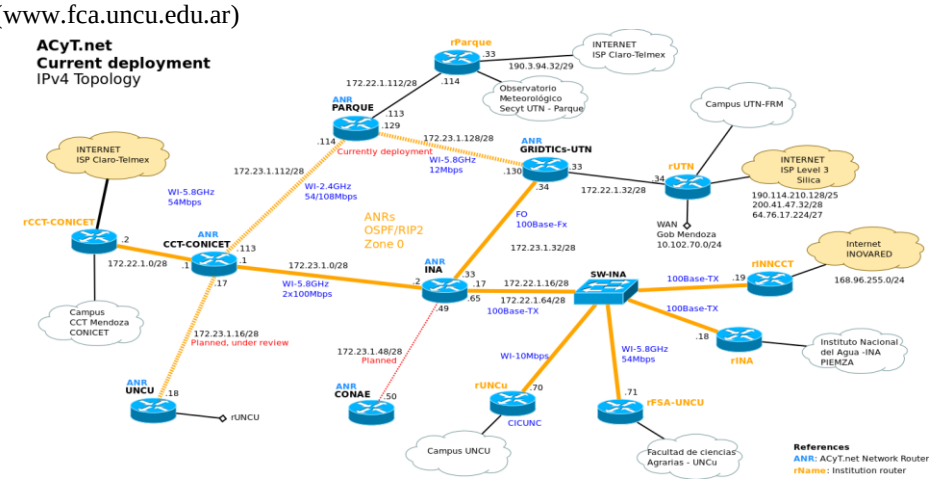


Fig. 3. Despliegue de ACyT.net actualmente implementada.

La tecnología de interconexión desplegada es mayoritariamente inalámbrica con algunos enlaces implementados en fibra óptica y su topología se puede ver en la figura 3.

En lo referente a aplicaciones, se avanza sobre el objetivo de "Convergencia de multimedia y Qos sobre redes de avanzada con IPv6", realizando un análisis de convergencia de multimedia y Qos sobre IPv6 y diseñando e implementando funcionalidades de VoIP y sus sistemas más usados.

Igualmente se avanza sobre el objetivo de "Movilidad en redes de avanzada con IPv6" [7], previendo en primer lugar, la implementación de Movilidad IPv6 (MIPv6) [8]. Un Home Agent vinculado a ACyT.net proveerá movilidad global a los usuarios locales de la red, permitiéndoles conservar su dirección IPv6 local independientemente del punto y tecnología de conexión a Internet. Posteriormente, se implementará un esquema de movilidad localizada, Proxy Mobile IPv6 (PMIPv6) [9], orientado a mejorar la experiencia en el uso de aplicaciones en tiempo real dentro del ámbito de la red de avanzada.

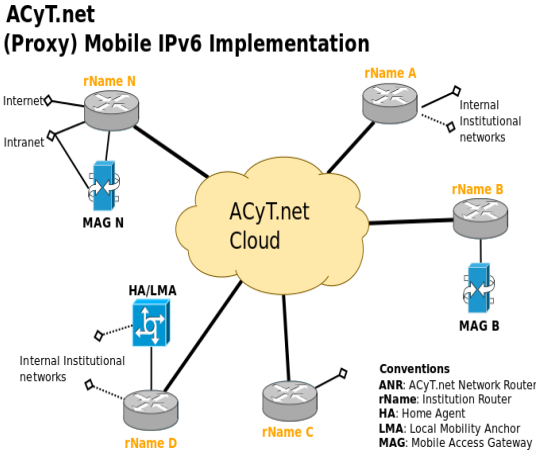


Fig. 4. Implementación de PMIP6

Se avanza sobre el objetivo “Sistema de paralelismo y Distribución de cómputo para la Visualización de Volúmenes en tiempo real”, el cual permite la representación gráfica (rendering) distribuida de volúmenes mediante la utilización de numerosos nodos CPU-GPU procesadores e integradores conectados por red Gigabit Ethernet con la utilización de un protocolo [10] especialmente diseñado para tal fin. Si bien actualmente tanto el control del sistema visualizador como la visualización del resultado se realiza sobre la misma red local Gigabit en donde se realiza el procesamiento, se está desarrollando una implementación que permite el control mediante IPv6 multicast y la visualización sobre IPv6 unicast mediante compresión H264 en tiempo real con GPU ambos remotamente desde puntos alejados en la red ACyT.net según lo establecido en RFC 3306 [11] tal como se puede observar en la figura 5.

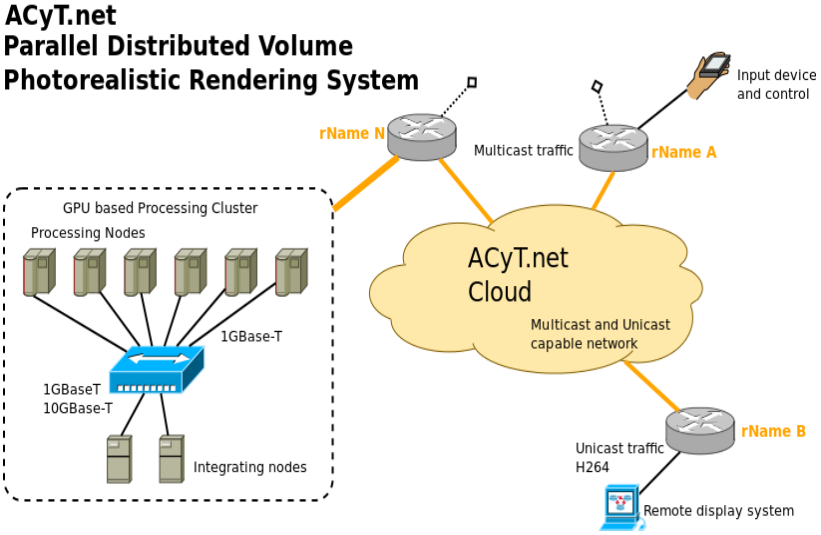


Fig. 5. Concepto de sistema de procesamiento de volúmenes gráficos fotorealistas integrado a ACyT.net con capacidad de tráfico multicast.

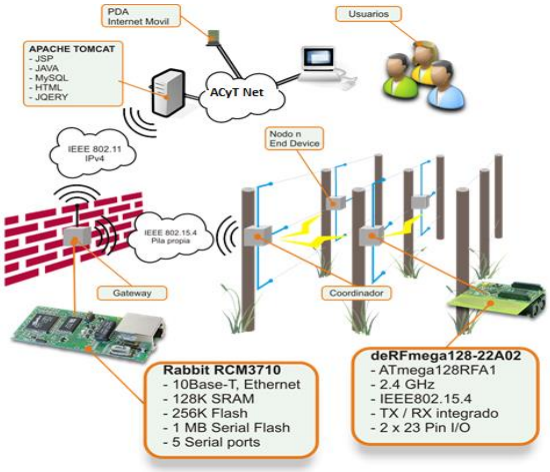


Fig. 6. Red SIPIA, (gridTICs) para la WSN

Se avanza sobre el objetivo "Clustering y Virtualización en redes de avanzada con IPv6" [12] realizando estudios de métodos y tecnologías de clustering y virtualización y su implicancia en IPv6. Se comienza con la construcción de la extensión de la red ACyT.net desde su nodo principal hasta el laboratorio del gridTICS de la UTN mediante el tendido de fibra óptica.

Se avanza en el objetivo de integrar Redes de Sensores a la red. Se ha diseñado e implementado la red SIPIA [13], red de sensores inalámbricos basada en la norma IEEE 802.15.4, aplicada al entorno agropecuario en el ámbito de agricultura de precisión, en convenio entre la UTN-FRM y la Facultad de Ciencias Agrarias de la Universidad Nacional de Cuyo (www.fca.uncu.edu.ar). La red SIPIA se encuentra

implementada en una parcela de la FCA y se conecta a la red experimental, como se observa en la figura 6.

4. Líneas de investigación y Agradecimientos

El proyecto ACyT.net tiene sus antecedentes de creación, sostén y apoyo tecnológico, académico y financiero en los proyectos de Investigación y Desarrollo.

"CODAREC6 INTRANET Diseño y simulación de la implementación de tecnologías y procedimientos de transición del protocolo IPv6 en INTRANETS usando el `ipv6 test bed" UTN FRM, [14]

"SARA-Six: Análisis, implementación y evaluación de servicios colaborativos competitivos Aplicados a Red de Avanzada experimental de la región", [12] [15][16]

"SARA PMIP - Análisis, Evaluación y Comparación de ambientes Proxy Mobile IP en versión 6, aplicado a Redes de Avanzada" [7] [17]

PID FAC PICTO 2010/29 "Visio GPU - Procesamiento para visualización utilizando algoritmos paralelos en GPU y distribuidos en red" UTN FRM, Director. Pérez Monte, Cristian Federico, 2011 – 2012 [10] [18]

"RED SIPIA - Estudio a campo de red de sensores inalámbricos para adquisición de parámetros ambientales, de uso en investigaciones agronómicas y biológicas", [13] [19] [20]

"SARA-Operation: Diseño y evaluación de metodologías operativas para Servicios colaborativos competitivos aplicados a Red de Avanzada experimental de la región", [4]

5. Referencias

1. José Silvio "Redes Académicas y Gestión del Conocimiento en América Latina: En Busca de la Calidad Educación Superior y Sociedad", Vol 3, No 2 1992 Copyright © 2010 UNESCO-IESALC.
2. Carmen Labbé, "Redes Avanzadas y Comunidades de Investigación para la Gestión del Conocimiento" Organización Universitaria Interamericana, Dec 2009
3. Pedro Munive S. y Mariana Rodríguez M, "Las redes académicas avanzadas, una oportunidad para compartir eventos interactivos de divulgación", X Reunión de la Red de Popularización de la Ciencia y la Tecnología en América Latina y el Caribe (RED POP - UNESCO) San José, Costa Rica, 9 al 11 de mayo, 2007
4. Gustavo Mercado, Javier Gitto, Rubén Soria, Carlos Taffernaberry, Raúl Moralejo, Cristian Pérez Monte, María Inés Robles, Marcela Orbiscay, Sebastián Tobar, Carlos Tiviroli, Pablo Castorino, Joel Noguera,"ACyT.net Red Académica Científica y Tecnológica Experimental de Mendoza Argentina", WICC 2014, XVI Workshop de Investigadores en Ciencias de la Computación ISBN: 978-950-34-1084-4, organizado por la Universidad Nacional de Tierra del Fuego, Ushuaia, Tierra del Fuego, Argentina, 7 y 8 mayo 2014.
5. R. Coltun, D. Ferguson, J. Moy, A. Lindem "OSPF for IPv6" , RFC 5340, IETF, July 2008
6. G. Malkin, R. Minnear, "RIPng for IPv6", RFC 2080, IETF, January 1997
7. Carlos Taffernaberry, Sebastián Tobar, Gustavo Mercado, Joel Noguera,Cristian Perez Monte, Raúl Moralejo, Santiago Perez, "IPv6: Comparison of mobile environments in academic networks", Anales de CACIC 2014 XX CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN, Universidad de Moron, 20 al 24 de Octubre de 2014.
8. C. Perkins, D. Johnson, J. Arkko, "Mobility Support in IPv6", RFC 6275, IETF, July 2011

9. S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, B. Patil, " Proxy Mobile IPv6", RFC 5213, August 2008
10. C. F. Perez-Monte, G. J. Mercado, J. C. Taffernaberry , M. F. Piccoli. "Protocolo de comunicaciones para renderización distribuida en tiempo real". I Workshop Pre-IETF - CSBC 2014 - XXXIV Congresso Da Sociedade Brasileira de Computao - Sistemas Sociais e Eventos de Grandes Massas: Ampliando Desafios da Computação. PP 2172-2182. ISSN 2175-2761. Brasilia, Brasil. Julio 2014
11 B. Haberman y D. Thaler, RFC 3306 "Unicast-Prefix-based IPv6 Multicast Addresses", IETF , Agosto de 2012.
12 Gustavo. Mercado, Carlos Taffernaberry, Raúl Moralejo, Cristian Pérez Monte, María Inés Robles, Marcela Orbiscay, Sebastián Tobar, Patricia Clérigo, Carlos Tiviroli, "SARA Six Análisis, Implementación y Evaluación de Servicios Colaborativos Competitivos Aplicados a Redes de Avanzada" publicado en Anales del XV Workshop de Investigadores en Ciencias de la Computación. (WICC 2013). ISBN 978-987-28179-6-1
13 Gustavo Mercado, Roberto Borgo, Francisco González Antivilo, Gisela Ortiz Uriburu, Ana Diedrichs, Sebastián Tromer, Nicolás Ledezma, Matías Aguirre, Cristian Panella, Germán Tabacchi, Juan Martí, Guillermo Grünwaldt, Ignacio Rigoni, Gabriel Antón, Fernando Chamorro, Rodrigo Moreno, Santiago Pérez,"RED DE SENSORES SIPIA", WICC 2012 - XIII Workshop de Investigadores en Ciencias de la Computación, ISBN: 978-950-34-0756-1, Abril 2012, Misiones.
14 C. Taffernaberry, G. Mercado, C.Pérez, R. Moralejo, S.Pérez y S. Tobar, "Pro6: Implementación de una Aplicación Layer Gateway (ALG) para transición hacia redes IPv6" EnIDI 2009 Encuentro de Investigadores y Docentes de Ingeniería UTN Facultad Regional San Rafael, ISBN 978-950-42-0121-2, 12 al 14 de noviembre 2009, San Rafael, Mendoza,
15 Cristian Pérez Monte, María Inés Robles, Gustavo Mercado, Carlos Taffernaberry, Marcela Orbiscay, Sebastián Tobar, Raúl Moralejo y Santiago Pérez, "Implementation and Evaluation of Protocols Translating Methods for IPv4 to IPv6 Transition", Journal of Computer Science & Technology, ISSN 1666-6038 Vol. 12 No. 2 August 2012
16 G. Mercado, C. Pérez Monte, C. Taffernaberry, M. Robles, M. Orbiscay, S. Tobar, R. Moralejo, S. Pérez, "Implementación y Evaluación de métodos de Traslación de Protocolos para la transición IPv4-IPv6". Anales del CACIC 2011,ISBN 978-950-34-0756-1,UNLP,La Plata, Octubre 2011.
17 C. Taffernaberry, G. Mercado, S. Tobar, C. Pérez Monte, P. Clérigo,I. Robles, M. Orbiscay, S. Pérez, R. Moralejo, "PMIP6: Análisis, Evaluación y Comparación de ambientes Proxy Mobile IP en versión 6, aplicado a Redes de Avanzada", Anuales del WICC 2011. XIII Workshop de Investigadores en Ciencias de la Computación, ISBN 978-950-673-892-1, UNR Editora. Editorial de la Universidad Nacional de Rosario, 2011
18 C. F. Perez-Monte, M. F. Piccoli, C. Luciano, S. Rizzi. "Sistema de Comunicación para Renderización Paralela de Volúmenes en Tiempo Real", CoNaIISI 2014, 2° Congreso Nacional de Ingeniería Informática/Sistemas de Información. ISSN: 2346-9927. San Luis, Argentina. Noviembre 2014 PP. 1250-1259.
19 Ana Laura Diedrichs, María Inés Robles, Diego Dujovne, Facundo Bromberg and Gustavo Mercado, "Characterization of LQI behavior in WSN for glacier area in Patagonia Argentina", publicado en Regular Papers, 2013 Fourth Argentine Symposium and Conference on Embedded Systems (SASE/CASE), Book ISBN 978-1-4799-1098-4, Printed Book IEEE CATALOG CFP1346V-PR, pag 15-20, Buenos Aires, Aug 14th-16th, 2013
20 Ana Laura Diedrichs, Germán Tabacchi, Guillermo Grünwaldt, Matías Pecchia, Gustavo Mercado and Francisco Gonzalez Antivilo,"Low-power wireless sensor network for frost monitoring in agriculture research", Anales del Biennial Congress of Argentina (ARGENCON), IEEE, 11 la 13 de Junio 2014, Bariloche.

Estrategias de control topológico para reducir la interferencia en Redes Inalámbricas Multisalto

Nelson Rubén Rodríguez[1], María Antonia Murazzo[2], Edilma Olinda Gagliardi[3]
[1,2]Departamento e Instituto de Informática, Universidad Nacional de San Juan, Complejo Universitario Islas Malvinas, Rivadavia, San Juan, Argentina
[3]Departamento de Informática, Universidad Nacional de San Luis, San Luis, Argentina

nelson@iinfo.unsj.edu.ar[1], maritemurazzo@gmail.com[2], oli@unsl.edu.ar[3]

Abstract. Las redes inalámbricas presentan características distintivas en cuanto a interferencia, tiempo de vida de batería y pérdida de conectividad en redes móviles. Diversas estrategias se han tratado de proponer ya sea a nivel de la capa MAC, o mediante la reducción de tráfico. El control topológico es una estrategia basada en la teoría de grafos y la geometría computacional que permite obtener un subgrafo que mantenga la conectividad, dado que la comunicación se efectúa por medio de saltos intermedios, y reduzca la interferencia para cada tipo de red. Este trabajo presenta estrategias aplicadas para la reducción de la interferencia en redes multisalto mediante el control topológico y las compara con el modelo basado en distancia, que es el tradicional que utilizan los algoritmos de ruteo.

Keywords: Topological Control, Interference, MANET, Wireless Networks

1 Introduction

El avance de las redes inalámbricas ha sido sorprendente. Esto se debe a muchas razones como son: la expansión de la telefonía celular, la reducción de costos de todos los dispositivos que tienen que ver con la comunicación (*handsets*, *switchers*, *routers*, antenas, etc.), la expansión de los *hotspots*, el incremento de las ventas de los equipos de usuarios (celulares, teléfonos inteligentes, *notebooks*, *tablets*, *netbooks*), la difusión de la tecnología por parte de usuarios Linux (creando comunidades inalámbricas) y la cultura a los *gadgets* por parte de los jóvenes [1]. El rápido desarrollo que las redes inalámbricas han experimentado en los últimos años ha permitido ofrecer a los usuarios diferentes soluciones. Se están presenciando cambios fundamentales en la tecnología, servicios, y los paradigmas de negocios. Estos cambios son esenciales para la creación del paradigma de comunicación conocido como 4A ("*anytime*, *anywhere*, *anyhow*, *anyone*") o sea en cualquier momento, lugar con cualquier dispositivo y como sea [2].

El área de aplicación de estas redes es amplia, como misiones de búsqueda y rescate, monitoreo de medio ambiente, vigilancia, comunicación militar en campo de

batalla, exploración del espacio, operaciones bajo el mar, sistemas de transporte inteligente, sensores para la agricultura, etc. Además pueden ser usadas como tecnología de la última milla para proveer acceso Internet en ciudades altamente pobladas [3].

Las redes Mesh son utilizadas en redes comunitarias o de vecindad y también para proveer Internet de banda ancha para redes "caseras", interconexiones de equipamiento de consumo electrónico y comunicaciones de servicio público, cobertura WLAN puertas adentro y acceso de usuarios móviles [4].

Las WSN se utilizan además para entornos inteligentes, en la toma de datos en áreas peligrosas como volcanes o químicos industriales, temperatura, humedad y radiación en zonas inhóspitas, monitoreo de control (por ej. robots), casa inteligentes con alta seguridad, seguimiento, identificación y personalización [5].

En el caso particular de las redes Ad hoc, son muy convenientes para situaciones donde la infraestructura o no está disponible o no es confiable, mientras que las redes Mesh deben contar con una infraestructura para ofrecer los servicios para la que fueron diseñadas.

2 Wireless Networks

Las redes inalámbricas multisalto pueden ser clasificadas según sus características generales en 4 tipos: redes ad-hoc, redes de sensores, redes Mesh y redes híbridas.

Las redes ad-hoc pueden ser fijas o móviles, en este último caso se denominan MANETS (Mobile Ad hoc Networks), y la movilidad agrega una dificultad más a la conectividad de la red. La característica más destacada de las MANETS es que a diferencia de las redes WLAN, no dependen de ninguna infraestructura fija para establecer la comunicación, en cambio, los nodos inalámbricos cooperan entre sí para establecer la comunicación. Las MANETs por lo tanto son conocidas como redes inalámbricas sin infraestructura [6].

Los objetivos de las redes son diferentes. La siguiente ilustración (Fig. 1) permite reflejar gráficamente las coincidencias y diferencias de las mismas.

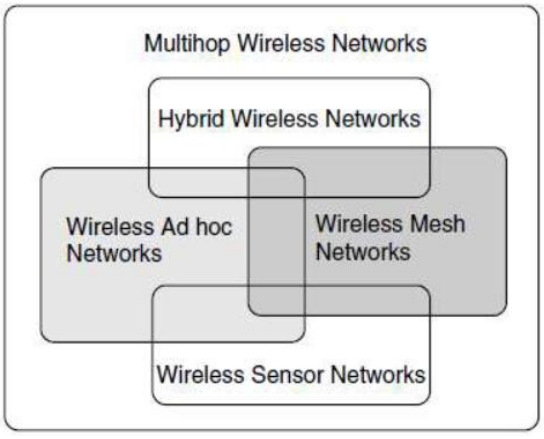


Fig. 1. Clasificación de distintas redes inalámbricas multisalto [7]

A diferencia de las MANETs, los nodos en las WMNs (Wireless Mesh Networks) son estacionarios y los cambios dinámicos de la topología son menos frecuentes. También los nodos inalámbricos en WMNs son puntos de accesos y Gateway de Internet y por no tanto no presentan restricciones de energía.

Por otro lado, las WSN (Wireless Sensor Networks) usualmente presentan una alta densidad de nodos, debido a que por restricciones de energía, muchos sensores pueden quedarse sin batería, inclusive aquellos con energía solar.

Las WSN usualmente alcanzan velocidades menores que el resto debido a reducción de consumo de energía y la utilización de protocolos como *Zigbee* (del orden de Kbps). A diferencia las WMN, al soportar múltiples canales, permiten alcanzar velocidades bastante mayores.

La ventaja principal de una WMN es su inherente tolerancia a fallas, la simplicidad de configuración de red y la capacidad para soportar banda ancha. A diferencia de las redes celulares donde la falla de una sola estación base deja a una región geográfica sin servicio, las WMN proveen alta tolerancia a fallas, aun cuando algún nodo deja de funcionar. En cambio, en WSN la única manera de proveer tolerancia a fallas es con una mayor densidad de sensores.

Las redes inalámbricas híbridas pueden incluir una variedad de tecnologías tanto para el *backbone* como para el *back haul*. Las redes híbridas son aquellas en las cuales cualquier nodo móvil puede tener conectividad de forma directa o vía un nodo *gateway*, para una infraestructura de red. Esta red puede ser una red IP como Internet, una red WAN 3G o LTE o una WLAN 802.11. Actualmente otras tecnologías de red pueden ser consideradas. [8].

3 Consumo de energía

El consumo de energía es quizás el problema más importante de las redes inalámbricas, dado que un buen uso de la misma permite que extender la vida de la red. En redes de sensores y redes ad-hoc, la energía es consumida principalmente por 3 propósitos: transmisión de datos, procesamiento de señal y operación de hardware.

El consumo de energía es mayor cuando se transfieren con más potencia para lograr más alcance. Por ejemplo un sensor puede consumir 35 mA transmitiendo a 8dBm, mientras que a 4dBm consume 27 mA Estos valores son aproximados dado que dependen de diversos factores como tipo de sensor y protocolo de comunicación, pero en todos los casos aumentar la potencia de transmisión para alcanzar a más nodos incrementa notablemente el consumo [9] [10].

Teniendo en cuenta lo expresado, si se ajusta la energía de transmisión de cada nodo de forma tal de mantener las red conectada y llegar de un nodo a otro de forma directa o por medio de saltos si está más alejado, el consumo de energía resultará menor. Además si se reduce la interferencia producida por la conexión de todos con todos, la cantidad de retransmisiones por ruido se reducirán también.

La eficiencia de energía de un nodo es definida como la relación entre la cantidad de datos entregados por un nodo y el total de energía gastada. Alta eficiencia de energía implica que un gran número de paquetes puedan ser transmitidos por un nodo

con una determinada cantidad de reserva de energía. Las principales razones para la administración de energía en redes ad-hoc inalámbricas y WSN son: reserva de energía limitada, dificultades en el reemplazo de baterías, falta de coordinación central, restricciones en la elección de fuentes de energía y utilización de canales compartidos.

Por otro lado hay que tener en cuenta que un modelo de consumo de energía debe incluir los cuatro estados en los que se encuentra un nodo: adquisición, transmisión, recepción y espera. Aplicando las estrategias adecuadas se reducen los consumos de empaquetado de datos, codificación, entramado y envío para la transmisión, decodificación, detección de error y chequeo de la dirección para recepción.

4 Interferencia

El problema de la interferencia ocurre cuando un mensaje de un nodo se corrompe debido a otra transmisión concurrente de otro nodo en el mismo rango de transmisión. Para una red, cada nodo genera una transmisión que es modelada con un disco o círculo, la interferencia de un nodo es definida como el número de discos que lo incluyen. La interferencia causa pérdida de mensajes lo cual resulta en un alto consumo de energía. Esta situación puede frecuentemente ocurrir en redes densas con alto tráfico.

Algunos modelos de interferencia como el publicado por Meyer [11], están basados en el tráfico corriente de red. La cantidad y naturaleza del tráfico de red, sin embargo este tráfico es altamente dependiente de la aplicación elegida. Debido a que usualmente no existe información a priori acerca del tráfico de red, un modelo estático de interferencia es deseable, dado que depende solamente del conjunto de nodos.

Sugerencias de la capa física para reducir la interferencia, como analizadores de espectro, pueden resultar útiles en redes estáticas, al igual que adquirir *switchers* que administren problemas de interferencia; pero en MANETs y WSN móviles es imposible de llevar a cabo.

Otra de las alternativas es sobre diseñar la capacidad de la red, agregando alta densidad de puntos de acceso, pero esto puede producir interferencia de co-canal y tampoco es aplicable en MANETs y WSN.

Existen diversos trabajos que apuntan a reducir la interferencia en redes inalámbricas y en particular en MANETs y redes de sensores. Pero no todas las publicaciones se basan en control topológico utilizando geometría computacional. Algunos de los trabajos publicados hacen estimaciones estadísticas de la interferencia basada en características de la subcapa MAC (con sus variantes) por ejemplo lo referido en el capítulo 8 del libro “Ad-hoc Networks” de Ramin Hekmat [12], otro trabajo presenta un modelo para analizar la performance de las estrategias de transmisión en redes de packet radio multisalto donde cada estación tiene radio de transmisión ajustable [13]. En ambos casos se presentan análisis de *throughput* pero no se proponen cambios ni estrategias para mejorar la eficiencia de las redes.

A menudo se argumenta que las tecnologías dispersas con pequeños grados o limitadas en grados son apropiadas para reducir la interferencia. Sin embargo grados

bajos no necesariamente implica baja interferencia. Además se puede demostrar que la mayoría de los algoritmos de control topológico obtienen valores que no están cercanos a la interferencia óptima, y por lo tanto se debe encontrar el grafo que minimice la interferencia en primer lugar [14].

5 Control Topológico

En redes inalámbricas cada nodo de la red tiene la posibilidad de variar la topología de la red por medio del ajuste de la potencia de transmisión en relación a otros nodos vecinos. En contraste las redes cableadas tienen una infraestructura fija pre-configurada. Este ajuste podría ir al extremo de tener cero alcance (sin consumo de energía en transmisión pero resultando una red in conexa), hasta alcanzar a todos los nodos resultando conexa pero con interferencia máxima.

La razón fundamental del esquema de control topológico es proveer un mecanismo de control para mantener la conectividad de la red y optimizar la performance para prolongar el tiempo de vida de la red y maximizar la velocidad de transferencia de la misma [15]. Por otro lado, establecer la conectividad en redes inalámbricas puede ser una tarea compleja para lo cual varios (algunas veces en conflicto) objetivos puedan necesitar que sean optimizados. Para permitir que un paquete sea ruteado desde cualquier nodo origen a cualquier nodo destino de la red, el correspondiente grafo de comunicación debe estar conectado. En resumen, además del requerimiento de conectividad, varias propiedades pueden ser impuestas en la red., incluyendo mejor consumo de potencia, carga promedio de tráfico limitada, menor cantidad o distancia de saltos promedio entre pares de emisor-receptor y existencia de mínima interferencia [16].

Una red inalámbrica puede depender de factores incontrolables tales como movilidad de los nodos, clima, interferencia, ruido y también factores controlables como son potencia de transmisión, comunicación con antenas direccionales y comunicación multicanal.

Según el tipo de red, el objetivo del control topológico es diferente. En WMN se trata de obtener rutas con baja latencia o alta velocidad de transferencia que sean eficientes en consumo de energía. Este consumo es sumamente importante en WSN y en redes ad hoc, por lo que en este tipo de redes, se trata de reducir la interferencia para obtener menos cantidad de retransmisiones.

Las técnicas empleadas para aplicar el control topológico se basan en la geometría computacional. Dicha disciplina brinda un marco teórico y formal para el diseño de estructuras y análisis de algoritmos requeridos para dar soluciones a problemas que surgen en diversas áreas de la informática, resolviendo los mismos aplicando geométrica de modo constructivo [17]. Existen una variedad de trabajos publicados sobre la construcción de algoritmos sobre grafos aplicando estas técnicas e inclusive resolviendo problemas de ruteo en redes [18].

Las redes inalámbricas son modeladas como Grafo de Disco Unidad (UDG) [19]. En este tipo de grafo, dos nodos sólo pueden comunicarse si la distancia que los separa es como máximo la unidad, entendiendo por unidad el radio de alcance de cada nodo. Suele utilizarse también en intersección con otros grafos planos para

obtener planaridad. Se muestra en la figura 2 una arista que es la comunicación entre dos nodos y un grafo de disco unidad de ejemplo para una red dada. A partir de este disco unidad se trata de obtener un grafo conexo que minimice la interferencia.

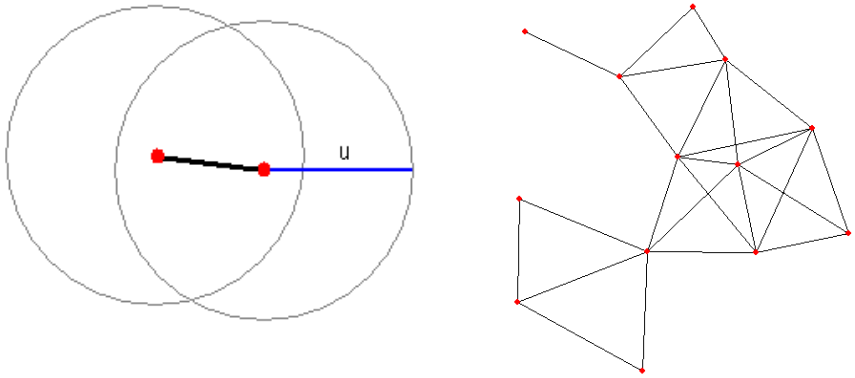


Fig. 2 Arista del UDG y Grafo de Disco Unidad [20]

Debido a limitaciones de potencia y memoria, un nodo inalámbrico prefiere solamente mantener la información de un subconjunto de vecinos con los cuales comunicarse, el cual es llamado control topológico. Los algoritmos de control topológico fueron diseñados para diferentes objetivos: minimizar la longitud máxima del enlace (o potencia del nodo) mientras mantiene la red conectada [21], limitar el grado de los nodos, limitar la tasa de expansión, o construir localmente un *spanner planar* (en teoría de grafos, el concepto de *t-spanner*, consiste en encontrar un subgrafo G′ de un grafo G tal que aproxima las distancias en G con un factor de precisión t). [22].

Algunos investigadores han probado reducir la interferencia disminuyendo el consumo de energía de cada nodo (o sea reduciendo la potencia de transmisión) o proveyendo controles para proveer topologías de bajo grado para los nodos, pero ninguna de estas estrategias garantiza reducir la interferencia en todos los casos.

Burkhart y otros [23] proponen varios métodos para construir topologías cuya interferencia del enlace máxima es minimizada mientras la topología mantiene a los nodos conectados.

En la mayoría de los casos en los cuales las redes ad-hoc y WSN son utilizadas, los nodos no tienen conocimiento de la ubicación de otros nodos. Con lo cual el objetivo es obtener un método para construir la red de forma local. Que es, que el radio de comunicación de un sensor o nodo no depende de la ubicación de un sensor o nodo que esté lejano [24].

Cada nodo de la red puede ajustar su potencia de transmisión a cualquier valor entre 0 y su máximo nivel de potencia de transmisión. El algoritmo de control topológico obtiene un subgrafo a partir del grafo origen, reduciendo la potencia de transmisión de los nodos, teniendo en cuenta la interferencia de los enlaces.

6 Simulaciones

Las topologías regulares (triángulo, cuadrado, hexágono, u otras figuras geométricas) y las distribuciones aleatorias uniformes son ampliamente usadas en modelos tratables analíticamente. Pero los casos reales resultan en topología malla o arbitrarias, es cuyo caso es más adecuado un modelo de simulación.
Los resultados presentados a continuación fueron desarrollados bajo determinados supuestos:

Los enlaces son bidireccionales y simétricos

Se considera a los nodos estacionarios, dado que pueden existir diferencias con los modelos de tráfico y movilidad.

El rango de transmisión de cada nodo es modelado por un círculo.

No se modelan antenas direccionales ni por sector.

Los nodos son generados aleatoriamente en el 50% de los grafos, en los restantes se generan grafos de topologías conocidas como lineales, anillos y figuras geométricas.

Los valores de las simulaciones se han contrastado con el modelo de distancia mínima. Es decir, el subgrafo obtenido reduce la interferencia y produce un grafo de mayor longitud, por otro lado el modelo de distancia mínima presenta el grafo de menor longitud pero con mayor grado de interferencia.

Se realizaron simulaciones para redes con 5 nodos, 10 nodos, 50 nodos y 100 nodos.

Para grafos con 5 nodos, en el 50% de los casos, coincidieron los valores de interferencia entre el subgrafo obtenido aplicando el modelo de interferencia y el modelo de distancia mínima; y por supuesto en el 50% restante el valor de interferencia resultó menor. En cuanto a la longitud total del grafo obtenido, también en el 50% de los casos coincidieron y en el resto la longitud del grafo de interferencia resultó mayor.

Para grafos de 10 nodos, los resultados fueron similares, pero un resultado importante es que en las topologías con nodos generados aleatoriamente la interferencia es importante y por lo tanto conviene un subgrafo basado en la interferencia, pero en topologías basadas en figuras geométricas, la interferencia resultante es baja y el modelo de distancia mínima resulta conveniente, porque mantiene la misma interferencia, pero minimiza la distancia total de la red.

Con 50 nodos y 100 nodos se mantienen las relaciones entre ambos modelos comparados, pero lógicamente los valores de interferencia al ser mayor cantidad de nodos, aumentan a 5 y 10 en promedio en ambas simulaciones, resultando conveniente el modelo de interferencia.

A modo de ejemplo en la fig. 3 se muestran los resultados obtenidos para grafos con 10 nodos, comparando el modelo de interferencia con el modelo de distancia mínima. Se puede notar en la gráfica que compara las interferencias (la de la izquierda) que el modelo de interferencia es mejor, y en la gráfica que compara las distancias el modelo de distancias presenta ventajas. No obstante cuando se transmite una retransmisión por interferencia produce mayor demora que unos metros más en el enlace y además consume energía de las baterías.

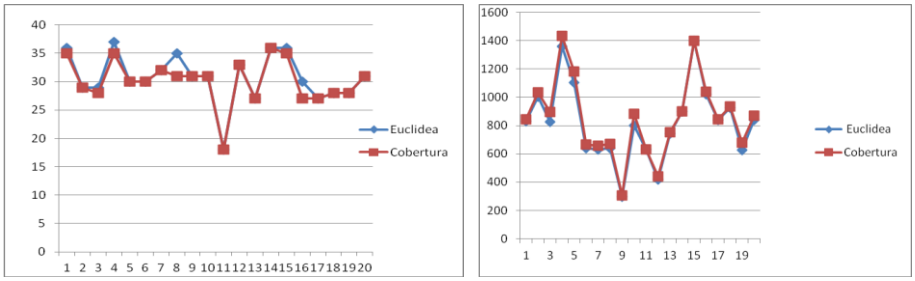


Fig. 3 a) Comparación Interferencia b) Comparación Longitud

7 Conclusiones y futuros trabajos

El control topológico permite obtener una subred de una red dada que mejora determinados parámetros de la red. Debido a que las redes inalámbricas son muy diversas, los objetivos de este control también cambian.

En WSN y MANET la reducción de la interferencia impacta directamente en el consumo de energía que en este tipo de redes determina la "vida" de la red. Sin embargo en WMN la performance es más importante dado que los nodos suelen estar conectados a alguna línea de alimentación, aunque en este tipo de redes si se reduce la interferencia también impacta en la performance.

Se prevé trabajar en un futuro con escenarios de movilidad, aunque sin implementar modelos de tráfico. También se prevé modelizar las redes aplicando el modelo de control topológico con comunicaciones cooperativas [25], pero contratándolos con los resultados obtenidos. Este tipo de sistemas son aquellas redes donde los usuarios comparten y coordinan sus recursos para mejorar la calidad de la transmisión de información.

También se ha previsto realizar el análisis de QoS por clase de aplicación (en este caso sin aplicar modelos de movilidad). Por ejemplo por un lado aplicaciones sensibles a la latencia y a la capacidad como voz sobre *wireless* LAN (VoWLAN) y por otro lado aplicaciones menos sensibles a la demora como navegación Web.

Referencias

1. Rodríguez N. : Nuevas configuraciones de redes inalámbricas y consideraciones en la elección. In: II Congreso de Informática del nuevo cuyo - Jornadas de Informática – San Juan (2007).
2. Wirth P., AT&T Labs: The Role of teletrafic modeling in the new communications paradigms. In: IEEE Communications Magazine. (1997).
3. Gaurilovska L., Prasad R. ; Ad hoc netwoking towards seamless communications. Ed. Springer, Holanda (2000).
4. Sichitiu M.: Wireless Mesh Networks: Opportunities and Challenges. In: Wireless World Congress.(2005). http://www4.ncsu.edu/~mlsichit/Research/Publications/wwcChallenges.pdf

5. Min R., et al.: Low Power Wireless Sensor Networks. In: in the Proceedings of International Conference on VLSI Design. Bangalore, India. (2001).
6. Chandra P.: Bulletproof Wireless Security - GSM, UMTS, 802.11, and Ad Hoc Security – Elsiever Communications engineering series pp. 121-127 (2005).
7. Garg P., Pawar N.: A Review on the topology control approaches for utilizing MRMC over WMNs. In: SSRG International Journal of Computer Science and Engineering (SSRG-IJCSE) – EFES (2015). ISSN: 2348 – 8387, www.internationaljournalssrg.org
8. Tchepnda , Moustafa H., Labiod H. France Telecom R&D: Hybrid Wireless Networks: Applications, Architectures and New Perspectives Christian Sensor and Ad Hoc Communications and Networks, SECON '06. 2006 3rd Annual IEEE Communications Society on. Date 28-28 (2006).
9. Puccinelli D., Haenggi M.: Wireless Sensor Networks-Applications and Challenges of Ubiquitous Sensing. In: IEEE Circuits and Systems Magazine, 5:19-29 (2005).
10. Calle Torres M. G.: Energy consumption in wireless sensor networks using GSP. Master of Science in Telecommunications. University of Pittsburgh . (2006). Thesis Director: Kabara J., http://d-scholarship.pitt.edu/7682/1/callemariag072606.pdf
11. Meyer auf der Heide F., Schindelhauer C., Volvert K. : Congestion, Energy and Delay in Radio Networks. Proceedings of the fourteenth annual ACM symposium on Parallel algorithms and architectures. Pages 230-237. (2002).
12. Hekmat R. : Ad-hoc Networks: Fundamental properties and Network topologies – Cap. 8 : Interference in Ad-hoc Networks – Pag. 77 a 94 – (2006). Holanda.
13.Ting-Chao Hou : Transmission Range Control in Multihop Packet Radio Networks. In: IEEE Transactions on communications Vol. Com 34 – Nª 1. (1986).
14. Pascal von Rickenbach, Roger Wattenhofer, and Aaron Zollinger.Algorithmic Models of Interference in Wireless Ad Hoc and Sensor Networks. In: IEEE/ACM Transactions on Networking, vol. 17, No. 1, (2009).
15.Asha T., Muniraj N. Network Connectivity based Topology Control for Mobile Ad Hoc Networks. In: International Journal of Computer Applications (0975 – 8887) Vol.56– No.2, (2012).
16. Khabbazian M., Durochez S., Haghnegahdaz A. Bounding interference in wireless ad hoc networks with nodes in random position. In: Networking, IEEE/ACM Transactions on issue 99.(2014).
17. Gagliardi O., Taramilla M., Berón M. Hernández Peñalver G.: La geometría computacional a nuestro alrededor. In: IV Workshop de Investigadores en Ciencias de la Computación. (2002).
18. Beròn M., Gagliardi O., Hernández Peñalver G. Estrategias de ruteo alternativas para redes móviles. In: XI Congreso Argentino de Ciencias de la Computación. (2005).
19. Clark B.N., Colbourn C.J., Johnson D. S.: Unit Disk Graphs. In: Discrete Mathematics 86 165-177. (1990).
20. Peñalver Hernandez G., http://www.dma.fi.upm.es/gregorio/grafos/prorout/grafos.htm
21. Ramanathan, R., Rosales-Hain, R.: Topology control of multihop wireless networks using transmit power adjustment. In: INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE. (2000).
22. Chen Z., Xu P., Deng X. : A Distributed Planar t-Spanner Topology Control Algorithm in Wireless Sensor Networks. Journal of Computer Research and Development. Vol. 49, No. 3. (2012).
23. Burkhart M., Von Rickenbach P., Wattenhofer R., Zollinge A.: Does Topology Control Reduce Interference? In: In Proceedings of ACM Mobi-Hoc 04 9–19.(2004).
24.Korman M.: Minimizing interference in ad hoc networks with bounded communication radius. In: Information processing Letters 112 (2012) 748-752.
25.Guan Q., Yu R., Jiang S., Leung V.: Topology control in mobile ad hoc networks with cooperative communications.In: IEEE Wireless Communications. (2012).

Uso de dispositivos móviles en zonas rurales aisladas abastecidas con energía solar fotovoltaica

Sergio H. Rocabado Moreno[1], Javier Díaz[2], Carlos Cadena[3]
[1]C.I.D.I.A. – Centro de Investigación y Desarrollo en Informática Aplicada (UNSa)
[2]L.IN.T.I. – Laboratorio de Investigación en Nuevas Tecnologías Informáticas (UNLP)
[3]INENCO – Instituto de Investigaciones en Energía No Convencional (UNSa - CONICET)
[1]srocabad@cidia.unsa.edu.ar, [2]jdiaz@unlp.edu.ar, [3]cadenacinenco@gmail.com

Resumen. En el presente trabajo se propone el uso de dispositivos móviles, con recarga basada en energía solar fotovoltaica, para acceder a información digital desde zonas rurales que se encuentren fuera del alcance de los centros de distribución de energía. Se realizaron mediciones de la potencia utilizada por un dispositivo móvil ubicado en una zona rural aislada, inicialmente se determinó el consumo de energía del celular sin carga de trabajo (e_0), luego se midió el consumo adicional de energía (e_1) para ejecutar cierto trabajo T_1 en el celular. A continuación, se calculó la energía que puede entregar un sistema fotovoltaico portátil (e_{SF}) en función de las características técnicas del panel y los valores de radiación solar de la zona. Finalmente, se verificó la franja horaria en la que el sistema fotovoltaico es capaz de satisfacer los requerimientos energéticos del dispositivo para ejecutar el trabajo T_1 ($e_{SF}>e_0+e_1$). La propuesta se puede replicar en zonas rurales con características similares, introduciendo nuevas cargas de trabajo en función de los requerimientos informáticos de los pobladores de la zona.

Palabras Clave: Dispositivos móviles, redes móviles, consumo de energía, zonas rurales, energía solar fotovoltaica.

1. Introducción

Las zonas rurales aisladas del país se caracterizan, entre otros aspectos, por su baja densidad demográfica, población dispersa, cobertura de red celular limitada y carencia de servicio de distribución de energía eléctrica. Los habitantes de estas zonas utilizan energías alternativas, como paneles solares y grupos electrógenos, para cubrir necesidades energéticas elementales. La región Noroeste de Argentina (NOA) posee numerosas zonas de este tipo, donde las posibilidades de acceso a la información digital son prácticamente nulas debido a que es muy difícil suministrar energía eléctrica a los equipos computacionales.

Por su bajo consumo energético, respecto de computadoras convencionales, los dispositivos móviles (celulares, smartphones, tablets y otros) constituyen una alternativa viable para este tipo de zonas.

En este trabajo se presentan los resultados parciales de una investigación que busca reducir el consumo de energía de los dispositivos móviles para facilitar su

utilización en zonas rurales aisladas, posibilitando el acceso a la información digital a pobladores de estas zonas sin necesidad de que tengan que trasladarse a centros urbanos. La propuesta busca solucionar las siguientes carencias de las poblaciones rurales aisladas: 1) Imposibilidad de acceso a información digital utilizando dispositivos móviles seguros y de bajo consumo energético y 2) Falta de equipamiento basado en energías renovables que garantice el uso continúo de dispositivos móviles y conexión permanente a la red celular.

1.2 Consumo de energía en dispositivos móviles

En los últimos años se produjo una importante evolución en los dispositivos móviles en general y en particular en los teléfonos celulares, los nuevos equipos llevan integrados una gran variedad de componentes como CPU, memoria, cámara, acelerómetro, pantalla LCD, GPS, micrófono, altavoces, interfases de comunicación 2G/3G/4G, Wi-Fi, Bluetooth, etc. Estos nuevos celulares se denominan teléfonos inteligentes o *smartphones*, y brindan a sus usuarios funcionalidades avanzadas, entre otras: navegar por Internet, reproducir audio y video, tomar fotografías, filmar videos y utilizar aplicaciones de posicionamiento satelital.

La velocidad a la que ha evolucionado la capacidad de las baterías no ha sido la misma a la que han evolucionado los teléfonos móviles. La gran mayoría de los *smartphones* utiliza baterías recargables electroquímicas, normalmente de iones de litio (Li-ion) o de polímeros de litio (Li-po). El problema de estas baterías es su corta duración cuando se utiliza el dispositivo móvil de forma continua por muchas horas, y en especial cuando se mantienen conexiones a Internet a través de la interfaz de radio (2G/3G/4G). Este inconveniente ha planteado que los propios fabricantes tengan que diseñar el hardware y el software de los dispositivos con premisas de eficiencia energética para reducir los niveles de energía consumidos por estos equipos. No obstante, si un dispositivo será utilizado en zonas donde las posibilidades de recarga son limitadas (Ej: Zonas rurales aisladas) será necesario implementar mecanismos que permitan administrar y optimizar el uso de la energía disponible.

Para realizar una administración eficiente de energía se requiere tener un conocimiento previo sobre dónde y cómo se consume la energía. Determinar qué componentes del dispositivo hacen que las baterías se descarguen rápidamente puede ayudar a desarrollar técnicas para reducir el consumo de energía. Sin embargo, obtener el consumo de cada uno de los componentes de un dispositivo móvil no es una tarea fácil de realizar.

Se llevaron a cabo estudios que centran su problemática en analizar el consumo de energía de un dispositivo móvil y proponen modelos matemáticos para estimar este consumo utilizando herramientas de software y/o hardware para medir el consumo de energía de los distintos componentes de hardware.

En [1] se estudia un escenario real de uso de Smartphone, utilizando la aplicación Powertutor para Android [2] para realizar las mediciones de consumo energético. Con base en los resultados se propone un Framework para modelar y optimizar el consumo de energía de aplicaciones en dispositivos móviles.

En [3] se presenta una infraestructura para medir (con elevada precisión) la energía que precisa un dispositivo móvil para exhibir páginas web. Con los resultados

obtenidos en las mediciones se efectúan recomendaciones concretas para diseñar páginas web minimizando la energía necesaria para mostrar los contenidos. Las mediciones se realizaron utilizando un sistema que mide directamente la corriente eléctrica que circula hacia el dispositivo, utilizando una combinación de hardware externo (multímetro y batería abierta) y software para automatizar las mediciones (Figura 2).

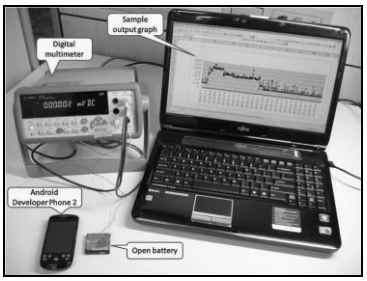


Fig. 2. Sistema de medición de alta precisión.

En [4] se efectúa un análisis detallado del consumo de energía de un Smartphone Samsung Galaxy SIII, las mediciones de potencia se realizaron utilizando un multímetro digital y conexiones a nivel circuito (Figura 3). Se ejecutaron una serie de pruebas de uso cotidiano de un celular, que incluyen: Reproducción de audio y video, juegos, llamadas telefónicas, SMS, envío de correo electrónico, navegación WEB, toma de fotografías y celular inactivo. Para cada prueba se midió el consumo de energía de los principales componente del Smartphone: CPU, RAM, pantalla, GPU, interfaces inalámbricas, cámara, GPS y sensores ambientales.

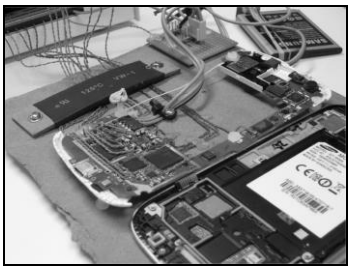


Fig. 3. Medición del consumo energético a nivel circuito

En [5] se presenta un estudio comparativo de la eficiencia energética en dispositivos y aplicaciones móviles, se utilizo el programa Trepn profiler [6] para medir el rendimiento y el consumo de energía de aplicaciones Android en dispositivos con procesadores SnapDragon desarrollados por la empresa Qualcomm. Trepn Profiler posibilita el acceso a la siguiente información: Porcentaje y frecuencia de uso de la CPU, consumo de energía, estadísticas de consumo de memoria física y virtual y utilización de la red celular y Wi-Fi. El consumo de energía se mide a través de sensores (sensor-based) incorporados en los procesadores SnapDragon, esto permite realizar mediciones con mayor precisión. Los sensores internos miden el

consumo de energía en cada núcleo del procesador y a partir de estos valores obtienen el consumo total de energía (Figura 4).

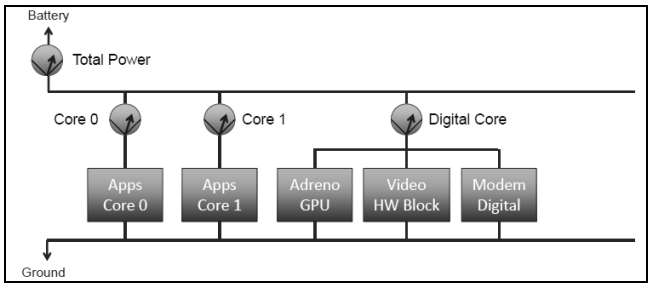


Fig. 4. Sensores incorporados en los procesadores Snapdragon.

También se efectuaron estudios desde el punto de vista del comportamiento humano, en [7] se presenta una investigación para determinar la forma en la que las personas cargan sus dispositivos móviles (patrones de carga). En el desarrollo del artículo se realiza una discusión de las implicancias de la conducta del usuario en la duración de la batería y el uso de la energía.

1.3 Energía solar fotovoltaica en zonas rurales

El recurso energético natural de la región del NOA (Noroeste Argentino) es la energía solar. La aridez del clima y la latitud tropical hacen que se cuente con una alta radiación solar la mayoría de los días del año. Éste es un recurso limpio y renovable que puede ser aprovechado en poblaciones rurales aisladas del NOA que se encuentran fuera del alcance de los centros de distribución de energía [8].

La tecnología fotovoltaica actualmente ya es competitiva para proveer de energía eléctrica a lugares alejados del Sistema Interconectado Nacional (SIN). Entre muchas de sus ventajas pueden mencionarse las siguientes:

- Se evita el costoso tendido y mantenimiento de las líneas eléctricas en zonas de difícil acceso.
- Contribuyen a evitar la emigración de pobladores de zonas aisladas.
- Es una energía descentralizada que puede ser captada y utilizada en cualquier lugar donde se localice la demanda.
- Una vez instalada tiene un costo energético nulo.
- Posee bajo costo de mantenimiento y riesgo de fallas muy bajo.
- Las instalaciones son modulares, se puede aumentar o reducir la potencia instalada fácilmente según las necesidades.

Si bien los sistemas fotovoltaicos actualmente son utilizados en muchas zonas aisladas, generalmente en comunidades rurales donde tuvo intervención el PERMER (Proyecto de Energías Renovables en Mercados Rurales) [9]; el aprovechamiento de la energía solar para la carga de dispositivos móviles no está lo suficientemente difundido. En este trabajo se introduce la aplicación de esta tecnología, contribuyendo a mejorar las posibilidades de comunicación de los pobladores de la región.

1.4 Cargadores solares para dispositivos móviles

En [10] y [11] se presenta el diseño de un cargador solar portátil para dispositivos móviles de bajo consumo energético, en el diseño se especifica el uso de un número reducido de componentes, lo que hace que el cargador sea económicamente viable además de ser altamente portátil.

En [12] se efectúa una revisión de las diferentes técnicas de adquisición de energía solar para alimentar dispositivos portátiles en forma limpia y sustentable. Se realiza un estudio para mejorar el rendimiento de una celda solar, mostrando algunos experimentos prácticos realizados en América del Sur. Además, se analizan algunas implementaciones de cargadores solares para dispositivos portátiles, mostrando los beneficios e inconvenientes de cada arquitectura. Finalmente, se propone una solución completa e integrada para el uso de energía solar fotovoltaica en la recarga de baterías de dispositivos portátiles.

2. Metodología utilizada

A continuación se enumeran los pasos de la metodología utilizada en este trabajo:

1. Medir el consumo de energía del celular sin carga de trabajo (e_0).
2. Seleccionar una tarea (T_1), la tarea en función de los requerimientos informáticos de los pobladores de la zona en zona rural (Ej.: Descarga de contenido Web).
3. Definir los niveles de rendimiento aceptables para la ejecución de la tarea (Ej.: Retardo, Throughput).
4. En función de la capacidad del celular seleccionar aplicaciones móviles para ejecutar la tarea (app_1, app_2,….., app_n).
5. Seleccionar una tecnología de comunicación según la disponibilidad en la zona (Ej.:GPRS/EDGE/UMTS/HSDPA).
6. Ejecutar la tarea utilizando las aplicaciones seleccionadas para medir el consumo de energía y el rendimiento de cada aplicación. Se realizan varias repeticiones en distintos horarios para contemplar diferentes niveles de carga de la red celular.
7. Seleccionar la aplicación con el menor consumo de energía y que se encuentre dentro de niveles aceptables de rendimiento, $e_1=\min(e_{app1}, e_{app2},….., e_{appn})$.
8. Calcular la energía total requerida $e = e_0 + e_1$
9. Efectuar un relevamiento de los niveles de radiación solar de la zona rural.
10. Estimar la energía que el sistema fotovoltaico puede entregar (e_{SF}) en función de valores de radiación solar de la zona rural y de las características técnicas del panel. La hora es una variable a considerar para realizar la estimación, debido a que la radiación solar es elevada cerca al mediodía y es reducida en las primeras horas de la mañana y últimas de la tarde.
11. Analizar los resultados obtenidos.
    Si el sistema fotovoltaico es capaz de entregar energía suficiente para realizar la tarea T_1 ($e_{SF} \geq e$), se debe determinar la franja horaria en la cual puede entregar la energía requerida. El remanente (e_{SF}-e) será utilizado para cargar la batería del celular.

Si el sistema fotovoltaico NO es capaz de entregar energía suficiente para realizar la tarea T_1 (e_{SF}<e), determinar el tiempo de carga necesario para transferir a la batería del celular la energía suficiente para ejecutar la tarea T_1.
Si los requerimientos energéticos del celular son muy superiores a los que el panel puede proporcionar, las mediciones efectuadas pueden ser utilizadas para ampliar el sistema fotovoltaico existente o dimensionar uno nuevo.

3. Escenario de estudio

En la Figura 5 se observa la representación gráfica del escenario implementado para realizar las pruebas y mediciones. En el mismo se conecta un Smartphone a un servidor a través de la red celular e Internet. El dispositivo móvil (Cliente) se comunica con el servidor utilizando un canal de comunicación TCP/IP extremo a extremo (end to end). La recarga del celular se efectúa utilizando energía solar a través de un panel fotovoltaico en forma directa, sin utilizar baterías externas, de esta manera: se puede medir sin pérdidas la potencia que puede entregar el panel, se evita la contaminación que producen las baterías externas y se eliminan las tareas de mantenimiento de las baterías, tareas que son costosas y difíciles de realizar en zonas aisladas.

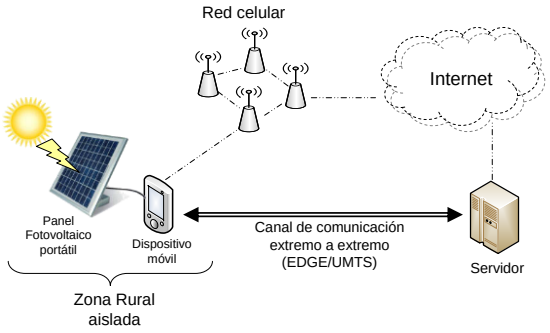


Fig. 5. Escenario de estudio.

3.1 Configuración del nodo cliente

La configuración del dispositivo móvil con el cual se realizaron las pruebas, es la siguiente:

Equipo:	Samsung I9300I Galaxy S III Neo
CPU:	Quad-core 1.4 GHz Cortex-A7
Chipset:	Qualcomm MSM8226 Snapdragon 400
RAM:	1024MB RAM.
SO:	Android OS ver. 4.4.4 (KitKat)
Root:	SI
Batería:	Lítio-íon, 2100 mAh, 3.7 v.

Este equipo fue especialmente preparado para minimizar el consumo de batería, se procedió entonces a: desinstalar las aplicaciones no indispensables para su funcionamiento, deshabilitar dispositivos de hardware no utilizados en las pruebas y habilitar el modo de bajo consumo.

3.2 Configuración del sistema fotovoltaico

La configuración del panel fotovoltaico utilizado durante las pruebas, es la siguiente:

Marca y modelo:	Solartec KS10T
Área útil del panel:	0,243 x 0,318 = 0,07727 mt2
Potencia nominal:	10W
Tensión máxima:	17,4V
Corriente máxima:	0,58ª
Radiación solar test:	1000 W x mt2

Potencia que recibe el panel = radiación solar x área útil del panel = 1000W x 0,07727 = 77,27 W
Eficiencia = potencia que entrega / potencia que recibe = 10W/77,27W = 0,1294 (12,94%)

3.3 Mediciones realizadas

A continuación se presenta la configuración utilizada para las mediciones:

Tarea (T_1):	Descarga de contenidos Web
Protocolo:	HTTP
Tecnologías de comunicación:	EDGE y WCDMA
Aplicación 1 (app_1):	Chrome
Aplicación 2 (app_2):	Mozilla Firefox
Franja horaria de pruebas:	8:00 a 19:00

El consumo de energía en el nodo cliente se midió utilizando la aplicación Trepn Profiler [6], esta herramienta utiliza sensores internos incorporados en los chipset Qualcomm para determinar con alta precisión el consumo de energía del sistema y de el consumo de una aplicación en particular. Trepn genera un archivo de texto con los resultados de cada medición (“log”), esto posibilita un análisis exhaustivo posterior a la medición [13].

En la figura 5 se presentan capturas de pantalla de la aplicación Trepn Profiler. A la izquierda se observa el consumo de energía del celular (Profile system) y a la derecha se visualiza el consumo de energía de la aplicación Chrome al realizar una descarga de contenido web (Profile app).

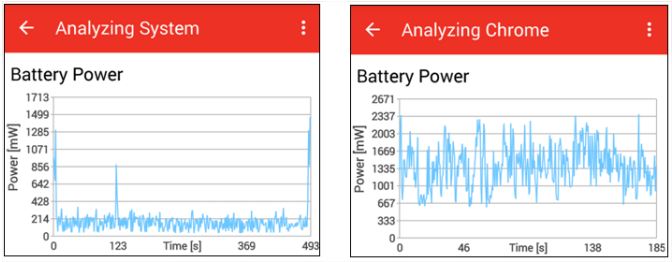


Fig. 5. Consumo de energía del celular sin ejecutar aplicaciones y consumo de energía de una aplicación.

Para determinar la energía entregada por el panel fotovoltaico, se utilizaron los valores de radiación del día 29/07/2015 de 8:00 a 19:00, los mismos se expresan en potencia por metro cuadrado (Watt x m^2) y fueron normalizados de acuerdo a la eficiencia del panel (12,94%). Ejemplo:

Hora 12:00, radiación solar = 663Wm^2
Potencia recibida = radiación solar x área útil del panel = 663Wm^2 x 0,07724 m^2= 51,21W
Potencia entregada = 51,21W x 0,1294 = 6,62 W

4. Resultados

A continuación se presentan algunos de los resultados obtenidos, utilizando gráficos que resumen los aspectos estudiados.

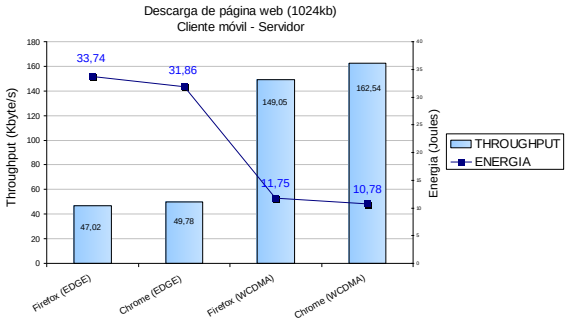


Fig. 6. Throughput y consumo de energía para tráfico HTTP generado por Chrome y Firefox

Los resultados presentados en la figura 6 evidencian que Chrome alcanza un rendimiento ligeramente superior al navegador Firefox. Se observa que el uso de la tecnología WCDMA en lugar de EDGE, reduce el gasto energético en un 33% (de 31,86 a 10,78 Joules) y triplica el throughput.

El consumo de energía se calculó en función de la potencia utilizada por la aplicación, en el caso de Chrome sobre WCDMA la potencia fue de 1,71W (en promedio), a este valor se le debe agregar la potencia mínima que requiere el celular para su funcionamiento 0,21W, totalizando 1,92W de potencia, que en una hora será equivalente a 1,92Wh.

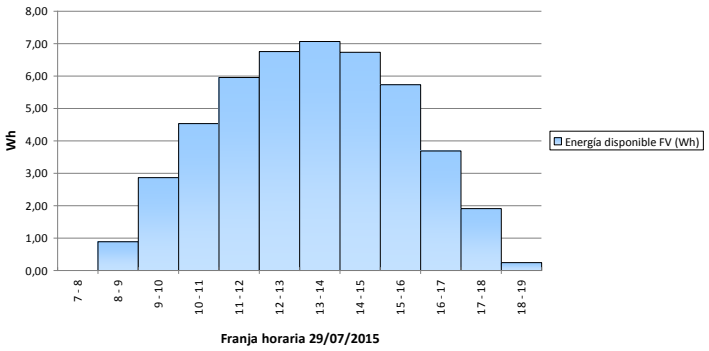


Fig. 7. Energía entregada por el panel fotovoltaico

En la figura 7 se presentan los valores entregados por el panel fotovoltaico de 7:00 a 18:00 el día 29/07. La energía entregada entre las 9:00 y 18:00 posibilitó la ejecución de las aplicaciones (Chrome/Firefox) y el remanente fue utilizado para cargar la batería del celular. Ejemplo: al ejecutar Chrome sobre WCDMA en el horario pico (13:00 a 14:00) se generó un remanente de (7Wh-1,92Wh)=5,08Wh de energía.

6. Conclusiones y trabajos futuros

La tecnología WCDMA ofrece un mejor rendimiento que EDGE y reduce el consumo de energía, lamentablemente no siempre esta disponible en zonas rurales aisladas, donde solamente se encuentran tecnologías EDGE y GPRS y en algunos casos es necesario instalar amplificadores de señal.

El remanente de energía que entrega el panel se puede utilizar para ejecutar otras tareas en función de los requerimientos del usuario de la zona rural: Acceso a correo electrónico, mensajería, posicionamiento asistido y otras.

Conociendo la potencia que entrega el panel y la potencia que requiere el celular para funcionar, se puede determinar el tiempo de carga de la batería del celular. Este valor puede ser importante para viajeros con limitaciones de tiempo y también en casos de ser necesario el uso en horarios de baja radiación solar.

Si la energía que entrega el panel esta muy por encima de la energía que requiere el celular, se pueden modificar las dimensiones del panel para que se ajuste a los requerimientos de energía del celular, un panel de menor tamaño será mas fácil de transportar (portabilidad) y de orientar para obtener mejor ganancia de radiación solar.

Para continuar con esta línea de investigación se tiene previsto la construcción de un cargador solar portátil que interactúe con el dispositivo móvil, posibilitando la optimización de la carga en función de los requerimientos de energía del dispositivo y la activación de mecanismos de ahorro de energía en el dispositivo cuando el cargador se encuentre en condiciones desfavorables para la adquisición de energía.

Referencias

1. Zhang, Lide. (2013). *Power, Performance Modeling and Optimization for Mobile System and Applications.* (Thesis - Doctor of Computer Science and Engineering), University of Michigan.
2. ZHANG, Lide; TIWANA, Birjodh; QIAN, Zhiyun and WANG, Zhaoguang. (2010). *Accurate online power estimation and automatic battery behavior based power model generation for smartphones*. Paper presented at the 2010 IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), Scottsdale, AZ. http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5751489
3. Thiagarajany, Narendran; Aggarwal, Gaurav; Nicoara, Angela (2012, April). *Who Killed My Battery: Analyzing Mobile Browser Energy Consumption*. Paper presented at the WWW 2012 – Session: Mobile Web Performance, Lyon, France.
4. Aaron, Carroll; Gernot, Heiser. (2013, July). *The Systems Hacker's Guide to the Galaxy Energy Usage in a Modern Smartphone.* Paper presented at the 4th Asia-Pacific Workshop on Systems (APSys'13), Singapore.
5. Metri, Grace; Shi, Weisong; Brockmeyer Monica (2015, February). *EnergyEfficiency Comparison of Mobile Platforms and Applications: A Quantitative Approach*. Paper presented at the HotMobile Santa Fe, NM, USA.
6. Qualcomm Technologies. (2014). Trepn Profiler. from https://developer.qualcomm.com/mobile-development/increase-app-performance/trepn-profiler
7. Ferreira, Denzil;Dey, Anind;Kostakos, Vassilis. (2011). *Understanding Human-Smartphone Concerns: A Study of Battery Life.* Paper presented at the Pervasive'11 Proceedings of the 9th international conference on Pervasive computing.
8. Ottavianelli, Emilce; Ibarra Marcelo; Cadena Carlos. (2013, Noviembre). *Uso de indicadores sociales en estudios de factibilidad de instalación de sistemas solares para generación de electricidad en localidades rurales*. Paper presented at the XX Simposio peruano de Energía solar, Tacna – Perú.
9. Secretaria de Energia - Ministerio de Planificación. (2015). PERMER, Proyecto de Energías Renovables en Mercados Rurales. from https://www.se.gob.ar/permer/
10. Attia, Y. I. Al-Mashhadany; H. A. (2014, April). Novel Design and Implementation of Portable Charger through Low-Power PV Energy System. *Advanced Materials Research, 925*, pp. 495-499 (ISSN 1662-8985).
11. Attia, Yousif I. Al-Mashhadany; Hussain A. (2014, November 12). *High performance for real portable charger through lowpower PV system.* Paper presented at the International Journal of Sustainable and Green Energy.
12. Távora, Filipe; Maia, Andrea Sarmento. (2012, june). *Solar Battery charger for portable devices application*. SiliconReef Consultoria, Pesquisa e Projetos em Tecnologia da Informação. Retrieved from www.siliconreef.com.br
13. Qualcomm Technologies. (2013). *When mobile apps use too much power: A Developer Guide for Android App Performance*. Retrieved from https://developer.qualcomm.com/qfile/27292/trepn-whitepaper-apps-power.pdf

Análisis de despliegue de una IaaS utilizando Openstack

Brian Galarza[1], Gonzalo Zaccardi[1], Diego Encinas[1,2], Martín Morales[1]

[1] Instituto de Ingeniería y Agronomía - Universidad Nacional Arturo Jauretche
[2] Instituto de Investigación en Informática LIDI – Facultad de Informática – Universidad Nacional de La Plata

briangalarza@hotmail.com, pcg.computers@hotmail.com, dencinas@unaj.edu.ar, martin.morales@unaj.edu.ar

Resumen. El presente trabajo tiene como objetivo realizar un despliegue de una Infraestructure as a Service (IaaS). Para ello se utilizará software de código abierto (Openstack). Se demostrará un escenario con la arquitectura Legacy Networking (nova-network) en la cual se tendrán 2 nodos (controller y compute) para poder lanzar instancias virtualizadas. Se mostrarán los resultados obtenidos de la implementación realizada para, finalmente, concluir con un análisis de las ventajas que conllevarían la implementación de la misma en un ámbito universitario.

Palabras claves: Cloud-Computing, IaaS, Openstack, Node, Networking, Hypervisor.

1 Introducción

Cloud Computing es un paradigma que está en constante crecimiento durante estos últimos años, cada vez más compañías y grupos de investigación trabajan en conjunto con el fin de explotar las oportunidades ofrecidas por el mismo. Dicho paradigma ofrece muchas ventajas, tales como el bajo costo de implementación, ya que no se necesitan computadoras de última tecnología debido a que éstas trabajan conjuntamente (Clustering) con la posibilidad de escalar horizontalmente de manera sencilla. Además, hay software Open Source disponible para los nodos en el clúster como las infraestructuras Eucalyptus, OpenNebula, CloudStack u OpenStack integradas con GNU/Linux y compatibles, por ejemplo, con Amazon WebServices.
Como objetivos principales de esta investigación y aplicación práctica/técnica se proponen:

- Mostrar las ventajas y desventajas de OpenStack como sistema de administración de nube (Cloud Computing) a partir de la infraestructura básica (un nodo controller y, al menos, un nodo compute).
- Proponer una infraestructura de red propia e independiente en el ámbito universitario.

1.1 Cloud Computing

Cloud computing [1] es un paradigma de computación en franco avance y tiene muchos servicios que aún no han sido desarrollados. La capacidad de ofrecer servicios a usuarios de manera sencilla y económica propendió que empresas como Google, Facebook, Outlook, Yahoo, Hotmail, etc. utilizaran sus recursos en hardware y software para desarrollar "nubes" que almacenan todos los datos de los usuarios ofreciendo distintos tipos de servicios:
IaaS (Infraestructure as a Service): Se contrata capacidad de proceso (CPU) y almacenamiento. En este caso sólo se paga por lo utilizado. Ejemplos: EC2 de Amazon y Azure de Microsoft.
PaaS (Platform as a Service): Se proporciona almacenamiento, un servidor de aplicaciones (en donde se ejecutarán las aplicaciones requeridas) y una base de datos. Ejemplo: Google App Engine.
SaaS (Software as a Service): Es una aplicación para el usuario final en donde se paga un alquiler por el uso de software. Muchas de estas aplicaciones son actualmente gratuitas. Ejemplos: Google Docs, Office 365, Dropbox.
HaaS (Hardware as a Service): Consiste en ofrecer hardware de manera que sea utilizada a través de internet logrando que el sistema utilice los recursos distribuidos geográficamente como si estuvieran localmente. Ofreciendo una mayor eficiencia de los mismos.
A su vez cada uno de ellos puede ser:

- Público: Una empresa ofrece IaaS a terceros, encargándose de toda la gestión del Cloud. El caso más conocido es Amazon Elastic Compute Cloud (EC2).
- Privado: Una organización configura sus propios recursos como IaaS para tener más flexibilidad y control total sobre sus recursos.
- Híbrido: Algunos servicios se gestionan en el cloud privado y otros se transfieren a uno público, normalmente utilizan una API común que permita una buena integración.

1.2 OpenStack

OpenStack [2] [3] es un software de código abierto que permite la implementación de una Infraestructure as a Service (IaaS) a través de múltiples servicios que, de manera coordinada, cumplen diferentes propósitos para lograr el correcto funcionamiento de dicha infraestructura. Cada servicio ofrece una interfaz de programación de aplicación (API) que facilita la integración de las mismas. Dependiendo de las necesidades de cada usuario, puede instalar algunos o todos los servicios disponibles. Estos servicios disponibles se describen en la Figura 1.

Servicio	Nombre del proyecto	Descripción
Dashboard	Horizon	Proporciona una interfaz gráfica web basado en los servicios subadyacentes de OpenStack permitiendo el lanzamiento de una instancia, la asignación de direcciones IP y la configuración de los controles de acceso, etc.
Compute	Nova	Gestiona el ciclo de vida de las instancias de proceso en un ambiente de OpenStack. Las responsabilidades incluyen la descomposición, programación y el cierre definitivo de las máquinas en la demanda.
Networking	Neutron	Permite la conectividad de la red para uno o varios servicios de OpenStack, como OpenStack Compute. Proporciona una API para que los usuarios definan las redes y los archivos adjuntos en ellos. Tiene una arquitectura conectable que soporta varios proveedores y tecnologías de redes populares.
Almacenamiento		
Object Storage	Swift	Almacena y recupera los objetos de datos no estructurados de su elección mediante una REST, es una API basada en HTTP. Es muy tolerante a fallos con su réplica de datos y su posible escalabilidad. Su puesta en práctica no es como un servidor de archivos con los directorios montables.
Block Storage	Cinder	Proporciona almacenamiento persistente en bloques para las instancias en ejecución. Soporta dispositivos plug and play y permite gestionar, por lo tanto, todos aquellos dispositivos de almacenamiento en bloques.
Servicios compartidos		
Identity Service	Keystone	Proporciona un servicio de autenticación y autorización para otros servicios de OpenStack. Proporciona también un catálogo de endpoints para todos los servicios de OpenStack.
Image Service	Glance	Permite almacenar y recuperar imágenes de disco de máquinas virtuales. OpenStack Compute hace uso de esta instancia durante la instancia de provisión.
Telemetry	Ceilometer	Monitorea la nuve de OpenStack para la evaluación comparativa, la escalabilidad y con fines estadísticos.
Servicios alto nivel		
Orchestration	Heat	Organiza múltiples aplicaciones en la nube de material compuesto utilizando ya sea el formato de la plantilla HOT nativo o el formato de la plantilla AWS, tanto a través de una API REST nativa de OpenStack nativo y una forma de nube compatible con la API de querys.

Figura 1. Servicios de OpenStack.

La arquitectura conceptual de OpenStack se puede ver en la Figura 2:

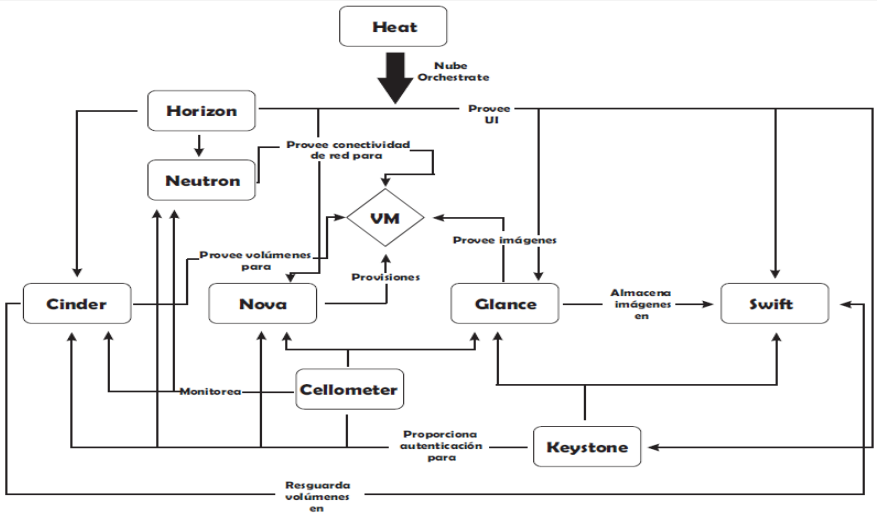


Figura 2. Arquitectura conceptual de OpenStack.

Varias compañías se han unido al proyecto, entre ellas, Red Hat, AT&T, Canonical, Cisco, Dell, Go Daddy, HP, IBM, Intel, Rackspace Hosting, Nexenta, AMD, Suse, VmWare, Oracle, Yahoo, entre otras.

2 Escenario

Para la implementación de la nube, se ha trabajado con dos equipos. Cada uno con una máquina virtual ejecutando Debian 8.1 [4] y en donde se instalaron los paquetes necesarios para que la infraestructura funcione correctamente.
Al utilizar la arquitectura legacy se tienen como mínimo dos nodos. Un nodo controller y un nodo compute.
Los servicios instalados en cada uno de los nodos son los indicados en la Figura 3.

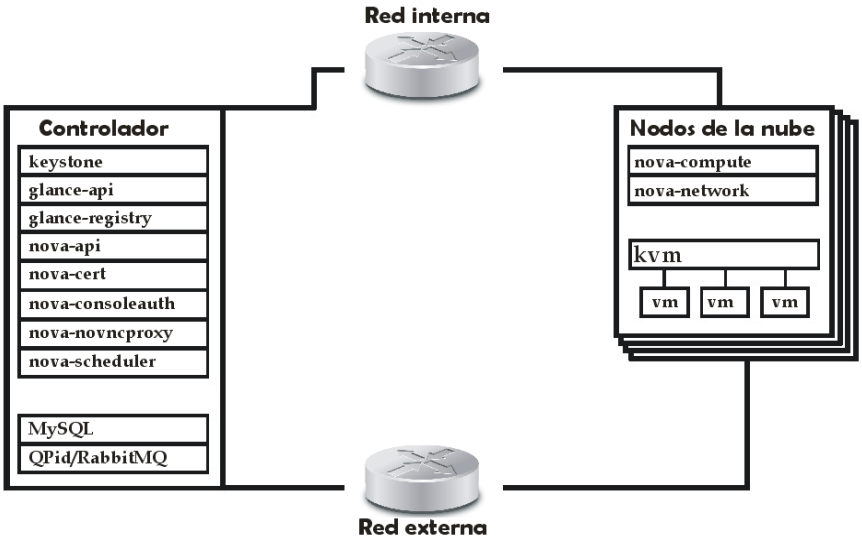


Figura 3. Legacy Networking.

El hypervisor seleccionado para el nodo compute depende del equipo utilizado, en caso de tener un equipo que soporte virtualización de tipo 1, se puede utilizar KVM o XEN, etc. En caso de no soportar, se puede utilizar QEMU [5].
Los dos equipos deben tener dos interfaces de red como mínimo. Una para la red externa y otra para la red interna. La red interna será utilizada para cuando se arme la red virtual legacy en la que se asigna un espacio de direcciones a los nodos que serán instanciados.
Los dos nodos se ejecutaron sobre máquinas virtuales utilizando Oracle virtualbox [6]. El utilizar máquinas virtuales reduce el rendimiento del nodo, sin embargo ofrece

las ventajas de tener todo virtualizado y poder realizar snapshots periódicamente lo cual permite retroceder fácilmente en caso de inconvenientes.
En las siguientes figuras (Figura 4 y Figura 5) se pueden observar cómo queda el esquema de los nodos.

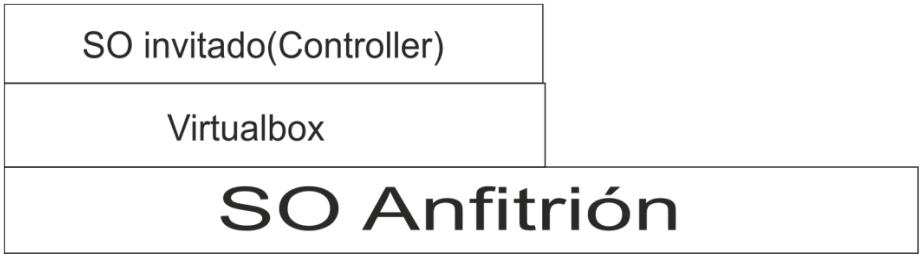


Figura 4. Controller node.

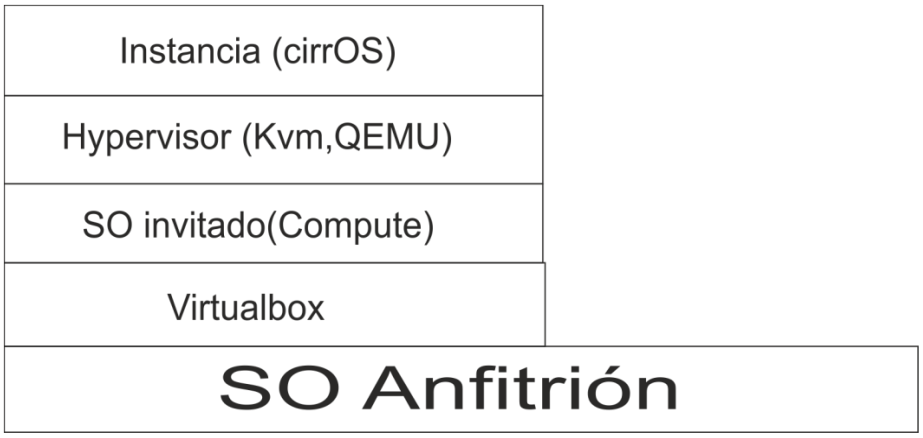


Figura 5. Compute node.

Openstack recomienda utilizar cirrOS [7] para entornos de prueba, sin embargo, es posible usar una imagen de cualquier distribución para el lanzamiento de instancias. Además es posible utilizar algún SO paravirtualizado [8] lo que optimizaría el rendimiento de las instancias lanzadas.

3 Resultados obtenidos

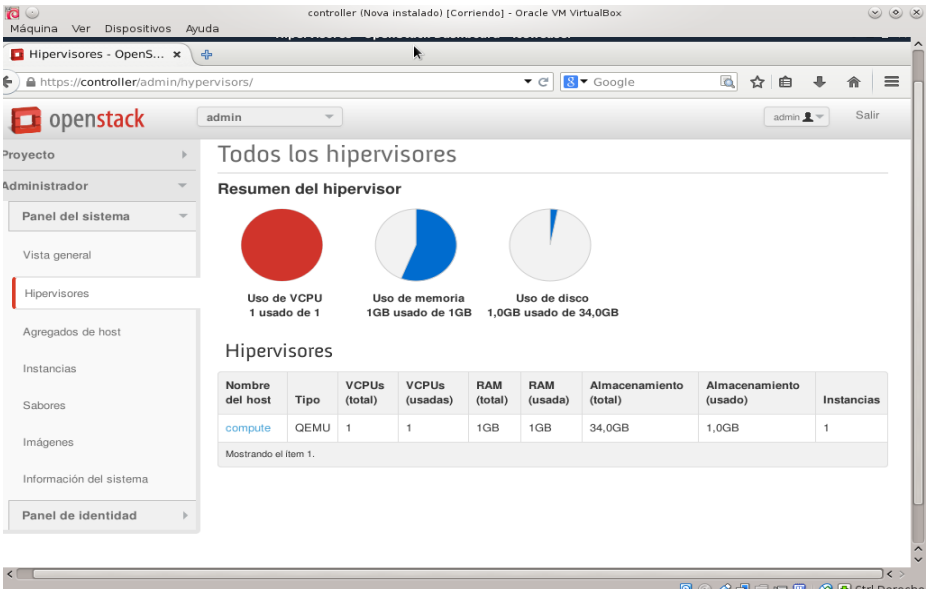


Figura 6. Hipervisor con una instancia lanzada.

En la Figura 6 puede observarse la interfaz que ofrece Openstack llamado Horizon en la cual se accede con el usuario y contraseña elegidos en la instalación y con los permisos de administración para poder realizar todas las operaciones. En ella se puede realizar todas las operaciones principales y observar todos los servicios instalados, nodos e instancias lanzadas. La principal ventaja que ofrece Dashboard consiste en la facilidad en usarlo para los usuarios sin elevado conocimiento en Linux, ya que la interfaz es muy intuitiva y ofrece todas las posibilidades que se pueden hacer por terminal.

Al dashboard se accede por medio de la dirección https://controller por lo que si todos los equipos de la red tienen configurada la resolución del host, pueden acceder remotamente [9]. También es posible configurar para acceder con la dirección IP del nodo controller, lo que permitiría acceder desde una red externa, sin embargo no es recomendable por razones de seguridad.

En la Figura 7 se puede observar como los servicios instalados en el controller se encuentran funcionando en el mismo (dirección IP usada 192.168.0.105)

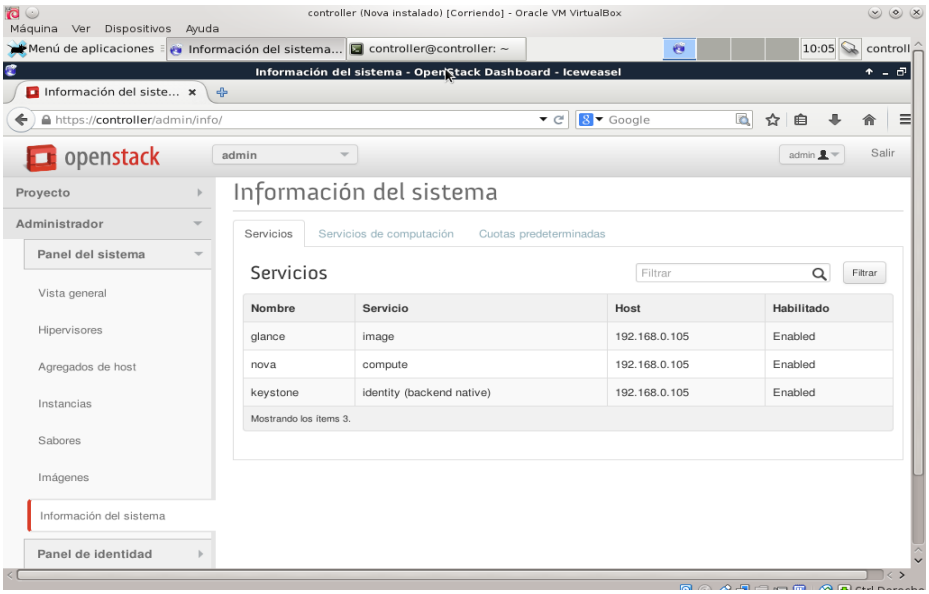


Figura 7. Servicios instalados y habilitados correctamente.

Estos servicios son los que necesita el nodo controller para poder funcionar correctamente. Al momento de lanzar una instancia se deben tener en cuenta la existencia de los sabores y de crear una red virtual. La red virtual es creada por el servicio Nova el cual otorgará direcciones IP flotantes a las instancias lanzadas y que ofrece configurar grupos de seguridad con reglas como el tipo de conexión (TCP, UDP) abrir un determinado puerto o un rango de ellos como también restringir las direcciones IPS con las que se permiten el tráfico, los nodos de un grupo de seguridad se podrán comunicar entre ellos y con otros grupos de seguridad (hay que indicarlo en las reglas de los mismos). Los sabores son configuraciones de los recursos que se tomarán al lanzar la instancia, estos son configurables por lo que es posible disponer de varias configuraciones según las necesidades de cada caso. Openstack ofrece algunos sabores predeterminados como el m1.tiny que cuenta con 1 VCPU, 512 MB RAM y 1 GB de disco como también el m1.xlarge que cuenta con 8 VCPUS, 16384 MB RAM y 160 GB de disco. En la Figura 8 se pueden observar los distintos sabores.

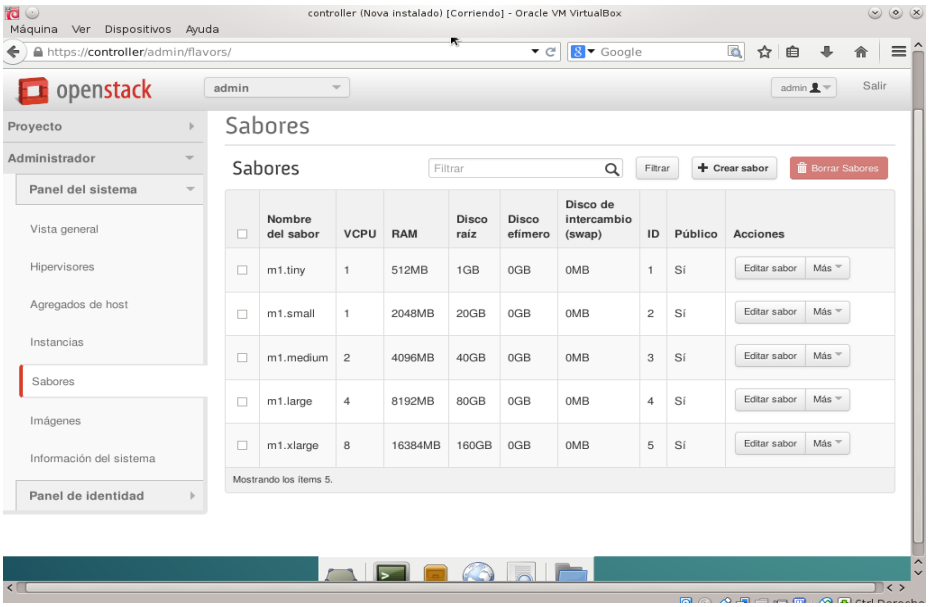


Figura 8. Sabores de openstack.

La principal problemática encontrada consiste en que los lanzamientos de Debian 8 y Openstack Kilo fueron muy cercanos (la salida de Debian 8 fue 5 días antes de la salida de Kilo), esto generó algunos problemas en los paquetes de instalación que se encuentran en los repositorios de Debian Jessie, además se remarca que aún no existe documentación sobre la instalación en Debian 8 por lo que es recomendable seguir con otra distribución de las que aparecen en la documentación (Ubuntu, CentOS, Fedora, etc.) o esperar a que Openstack se establezca en la nueva versión de Debian lanzada recientemente.

Conclusiones

Si bien OpenStack no es sencillo de implementar y hay que tener conocimientos básicos de redes una vez iniciada la expansión de la nube o clúster, es muy sencillo escalarlo. Se puede decir que puede hacerse de manera exponencial, ya que es un sistema muy modulado lo que permite tanto expandirse como reducirse de manera rápida y segura, una vez implementado correctamente los cimientos del sistema.
Finalmente, a partir de lo que se ha mostrado en el presente trabajo, puede decirse que OpenStack ofrece una gran variedad de servicios, los cuales, llevados a una buena práctica se traducen a un sin fin de posibilidades.

En línea con lo trabajado y expuesto, se busca:
Implementación de un IaaS encargado de realizar operaciones en procesamiento paralelo aumentando la eficiencia y reduciendo los costes generados.
Implementación de OpenStack Dashboard y de un sistema propio desarrollado para poder controlar/administrar de manera visual (WEB) y más básica cada uno de los servicios.
Implementación de una nube académica en la cual se puedan ofrecer distintos servicios personalizados y atender a las necesidades de cómputo y procesamiento de los distintos departamentos de la universidad.

Por último, cabe destacar que Openstack es una herramienta con una comunidad que periódicamente realizan actualizaciones con el fin de ofrecer mejores servicios y más variantes para poder adaptarse.

Referencias

1. Galarza, B.; Tuamá, C.; Zaccardi, G.; Encinas, D.; Morales, M. "Implementaciones de Cloud Computing y aplicaciones en el ámbito universitario". I Congreso Nacional de Ingeniería Informática y Sistemas de Información (CoNaIISI 2013). Ciudad de Córdoba, Argentina.
2. Encinas, D; Kunysz, E; Galarza, B; Zaccardi, G; Morales, M. "Performance de arquitecturas multiprocesador: técnicas de modelado y simulación, plataformas reconfigurables y cloud computing." XVII Workshop de Investigadores en Ciencias de la Computación (WICC 2015). Ciudad de Salta, Argentina.
3. OpenStack Cloud Software: Open source software for building private and public clouds. http://www.openstack.org. Agosto 2015.
4. Debian. The universal operating system. https://www.debian.org Agosto 2015
5. QEMU. Open source processor emulator. http://wiki.qemu.org/Main_Page. Agosto 2015
6. Oracle Virtualbox. https://www.virtualbox.org/. Agosto 2015
7. CirrOS. Tiny OS that specializes in running on a cloud. https://launchpad.net/cirros. Agosto 2015
8. Tanenbaum, Andrew S. 3rd Edition. Modern Operating Systems. Pearson
9. Kurose, James F; Ross, Keith W. 5th Edition. Computer Networking. Pearson

VII Workshop Innovación en Sistemas de Software (WISS)

7435 | Tumaco: Collaboration of Mobile Devices in Hypermedia-based In-Vehicle Infotainment Systems
Christian Grèvisse, Steffen Rothkugel

7499 | Prototipo móvil 3D para el aprendizaje de algoritmos básicos
Federico Cristina (UNLP), Sebastián Dapoto (UNLP), Pablo Thomas (UNLP), Patricia Pesado (UNLP)

7525 | Desarrollo de Herramientas para Warehousing en el Municipio de Junín
Ado Mariana (UNNOBA),
María Cecilia Rastelli (UNNOBA),
Ana Smail (UNCPBA), Rodolfo Bertone (UNLP)

7599 | istema de interacción con la ECU de un automóvil empleando procesamiento de lenguaje natural
Martín Jorge Agüero (UP), Maximiliano Colman (UP), Fabrice Dubois (UP), Marcelo Rabadán (UP), Santiago Amarila (UP)

7639 | EnMa Tool: una herramienta de software en apoyo al estándar ISO 50001 con basamento en Indicadores Clave de Desempeño para la Búsqueda de Eficiencia Energética en Procesos Productivos de PyMEs
Leopoldo Nahuel (UTN FRLP), José Maccarone (UTN FRLP), Javier Marchesini (UTN FRLP), Marcela Rognoni (UTN FRLP), Gastón Andres (UTN FRLP), Rodrigo María García (UTN FRLP)

7641 | WhatsINFO: a Gamified Mobile Application Promoting the Integration of Computer Science School Freshmen to the UNLP
Isabel Miyuki Kimura (UNLP) Matías Brown (UNLP), Claudia Queiruga (UNLP), Javier Díaz (UNLP)

Tumaco: Collaboration of Mobile Devices in Hypermedia-based In-Vehicle Infotainment Systems

Christian Grévisse and Steffen Rothkugel

Computer Science and Communications Research Unit
University of Luxembourg
6, rue Richard Coudenhove-Kalergi L-1359 Luxembourg
christian.grevisse.001@student.uni.lu, steffen.rothkugel@uni.lu

Abstract. In-Vehicle Infotainment systems nowadays are largely limited to one-to-one relationships with the drivers' mobile phone. In this paper, we propose Tumaco, a SOA-based IVI system, which enables the seamless integration of all devices of all passengers. Collaborative services can be executed within a cross-platform ecosystem through the use of semantic hypermedia. In addition, passengers may customize their own user experience in order to reduce mental workload. Our prototypical implementation showed the benefits of opening the user experience to all passengers on a car. Finally, we present a framework that enables developers to provide their own services within our ecosystem.

Keywords: In-Vehicle Infotainment Systems; Collaborative Computing; Customization; Hypermedia

1 Introduction

Today's In-Vehicle Infotainment (IVI) systems are often centered around a one-to-one relationship between the car's on-board unit and a single smartphone. Even upcoming industry products like Apple CarPlay or Android Auto focus on, partly cable-based, connections where the corresponding IVI UI is shown on screen, but most application logic runs on the single connected phone. There are two major issues with this approach. First, an IVI system should not just be a unidirectional graphical data stream ported from the smartphone to a car's screen [14]. Second, the mentioned one-to-one mapping does not allow a collaboration between data and services of all brought-in mobile devices of all passengers [4].

In addition, the user experience (UX) is often focused on the driver, although other passengers might also feel the need to customize their very own IVI UX. Corresponding to the philosophy of Marc Weiser, the father of Ubiquitous Computing, computing devices should evolve from being personal items to environmental ones [6]. In the IVI domain, this may only be achieved if it is possible to easily move user preferences (e.g. the design of a digital speedometer gauge)

from one car to another. This might come very handy in a car-sharing context, enabling a coherent, uniform UX. A UX known to the user avoids mental overload in opposition to sharing cognitive resources between the primary driving task and secondary tasks related to the adaptation to an unknown UX [11,3].

An IVI system communicating with only a single mobile device is probably not inline with Weiser's vision: He stated that the *real power of the concept comes not from any one of these devices – it emerges from the interaction of all of them* [20]. A heterogeneous peer-to-peer architecture introduces challenges because of the diversity of different operating systems and hardware requirements. Nevertheless, such a setup might result to be beneficial. Imagine the passengers aboard a car want to listen to some music. It would be desirable to determine, based on the music libraries of all present mobile devices, the most represented common music genre. Based on this genre, the different devices could, one at a time, be instructed to play a song and share the song's meta-information (e.g. artist, album cover, lyrics) on the infotainment screen(s). But maybe the device which is playing the current song has neither the corresponding album cover nor a data plan to download it, whereas another device might offer such a service and could help out.

In this paper, we propose Tumaco, a SOA-based IVI system with semantic hypermedia support, allowing collaboration of services among connected mobile devices. Tumaco is realised as an extension on top of existing mobile platforms, such as iOS and Android, seamlessly integrating with the genuine system. Our IVI ecosystem considers all passengers aboard a car by allowing them to customize their UX through the use of Web Components. In addition, we introduce `TumacoKit`, a framework which enables application developers to extend the Tumaco platform, without requiring architectural modifications. A prototypical implementation of Tumaco has been realised.

The remainder of this paper is organized as follows. In section 2, we summarize related work from different domains. In section 3, we present the major concepts of our approach and show their role within our introductive example. In section 4, we list some details about our prototypical implementation. We conclude and suggest some work for the future in section 5.

2 Related Work

The GENIVI Alliance has been working on an open source development platform for IVI systems based on Linux. Existing open source software components, not necessarily from the automotive domain, get adapted and integrated in order to reuse *non-differentiating* middleware features [1]. On the HMI side, the Tizen operating system has created a GENIVI-compliant IVI profile.

The support for a seamless integration of mobile devices in the IVI user experience recently has grown, as smart phones and tablets may actually provide services and data to improve safety and security rather than distract the driver, if they are properly integrated [2,10]. The exchange of user preferences, settings and functionalities between mobile devices and the IVI [4,3,5] as well as the use of

automotive data from the car together with the intelligence of connected mobile devices [14] can be advantageous. Rich interaction protocols may facilitate these interactions [13].

Many recent approaches are based on a service-oriented architecture. Jaguar Land Rover exchanges JSON-RPC messages between mobile devices and the IVI system to remotely regulate HVAC control [8]. Sonnenberg integrates mobile devices and the IVI system using web services [17,18,19], whose advantage is their independence of any platform or programming language. The Device Profile for Web Services (DPWS) standard allows to address services on the network [7].

An API with general media types needs additional knowledge to understand the semantics of a message [12]. However, creating a new media type for each new appliance would lead to a media type explosion [16], making it difficult to reach a consensus. There would be a lot of *superficially similar [...] but completely incompatible APIs*. JSON-LD allows self-descriptive, hypermedia-based machine-to-machine communication for truly RESTful web services by adding semantic annotations to regular JSON documents.

With Web Components, Google recently introduced reusable, encapsulated UI components into HTML5 [9]. They enable both component and data reusability, as they can be fed with JSON-LD data. Semantically enriched user interfaces, in contrast to ambiguous, locally understood UI descriptions based on pure XML or HTML, are reusable, allow cross-application interaction and ensure separation of concerns [15].

3 Tumaco

Inspired by existing SOA-based approaches, we further extended them by introducing semantic hypermedia support in order to allow a seamless integration of mobile devices in an IVI environment. Our goal is to enable collaboration while allowing *all* passengers aboard a car (including the driver) to customize their user experience based on their brought-in preferences. Linked Data is at the heart of our IVI architecture. Our approach should also allow the ecosystem to be extensible by application developers rather than by platform developers or OEMs.

An important assumption in our approach is that the car does *not* need to be directly connected to any cloud services through GSM, 3G or LTE networks. This way, network infrastructures will not need to cope with higher resource usage or overloads, as the connected mobile devices, working as relays between a car's IVI system and cloud services, can reduce the number of sent requests. Finally, the main business logic of services may stay at the side of the different connected mobile devices, which in general come with powerful hardware in comparison to embedded devices like IVI systems.

As a proof of concept, we have considered several use cases, such as showing the location of appointments saved on brought-in mobile devices on the IVI's map, configuring the layout of the car's digital speedometer with the user's preferences stored on his smartphone, playing music based on the most common

represented genre on the passengers' devices, as well as showing weather information on provided navigation points. These different use cases are representative for our three main pillars, namely collaboration of services, customization of the IVI UX and system extensibility through third-party apps.

3.1 Layered Architecture

In figure 1, we present the layered architecture of our approach. The Middleware layer, both at the IVI system and the mobile devices, includes RESTful servers and clients. The Service layer maps advertised services to their respective, platform-specific business logic. While the UI layer at the connected mobile devices is not a primary focus of our study, the UI layer of the IVI composes the user interface based on the user's preferences, the transmitted data and the available Web Components. The transmitted data (e.g. service descriptions, service requests, service input/output data) is serialized in JSON-LD, including semantic annotations to make the messages self-descriptive. This serialization binds the collaborative aspect, provided through the SOA, with the customization aspect, as Web Components may be fed with JSON-LD documents.

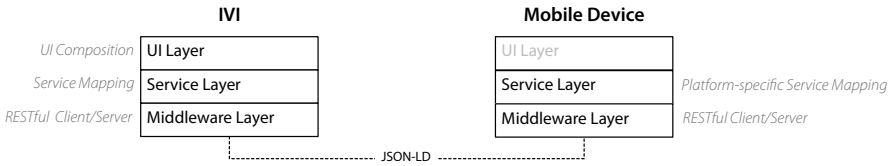


Fig. 1. Layered Architecture

3.2 SOA-based IVI Ecosystem

Both the IVI system and the connected mobile devices maintain a Service Directory within their respective Service layer. While mobile devices only hold their own service descriptions and stubs to IVI services in their Service Directory, the IVI itself knows its own services and the services from all connected mobile devices. When a mobile device registers with the IVI through the Tumaco platform app, it advertises its own service descriptions, serialized in JSON-LD, to the IVI and inquires latter's. Service descriptions include input and output types. They optionally include a UI component type, if a service performs a visual presentation. The word "type" here has to be understood as a semantic type (e.g. `Place`), not a programming language type.

In figure 2, we show a detailed view of the IVI's Service layer. The figure shows 3 collections, which are essentially database tables, containing service descriptions, service requests and data. Two further components, the *Request*

Dispatcher and the *Data Analyser* are responsible for actions concerning the service requests among the available services.

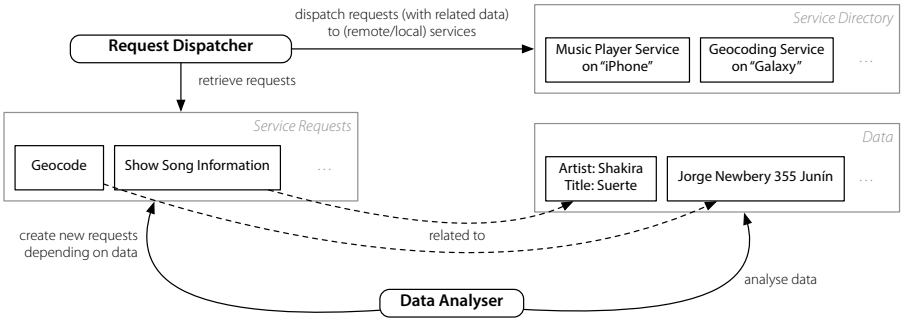


Fig. 2. IVI Service Layer

A service request may either come from a connected mobile device or from the IVI itself. The weak relation between service requests and their related data allows reusability of data items among different services. For instance, the information of the currently playing song, chosen by one connected smart device, may be reused by the IVI to ask another connected device to retrieve the corresponding album cover. Preference data, such as the background color of the speedometer, may also be pushed without a service request and directly used by the responsible Web Component. Service requests, once dispatched and the corresponding service executed, may provoke either another service request to be created (engaging a collaborative workflow) or data to be pushed.

The Data Analyser component checks, given a set of service-specific rules, whether data related to a service request is complete. For instance, the location of an appointment needs to include geocoordinates in order to be shown on the IVI's map. If the data is not complete, the Data Analyser will create a new service request to complete the data. In the previous example, this would mean that a service request for geocoding is created. Otherwise, the Request Dispatcher component executes the corresponding service request. Here, the map is reloaded to show all available appointments. If a service request maps to an IVI-own service, the IVI will execute the service's business logic. Otherwise, if the concerned service is offered by a connected mobile device, the service request is remotely dispatched through the underlying RESTful Middleware. Services, both at the IVI and on mobile devices, are defined in a descriptive, generic way, which means that the architecture or the platform app do not need to be altered in order to add new services.

Services offered by mobile devices can essentially be grouped into two categories. *Public Services* are altruistic mechanisms which reactively respond to a service request by (optionally) taking some input and returning some output. For

instance, the geocoding service may take an appointment location and return the corresponding geocoordinates. *Data Providers* proactively provide device-specific data (e.g. music library) if a matching service is offered by the IVI. In contrast to the altruistic nature of Public Services, Data Providers should be treated while respecting the user's privacy. The user should eventually be able to determine which data he really wants to share with the IVI.

Finally, we also wanted to open the ecosystem in a way that third-party apps may offer their services. This is achieved through Dynamic Services. While the platform app already offers a set of Public Services (e.g. geocoding) and Data Providers (e.g. music library), other apps may offer their service to the platform app and thus indirectly to the IVI (e.g. retrieve the weather of a location, provide navigation points). Both Public Services and Data Providers can be offered by other apps. At the IVI side, this separation is not visible, as the dispatching of any remote service is abstracted from the service execution. In order to offer services, an app needs to register them at the TUMACO platform app. This way, the platform app may advertise these dynamic services to the IVI system, which may later use them the same way as those directly offered by the platform app. Incoming service requests for dynamic services are relayed to the offering app. The inter-app communication process is platform-specific, but it always requires a way to address a precise service within a given app and optionally hand over some input data. The generic sequence of exchanged messages is shown in figure 3. We implemented `TumacoKit`, a Cocoa Touch framework for iOS which follows the described mechanism. An app needs to expose a URL scheme and identifier to be addressable from within the TUMACO platform app and handle incoming URL queries, possibly containing data in the form of URL parameters. An Android-specific implementation could realise the same mechanism by exposing Intent Filter Actions, data being exchanged and encapsulated within Intent Extras.

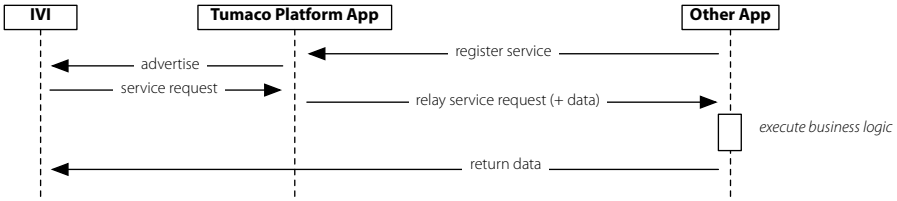


Fig. 3. `TumacoKit`: Service Integration

3.3 Custom User Experience & Web Components

To allow a consistent user experience for all passengers aboard a car, including the driver, the TUMACO platform app offers the user the possibility at the

connection to choose a screen. For the moment, we consider 5 screens: instrument cluster, central console, copilot, rear left, rear right. The screen - mobile device relationship is a one-to-one mapping, as a person may only sit at one seat at a time. The central console is a screen users may not check into, as it regroups all information sent from all connected mobile devices to give a global overview of the provided data to the driver (e.g. relevant location data). Due to privacy reasons, the other 4 screens show only the specific data transmitted from the corresponding mobile device. Customization can be achieved, as the user may design the layout of his specific screen by choosing the Web Components he wants to see, in the order and size he wants them to be put and even with the design options he wants (e.g. speedometer background color). For the driver, these customization aspects are especially important, as they may reduce confusion when using an unknown car with his own design preferences stored on his mobile phone. As Web Components are custom elements that can be imported into regular HTML5 code, they can be reused within the different screens, assigned with screen-specific data.

3.4 Use Case: Consensus on Music in Carpooling Scenario

In figure 4, we show how our introductive example is realised within Tumaco. After exchanging service capabilities, mobile devices detect that their Music Library Service (a Data Provider) matches the semantic types of the IVI's Music Collection Service and thus send their music library. The IVI detects the most represented common music genre and instructs a device, here the Android phone, to play a corresponding song. Latter now sees that the IVI has a service to display song information and sends the song's meta-information. Now, the IVI's Data Analyser component sees that this service also wants the album cover, which has not been transmitted by the Android device. Gracefully, the iPhone device provides a corresponding service and is instructed to download and send back the corresponding album cover. Now, the IVI has all necessary data and displays them in a Music Player Web Component at the UI layer. This example shows a complex workflow, where one service execution may generate a new service request. With Tumaco, it is thus easily possible to realise a high level service, distributed among different iOS and Android devices, without the need to change these mobile platforms. In addition, as the involved services are declared in a descriptive way and are semantically annotated, the RESTful Middleware layer and the above lying Service layer can reason and take decisions without human intervention.

4 Implementation

We have built a prototypical implementation of the Tumaco ecosystem, including platform apps for both iOS and Android. The IVI is represented through a Django web server running on a Raspberry Pi (Model B). We have tested the implementation on an iPhone 5, running iOS 8.4 and on a Samsung Galaxy S

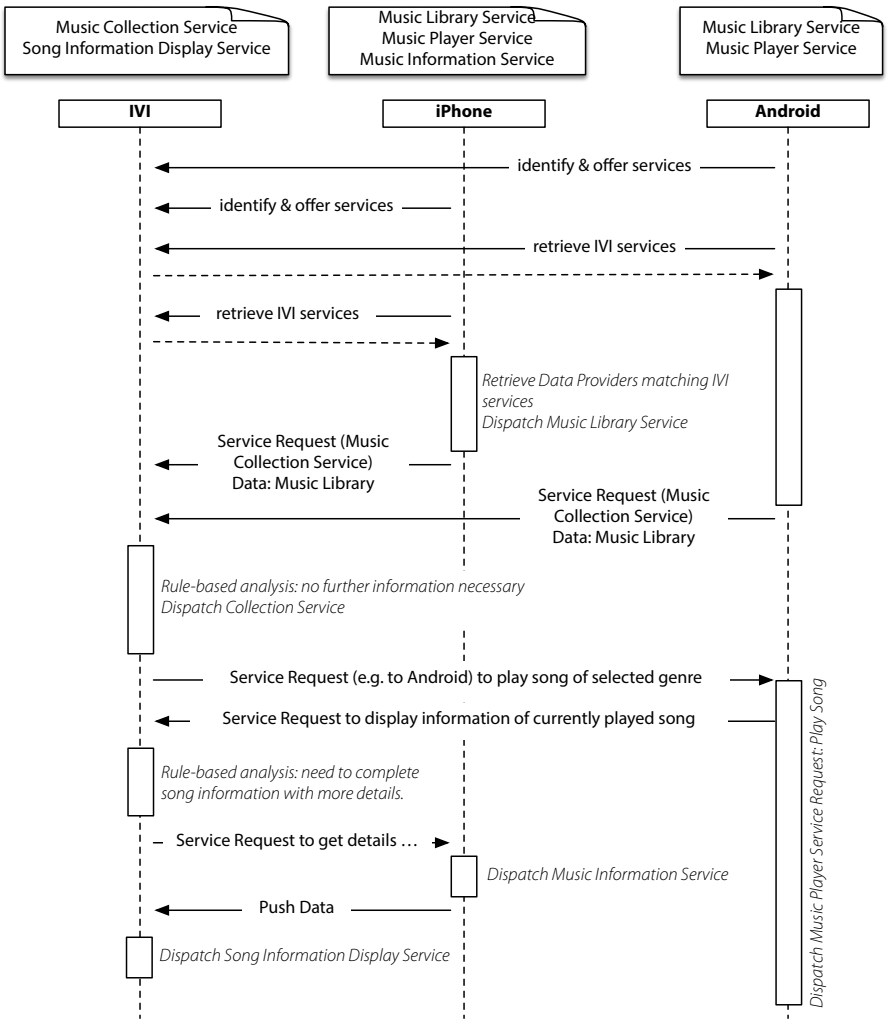


Fig. 4. Complete example: Consensus on Music

III, running Android 4.4. The implementation of the platform apps has shown the differences, advantages and disadvantages of building a common ecosystem using the iOS SDK in Swift and the Android SDK in Java. In particular, we use a WANET between the Raspberry Pi and the iPhone, and an infrastructure WLAN between the Raspberry Pi and the Galaxy S III, as Android does not support ad-hoc networks by default. The IVI's UI layer is built using HTML5, JavaScript and custom Web Components (e.g. navigation map, speedometer gauge, music player). For the moment, there only exists an iOS implementation of `TumacoKit`. Future work will include an Android implementation based on Intent actions and Extras equivalent to URL queries used in iOS.

5 Conclusion & Future Work

We proposed a SOA-based IVI system which seamlessly integrates mobile devices through semantic hypermedia support. Tumaco extends existing mobile platforms without requiring any changes to the genuine distribution. Services may engage in workflows and are both platform-independent and self-descriptive through semantics. `TumacoKit` allows mobile application developers to extend the ecosystem with their own services, without requiring architectural modifications. All passengers aboard may customize their user experience through Web Components, fed with semantically enriched data. We provided a full example of a complex workflow, which has been implemented in our prototype.

In the future, we plan to integrate available car data, either via the ubiquitously represented OBD-II interface or upcoming Web APIs accessing the CAN bus [14], into the Tumaco ecosystem. Privacy is of utmost importance, therefore the user needs to be given further control over which data gets shared with the IVI and on which screens it shall be accessible. New forms of displays such as HUDs on the windshield, replacing classical dashboards, may bring new opportunities for UI customization, to enable the driver to fully control his augmented reality user experience. Finally, services or even games over different cars could enable a new kind of social interaction in traffic jams.

References

1. G. Alliance. BMW Case Study. `http://www.genivi.org/sites/default/files/BMW_Case_Study_Download_040914.pdf`.
2. S. C. Boll, A. L. Kun, P. Fröhlich, and J. Foley. Automotive User Interface Research Moves into Fast Lane. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '13, pages 2525–2528, New York, NY, USA, 2013. ACM.
3. S. Diewald, T. Leinmüller, B. Atanassow, L.-P. Breyer, and M. Kranz. Mobile Device Integration and Interaction with V2X Communication. In *19th World Congress on Intelligent Transport Systems (ITS)*, 2012.
4. S. Diewald, A. Möller, L. Roalter, and M. Kranz. Mobile Device Integration and Interaction in the Automotive Domain. In *AutoNUI: Automotive Natural User Interfaces Workshop at the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI 2011)*, 2011.

5. S. Diewald, A. Möller, L. Roalter, and M. Kranz. DriveAssist – A V2X-Based Driver Assistance System for Android. In *Automotive HMI Workshop at Mensch und Computer 2012*, 2012.
6. M. Ebling and M. Baker. Pervasive Tabs, Pads, and Boards: Are We There Yet? *Pervasive Computing, IEEE*, 11(1):42–51, January 2012.
7. M. Eichhorn, M. Pfannenstein, D. Muhra, and E. Steinbach. A SOA-based middleware concept for in-vehicle service discovery and device integration. In *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 663–669, June 2010.
8. M. Feuer. Remote Vehicle Interaction - Architecture High Level Description. `https://download.automotivelinux.org/POC/PoC_Spec/15-456-POC-Tizen3-RVI-HLD_Draft4_clean.pdf`, 2014.
9. E. Gasperowicz. Creating semantic sites with Web Components and JSON-LD. `http://updates.html5rocks.com/2015/03/creating-semantic-sites-with-web-components-and-jsonld`, 2015.
10. F. Hüger. User Interface Transfer for Driver Information Systems: A Survey and an Improved Approach. In *Proceedings of the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, AutomotiveUI '11, pages 113–120, New York, NY, USA, 2011. ACM.
11. C. Kennington, S. Kousidis, T. Baumann, H. Buschmeier, S. Kopp, and D. Schlangen. Better Driving and Recall When In-car Information Presentation Uses Situationally-Aware Incremental Speech Output Generation. In *Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, AutomotiveUI '14, pages 7:1–7:7, New York, NY, USA, 2014. ACM.
12. M. Lanthaler and C. Gütl. On Using JSON-LD to Create Evolvable RESTful Services. In *Proceedings of the Third International Workshop on RESTful Design*, WS-REST '12, pages 25–32, New York, NY, USA, 2012. ACM.
13. S. Murphy, A. Nafaa, and J. Serafinski. Advanced service delivery to the Connected Car. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2013 IEEE 9th International Conference on*, pages 147–153, Oct 2013.
14. S. Murphy, P. Szczechowiak, J. Serafinski, and T. Zernicki. Car-mobile integration for advanced telematics services delivery. In *Consumer Communications and Networking Conference (CCNC), 2014 IEEE 11th*, pages 71–76, Jan 2014.
15. H. Paulheim and F. Probst. A Formal Ontology on User Interfaces Yet Another User Interface Description Language. In *2nd Workshop on Semantic Models for Adaptive Interactive Systems,(SEMAIS)*, 2011.
16. L. Richardson, M. Amundsen, and S. Ruby. *RESTful Web APIs*. O'Reilly Media, 2013.
17. J. Sonnenberg. A distributed in-vehicle service architecture using dynamically created web Services. In *Consumer Electronics (ISCE), 2010 IEEE 14th International Symposium on*, pages 1–5, June 2010.
18. J. Sonnenberg. Service and User Interface Transfer from Nomadic Devices to Car Infotainment Systems. In *Proceedings of the 2Nd International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, AutomotiveUI '10, pages 162–165, New York, NY, USA, 2010. ACM.
19. J. Sonnenberg. Connecting in-vehicle entertainment with CE devices via Multi-Platform Web Applications. In *Consumer Electronics (ICCE), 2011 IEEE International Conference on*, pages 515–516, Jan 2011.
20. M. Weiser. The Computer for the 21st Century. *Scientific American*, 265(3):66–75, 1991.

Prototipo móvil 3D para el aprendizaje de algoritmos básicos

Federico Cristina [1], Sebastián Dapoto[1], Pablo Thomas[1], Patricia Pesado[1,2],

[1] Instituto de Investigación en Informática LIDI
Universidad Nacional de La Plata – Argentina

[2] Comisión de Investigaciones Científicas de la Provincia de Buenos Aires - Argentina

{fcristina, sdapoto, pthomas, ppesado}@lidi.info.unlp.edu.ar

Resumen. El ámbito educativo debe adaptarse a los cambios y nuevas formas de aprendizaje. M-learning plantea métodos modernos de apoyo al proceso de aprendizaje mediante el uso de dispositivos móviles. De esta forma es posible disponer de las facilidades de e-learning en cualquier momento y lugar. En particular resulta de gran interés el desarrollo de herramientas de software que brinden soporte para el aprendizaje en los niveles iniciales de las carreras informáticas. Esto, sumado al potencial de los dispositivos móviles actuales, posibilita el desarrollo de un prototipo de aplicación visual 3D para el aprendizaje de algoritmos básicos, el cual se presenta en este trabajo.

Palabras clave: M-learning, Unity, algoritmos básicos

1 Introducción

Actualmente las personas utilizan dispositivos móviles tales como teléfonos celulares, smartphones o tablets, como un accesorio más dentro de su vida cotidiana. Por medio de diversas aplicaciones a las que se puede acceder, se obtiene información de distinta índole, servicios y entretenimientos, se realizan transacciones comerciales, y hasta es posible controlar aspectos de salud de las personas. Además, muchas de estas aplicaciones móviles permiten que sus usuarios estén conectados entre sí [1].

De la misma forma, esta nueva tecnología trae consigo un potencial que puede ser usado en procesos de enseñanza y aprendizaje [2][3]. M-learning es la evolución de e-learning a través del uso de dispositivos móviles. De este modo el aprendizaje se transforma en una actividad personalizada, portátil, cooperativa e interactiva.

Una de las características más importantes que plantea M-learning es la de brindar completa flexibilidad para los alumnos, es decir, posibilitar la selección del contenido que se desea ver, en el momento y lugar elegidos. Además, los contenidos no deben ser dependientes de un dispositivo en particular. Por último, a esta independencia tecnológica se le debe añadir la adaptación de los contenidos, teniendo en cuenta la

navegabilidad, capacidad de procesamiento y velocidad de conexión de los diferentes dispositivos.

No obstante, en la actualidad una gran parte de los dispositivos móviles posee un alto grado de sofisticación. Su evolución tecnológica permite ejecutar aplicaciones cada vez más complejas y con requerimientos de hardware más grandes. Esto es de gran importancia al momento de desarrollar herramientas con gráficos en 3D que necesitan de estas prestaciones para poder ejecutarse de forma adecuada.

Las herramientas que poseen un entorno 3D son visualmente más agradables y atraen más fuertemente a los usuarios que las utilizan. Esto es en gran parte porque los ambientes 3D simulados son más cercanos a la realidad que los bidimensionales, lo que permite al usuario involucrarse de forma más activa con el entorno.

El resto del trabajo está organizado de la siguiente forma: la sección 2 describe la motivación; la sección 3 muestra el análisis realizado sobre las herramientas de desarrollo móvil 3D existentes; la sección 4 presenta el prototipo desarrollado; la sección 5 expone las conclusiones; por último, la sección 6 plantea el trabajo a futuro.

2 Motivación

Los conceptos a incluir en los niveles iniciales de carreras informáticas y afines, generan un escenario propicio para poder crear herramientas que soporten las características que plantea m-learning.

En particular, en el curso de ingreso de las carreras de la Facultad de Informática de la Universidad Nacional de La Plata se utiliza una herramienta de software denominada R-Info para el aprendizaje de los conceptos básicos de la construcción de algoritmos [4][5][6][7][8]. A través de la misma, los alumnos pueden crear algoritmos de simple y mediana complejidad y visualizar la ejecución de dichos algoritmos. Es decir, la herramienta les permite resolver problemas en un entorno visual y atractivo.

R-Info es una máquina abstracta simple, un robot móvil controlado por un conjunto reducido de instrucciones primitivas, que permite modelizar recorridos y tareas dentro de una ciudad formada por calles (arterias horizontales) y avenidas (arterias verticales). En la figura 1 se puede observar que la ciudad es cuadrada, posee manzanas y consta de 100 calles y 100 avenidas.

El robot cuenta con las siguientes capacidades básicas a realizar en la ciudad:

1. Avanzar hacia adelante 1 cuadra.
2. Orientarse hacia la derecha (girar 90 grados en el sentido de las agujas del reloj).
3. Reconocer dos formas de objetos preestablecidas: flores y papeles. Éstos se hallan ubicados en las esquinas de la ciudad (intersecciones de calles y avenidas).
4. Transportar flores y papeles en una bolsa que dispone. Además, está capacitado para recoger y/o depositar cualquiera de los dos tipos de objetos en una esquina, pero de a uno a la vez. La bolsa posee capacidad ilimitada.
5. Posicionarse directamente en cualquier esquina de la ciudad.
6. Realizar cálculos simples, incluyendo variables de ser necesario.
7. Utilizar estructuras de control, tales como “mientras” o “repetir”.

8. Informar los resultados obtenidos.

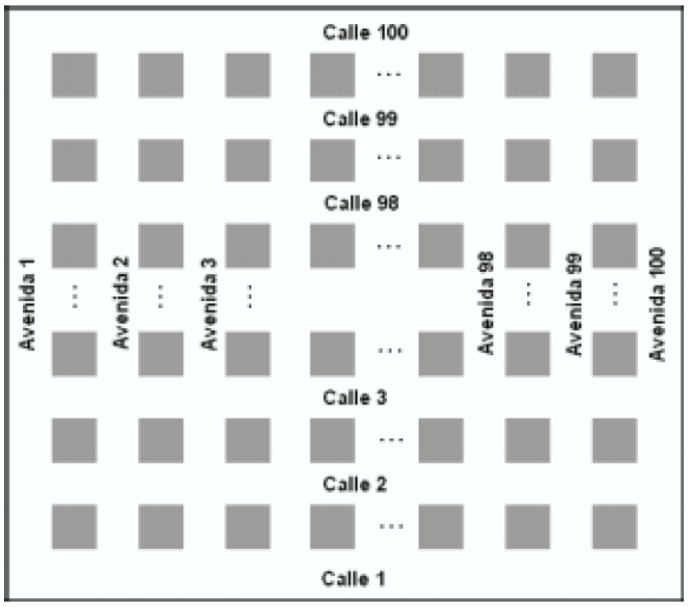


Fig. 1. La ciudad en donde se traslada el robot.

La pantalla principal de la aplicación R-Info cuenta con un panel de control que permite, entre otras funciones, escribir el código del algoritmo y luego ejecutarlo. Además la herramienta permite editar el estado de la ciudad (agregar flores y/o papeles) y visualizar una porción de dicha ciudad.

En la parte superior de la figura 2 se puede observar los controles del panel y el código del algoritmo. En la parte inferior de dicha figura se visualiza el recorrido que realizó el robot de acuerdo a la ejecución del algoritmo citado anteriormente. Este recorrido traslada al robot desde la esquina (1,1) hasta la esquina (3,2). Como se puede observar en la figura 2, también es posible ver las esquinas que poseen alguna flor y/o papel.

Actualmente, los alumnos ingresantes de las carreras informáticas cuentan en gran parte con un dispositivo móvil. Es por esto que se encuentran familiarizados con el uso de este tipo de tecnología, a la que transformaron en parte cotidiana de sus vidas, y con la que cuentan en cualquier momento y lugar. Además, para muchos de los alumnos ingresantes, el acceso a un equipo de escritorio resulta menos cómodo. Teniendo en cuenta esto, es muy conveniente contar con una versión móvil para incentivar el uso de las aplicaciones que estén destinadas al mejor aprendizaje de los alumnos.

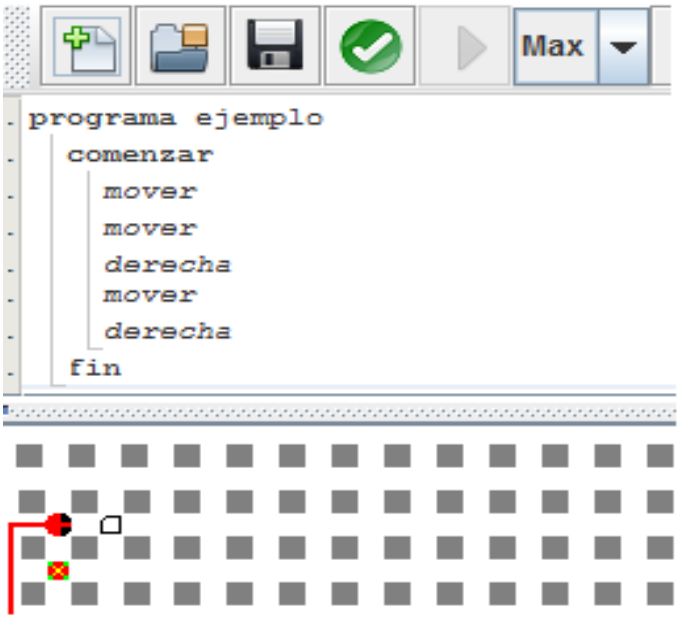


Fig. 2. R-Info.

Por otro lado, es posible añadir un mayor potencial a esta versión móvil mediante una mejora sustancial de la visualización del robot y las tareas que realiza dentro de la ciudad. Esto es posible desarrollando una interfaz gráfica 3D que transforme la nueva versión en una herramienta visualmente más atractiva, más interesante y propensa al uso [9][10].

3 Herramientas de desarrollo 3D

Si bien no es propósito de este trabajo realizar un análisis exhaustivo de las herramientas y librerías existentes para lograr el objetivo planteado, se han revisado básicamente los principales referentes actuales en el área.

En particular existen dos herramientas de software que se destacan: Unity[11] y Unreal Engine[12] debido mayormente a su popularidad, funcionalidad y versatilidad. Ambas difieren considerablemente en lo que respecta a lenguajes de programación, licencias de uso, soporte, etc. Es por esto que en una primera etapa se realizó una evaluación preliminar a fin de establecer la opción más adecuada para el desarrollo propuesto.

Para ésto se analizaron varios factores, como por ejemplo: curva de aprendizaje, costo, lenguajes soportados, performance en ejecución, comunidad de usuarios, requerimientos de hardware, plataformas de desarrollo y ejecución, etc.

Como resultante del análisis preliminar, se determinó que tanto Unity como Unreal Engine cumplen con los requisitos necesarios del caso (comprendidos entre los

criterios previamente mencionados). Sin embargo, Unity es más simple e intuitivo en cuanto a su utilización.

Adicionalmente, el uso del lenguaje C# en Unity vs el lenguaje C++ en Unreal Engine se consideró una ventaja, dado que el proyecto original de R-Info se encuentra escrito en el lenguaje Java (similar en varios aspectos al lenguaje C#).

Si bien con ciertas diferencias, ambas alternativas presentan opciones de utilización bajo licencia sin costo; con lo cual en este tema no hay mayores ventajas de uno sobre otro.

El número de usuarios - y por ende su comunidad y soporte - con el que cuenta Unity supera considerablemente el de Unreal Engine. Por ejemplo, el sitio StackOverflow [13] contiene 11147 preguntas con el tag Unity mientras que solo 140 con el tag Unreal Engine al momento de escribir este trabajo. La figura 3 muestra la popularidad según Google Trends [14] quien también presenta una considerable ventaja de Unity por sobre Unreal Engine, y su tendencia parece mantenerse al menos a mediano plazo.

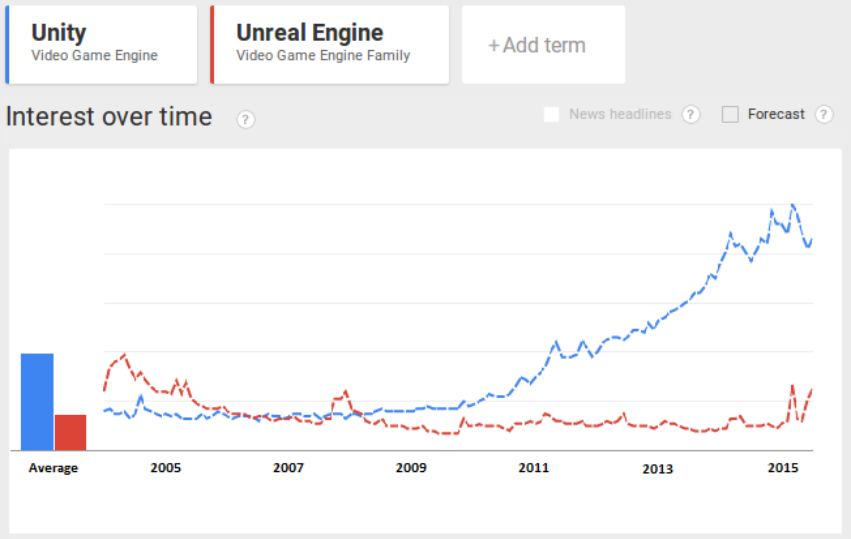


Fig. 3. Google Trends: Unity vs. Unreal Engine.

Por último, la ejecución de aplicaciones de prueba realizados sobre dispositivos móviles se notaron más fluidas en Unity. En dichas pruebas se intentó representar exactamente el mismo escenario en ambas plataformas, utilizando el mismo número de polígonos en los objetos, aplicación de texturas idénticas y configuración similar de iluminación y sombras.

Por todo lo expuesto, Unity es la herramienta elegida para desarrollar el prototipo que se presenta en este trabajo.

4 Prototipo móvil 3D

Con el propósito de disponer de una nueva herramienta de software, flexible, que pueda utilizarse en cualquier momento y lugar, siendo además más atractiva, inicialmente se desarrolló un prototipo móvil 3D básico. Dicho prototipo implementa un subconjunto de las instrucciones primitivas de R-Info dentro del entorno Unity.

La interfaz visual es completamente tridimensional, por lo que la ciudad y los objetos dentro de ella pueden ser observados desde distintos puntos de vista. Haciendo uso de esta ventaja, efectivamente el prototipo permite seleccionar diferentes cámaras o puntos de vista. Además, en todas las vistas es posible regular el nivel de zoom para acercarse o alejarse de la escena.

La figura 4 muestra un ejemplo del prototipo, en la cual puede observarse el editor de código, y en segundo plano la ciudad en donde se realiza la simulación al presionar el botón Ejecutar. La vista utilizada en este caso es la de primera persona.

La figura 5 muestra la ejecución del algoritmo en tercera persona con un nivel de zoom intermedio. En la parte superior se puede observar la información sobre la posición del robot, su orientación, el número de flores y papeles tanto en el robot como en la esquina en la cual se encuentra posicionado. En la parte inferior se visualiza la instrucción que está siendo ejecutada. Tanto el nivel de zoom como la velocidad son configurables con los controles de barra laterales.

Fig. 4. Panel de edición de código y vista en primera persona de la ciudad.

Además de los dos puntos de vista mencionados (primera y tercera persona), existe una tercera posibilidad que permite visualizar la ciudad desde el cielo, en lo que representa una visualización similar a la provista por la herramienta R-Info. Esta última opción se puede observar en la figura 6.

En todas las figuras mencionadas, también es posible observar las flores y papeles dispuestos sobre algunas de las esquinas de la ciudad.

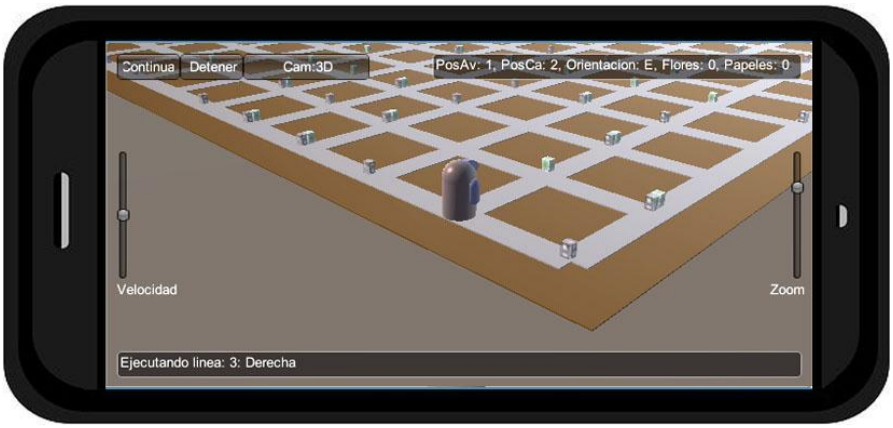


Fig. 5. Panel de ejecución. Vista en tercera persona.

El prototipo también permite otros ajustes tales como regulación de la velocidad de ejecución de la animación, apertura de códigos desde archivos externos y selección del idioma a utilizar. Adicionalmente, permite ejecutar el programa paso a paso o bien reiniciarlo en cualquier punto del mismo.

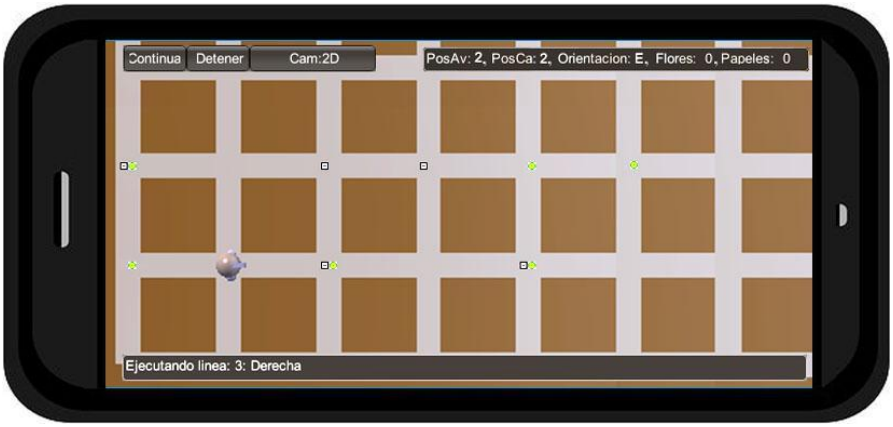


Fig. 6. Vista desde el cielo.

El ambiente tridimensional desarrollado permite a cada alumno seleccionar la vista de su preferencia, fortaleciendo su experiencia al ejecutar algoritmos. Además, la naturaleza móvil del prototipo permite a los alumnos utilizarlo en cualquier momento y lugar, lo que representa una gran ventaja con respecto a la versión de escritorio.

5 Conclusiones

Se ha desarrollado un prototipo móvil 3D de una herramienta que brinda soporte para el aprendizaje de algoritmos básicos en los niveles iniciales de las carreras informáticas.

Al ser un prototipo móvil, motiva y facilita al alumno la tarea de comprender los conceptos básicos de algoritmia, dado que permite realizar pruebas de ejecución de programas en cualquier lugar en que el alumno se encuentre. De esta forma se cumple con una de las características principales de M-learning, al brindar total flexibilidad respecto al momento y lugar de uso.

El entorno 3D es un atractivo adicional para que los alumnos sientan interés por la utilización del prototipo. Además, al disponer de varios puntos de vista diferentes, permite una mejor visualización de las tareas que realiza el robot sobre la ciudad.

En resumen, el prototipo móvil 3D intenta mejorar la experiencia del alumno en su tarea de aprendizaje, constituyéndose en una herramienta muy útil para el inicio de cualquier curso de enseñanza de algoritmos básicos.

6 Trabajo a futuro

El principal trabajo a futuro a llevar adelante es el de brindar el soporte completo de las funcionalidades originalmente ofrecidas por R-Info, adecuando la solución móvil 3D.

A fin de evitar realizar una migración completa del código base de R-Info, se prevé aplicar una refactorización que tenga como finalidad lograr soportar dos modos de uso de R-Info, uno de tipo “aplicación” o de uso tradicional, y uno de tipo “librería” o de soporte para uso externo.

Este último modo será sobre el que se apoye la solución móvil 3D. De esta manera se evita la migración completa del código y se garantiza la compatibilidad en caso de una eventual evolución posterior de R-Info.

Finalmente, se prevén mejoras visuales para la solución móvil 3D, como opciones de calidad en la renderización de video, selección de distintos tipos de robots, efectos visuales y sonoros en las animaciones, entre otras.

Referencias

1. Cristina, F.; Dapoto, S.; Thomas, P.; Pesado, P. “A simplified multiplatform communication framework for mobile applications”. IEEE 9th International Conference on Computer Engineering & Systems (ICCES); El Cairo, Egipto. Dec. 2014. ISBN 978-1-4799-6593-9.
2. Kantel E., Tovar G., Serrano A.”Diseño de un Entorno Colaborativo Móvil para Apoyo al Aprendizaje a través de Dispositivos Móviles de Tercera Generación.” IEEE-RITA 5, no. 4. Nov. 2010. ISSN 1932-8540.
3. Yadegaridehkordi, E.; Iahad, N.A.; Mirabolghasemi, M. “Users’ Perceptions towards M-learning Adoption: An Initial Study”. IEEE International Conference on Research and

Innovation in Information Systems (ICRIIS); Kuala Lumpur, Malaysia. Nov. 2011. ISBN: 978-1-61284-295-0.
4. De Giusti A.; Frati E.; Leibovich F.; Sanchez M.; De Giusti L.; Madoz M. “LMRE: Un entorno multiprocesador para la enseñanza de conceptos de concurrencia en un curso CS1”. XVII Congreso Argentino de Ciencias de la Computación (CACIC). Oct. 2011. ISBN: 978-950-34-0756-1
5. De Giusti A.; Frati E.; Leibovich F.; Sanchez M.; De Giusti L. "LIDI Multi Robot Environment: Support software for concurrency learning in CS1". International Conference on Collaboration Technologies and Systems (CTS); Denver, USA. May 2012. ISBN: 978-1-4673-1380-3
6. De Giusti L.; Leibovich F.; Sánchez M.; Chichizola F.; Naiouf M.; De Giusti A. "Desafíos y herramientas para la enseñanza temprana de Concurrencia y Paralelismo". XIX Congreso Argentino de Ciencias de la Computación (CACIC). Oct. 2013. ISBN: 978-987-23963-1-2.
7. De Giusti L.; Leibovich F.; Sánchez M.; Rodriguez Eguren S.; Chichizola F;, Naiouf M.; De Giusti A. "Herramienta interactiva para la enseñanza temprana de Concurrencia y Paralelismo: un caso de estudio”. XX Congreso Argentino de Ciencias de la Computación (CACIC). Oct. 2014. ISBN: 978-987-3806-05-6.
8. De Giusti A.; De Giusti L.; Leibovich F.; Sanchez M.; Rodriguez Eguren S. "Entorno interactivo multirrobot para el aprendizaje de conceptos de Concurrencia y Paralelismo". Congreso de Tecnología en Educación y Educación en Tecnología (TE&ET). 2014.
9. Paredes R.; Sánchez J.A.; Rojas L.; Strazzulla D.; Martínez-Teutle R. "Interacting with 3D Learning Objects". IEEE Latin American Web Congress; Merida, Mexico. Nov. 2009. ISBN: 978-0-7695-3856-3.
10. Hesse S.; Gumhold S. "Web based Interactive 3D Learning Objects for Learning Management Systems". International Conference on Education, Training, and Informatics (ICETI); Orlando, USA. Mar. 2011. ISBN: 978-161-8394-87-3.
11. Unity 3D Homepage: https://unity3d.com/
12. Unreal Engine Homepage: https://www.unrealengine.com/
13. Comparación entre Unity3D y Unreal Engine en cuanto al número de preguntas para cada plataforma: http://stackoverflow.com/questions/tagged/unity3d vs http://stackoverflow.com/questions/tagged/unreal-engine4
14. Comparación de interés entre ambas plataformas según Google Trends: https://www.google.com/trends/explore#q=%2Fm%2F0dmyvh%2C%20%2Fm%2F025wnp&cmpt=q&tz=Etc%2FGMT%2B3

Desarrollo de Herramientas para Warehousing en el Municipio de Junín

Mariana Adó, María Cecilia Rastelli, Ana Smail[1], Rodolfo Bertone[1,2]

[1] Universidad Nacional del Noroeste de la Provincia de Buenos Aires, Junín, Argentina
[2] Facultad de Informática, UNLP, La Plata, Argentina
{mariana.ado@itt.unnoba.edu.ar, cecirastelli@hotmail.com,anasmail@unnoba.edu.ar
pbertone@lidi.info.unlp.edu.ar}

Abstract. Desde hace algunos años, el Gobierno Local de Junín promueve iniciativas de modernización para lograr una gestión más simple, transparente y eficiente.
Uno de los pilares de este proceso de modernización ha sido el desarrollo basado en E-Govermnet. Esto implica la incorporación de Tecnologías de Información y Comunicación (TICs) en organizaciones del Estado, con el objetivo de mejorar no solo la gestión interna, sino también la interacción con los proveedores y los ciudadanos.
Estos motivos, llevaron a plantear el desarrollo del Data Warehouse (DW) para el municipio, que permita centralizar y administrar de una manera más eficiente la información de la comuna. El objetivo primario perseguido es buscar que el conocimiento generado sirva de apoyo a la toma de decisiones en la implementación de políticas públicas que permitan una mejora continua en los servicios que se brindan a la comunidad.
Este trabajo consistió, en una primera etapa, en estudiar el estado del arte de DW aplicado a E-Gobierno. Se evaluaron diferentes alternativas con el fin definir una metodología de trabajo consistente y adecuada para la implantación en el municipio. Posteriormente, y a partir del trabajo realizado, se utilizó una herramienta Open Source (OS) con la que se desarrolló el primer Data Mart aplicado al área de Recursos Humanos.

Keywords: E-Goverment, Data Warehousing, Data Mart

1 Introducción

La gestión de información en las empresas es, actualmente, una herramienta clave para sobrevivir en un mercado cambiante, dinámico y global. Aprender a competir con esta información es fundamental para la toma de decisiones, el crecimiento y la gestión de las organizaciones. La disciplina denominada como Business Intelligence (BI) permite el acercamiento a los sistemas de información que ayudan a la toma de decisiones en la organización. [1]

Una *decisión* es una elección que se hace entre varias alternativas [2] y optar por aquella que se considera más correcta, es un proceso esencial en el contexto social globalizado. La mayor parte de las decisiones gerenciales carecen de estructura y conllevan riesgo, incertidumbre y conflicto, por lo que es necesario contar con información de cada una de las alternativas de solución y sus posibles consecuencias según los resultados que se esperan obtener. La importancia de la información en la toma de decisiones queda clara en la definición de decisión propuesta por Forrester, entendiendo por esta "el proceso de transformación de la información en acción" [3].

BI es un proceso interactivo para explorar y analizar información estructurada sobre un área (normalmente almacenada en un data warehouse), para descubrir tendencias o patrones, a partir de los cuales derivar ideas y extraer conclusiones. El proceso de BI incluye la comunicación de los descubrimientos y efectuar los cambios. Las áreas incluyen clientes, proveedores, productos, servicios y competidores. [4] BI es un término paraguas que abarca los procesos, las herramientas, y las tecnologías para convertir datos en información, información en conocimiento y planes para conducir de forma eficaz las actividades de los negocios. BI abarca las tecnologías de data warehousing, los procesos en el 'back end', consultas, informes, análisis y las herramientas para presentar información y los procesos en el 'front end'. [5]

1.1 Data Warehouse

Un Data Warehouse (DW) es una colección de datos orientada al negocio, integrada, variante en el tiempo y no volátil para el soporte del proceso de toma de decisiones de la gerencia. [6] Un DW es una copia de los datos transaccionales específicamente estructurada para la consulta y el análisis. [7]

Una Arquitectura Data Warehousing (DWA) es una forma de representar la estructura total de los datos, la comunicación, el procesamiento y la presentación de los mismos, ante los usuarios finales que disponen de una computadora dentro de la empresa. [8]

Los tipos de tecnología que se pueden aplicar son:

- OLTP: La tecnología OLTP está conformada por todos aquellos medios que soportan las operaciones diarias de la organización. No ha sido diseñado para proporcionar funciones potentes de síntesis, análisis y consolidación de los datos. (Bases de datos relacionales, archivos de texto, XML, Word, Excel, etc.)
- OLAP: La tecnología OLAP facilita el análisis de datos en línea en un DW, proporcionando respuestas rápidas a consultas analíticas complejas.

El área de datos puede ser defina como:

- ODS: (Operational Data Store) es la información de los sistemas transaccionales, con la cual se actualizará el DW. Se admite un pequeño desfase o retraso sobre los datos operacionales.
- Staging Área: sistema que permanece entre la fuente de datos OLTP y el DW. Facilita la extracción de datos desde las fuentes de origen. Minimiza la afectación a los sistemas origen, ya que los datos de los sistemas

transaccionales pueden ser extraídos desde su almacenamiento al SA, en horarios donde no haya actividad, para no comprometer los sistemas OLPT, y ser usados desde el SA en otro momento.

- Data Mart: Un Data Mart es un subconjunto de los datos del DW con el objetivo de responder a un determinado análisis, función o necesidad y con una población de usuarios específica.

1.2 Metodología Ralph Kimball

Este enfoque también se lo denomina Bottom-Up, pues el DW final, consiste en la unión de los diferentes Data Marts. Esta característica hace que sea más flexible y sencillo de implementar, pues se puede generar un Data Mart como primer elemento del sistema de análisis, y luego incorporar otros que comparten las dimensiones ya definidas e incluyen otras nuevas. Con esta solución, los procesos ETL (Extrat, Transform, Load) extraen la información de los sistemas operacionales, los procesan y transforman en la Staging Area, para que los datos tengan la estructura del DW; y posteriormente se realiza la carga de cada uno de los Data Mart de una forma individual, aunque siempre respetando la estandarización de las dimensiones conformadas.

Los datos transaccionales son estructurados de una forma especial para el análisis, de acuerdo al Modelo Dimensional (no normalizado), que incluye las dimensiones de análisis y sus atributos, su organización jerárquica, así como los diferentes hechos de negocio y las medidas involucradas.

Las componentes de un modelo multidimensional son las que se detallan a continuación:

- Áreas temáticas: son las distintas áreas de interés que se quieren analizar en una organización.
- Hechos: es un patrón de interés o evento dentro de la empresa que se desea analizar.
- Medidas: son cuantificadores del desempeño de un hecho.
- Dimensiones: son una colección de entidades del mismo tipo, y mediante las cuales el usuario podrá filtrar y manipular la información almacenada, más precisamente, podrá analizar los hechos.

2 Investigación y Comparación de herramientas OSBI

Se investigaron diferentes herramientas OSBI. Las mismas fueron comparadas mediante el Método de Multicriterio por Scoring [9], y se optó por aquella que, a criterio de los autores, tiene las mejores características para implantar la solución al problema.

La selección de herramientas para la comparación se basó en el pronóstico publicado por la consultora Gartner en 2014, donde plantea como las tres mejores soluciones OSBI del mercado, a las herramientas: Pentaho, Jasper Sotf y Palo.

A partir de este estudio, se compararon dichas herramientas, teniendo en cuenta los siguientes criterios del BI:

1. Calidad de las herramientas de Integración de Datos.
2. Calidad de las herramientas de Análisis OLAP de los datos
3. Calidad de las herramientas de Reportes.
4. Calidad de las herramientas de Cuadros de Mandos.
5. Calidad de las herramientas de Minería de Datos.

El estudio realizado por el Método de Multicriterio por Scoring, arrojó como resultado, que la mejor herramienta OSBI del mercado es la Suite de Pentaho BI.

3 Data Mart de Liquidación de sueldos

Como parte del trabajo de desarrollo donde se implanten las técnicas analizadas, se realizó la construcción de un Data Mart de Liquidación de Sueldos para la Municipalidad de Junín. Si bien el objetivo final es el desarrollo de un DW que contemple todas las actividades del municipio se decidió comenzar por esta actividad a fin de analizar en campo el estudio realizado y proveer a las autoridades de una herramienta, al menos parcial, para la toma de decisiones. Por el ámbito en que se desarrolló el DM y los recursos destinados, se utilizó el enfoque de Ralph Kimball.

Para el caso práctico, se identifica un hecho: la liquidación. Cada transacción se traduce en una liquidación. El modelo multidimensional está compuesto por la tabla de hechos H_liquidación y las dimensiones Tiempo, Personal, Tanda, Nivel, Categoría, Régimen Horario, Contrato y Concepto. La figura 1 representa el diagrama del modelo de datos.

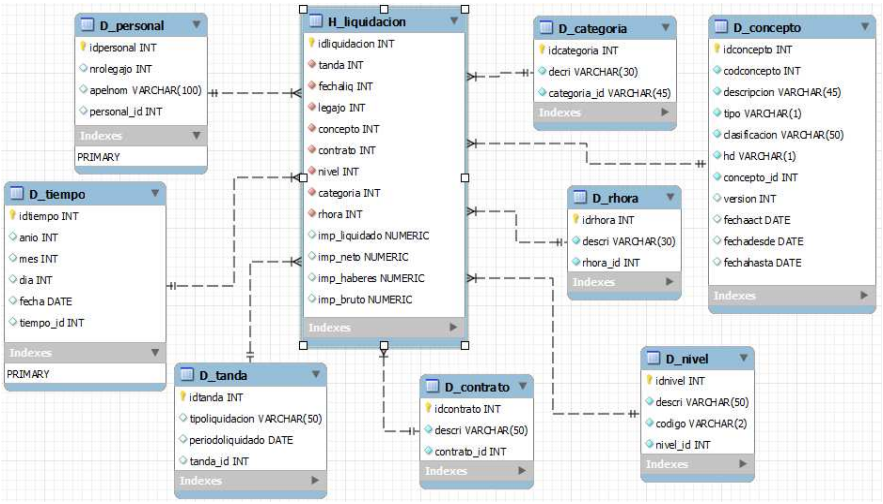


Figura 1. Modelo Multidimensional del Data Mart de Liquidación de Sueldos.

En la tabla 1 se describen las dimensiones del modelo realizado y la Tabla 2 describe las medidas del modelo.

Nombre	Descripción
Tiempo	Año y mes correspondiente a la liquidación.
Personal	Legajo al cual se le realizó la liquidación.
Concepto	Concepto liquidado.
Tanda	Tipo de liquidación que se realizó.
Nivel	Dependencia a la que pertenece el legajo.
Contrato	Tipo de contrato del legajo.
Categoría	Tipo de categoría del legajo.
Régimen Horario	Tipo de régimen horario del legajo.

Tabla 1. Dimensiones del modelo.

Nombre	Descripción
Importe Liquidado	Importe liquidado por concepto.
Importe Neto	Esta medida se calcula a partir del importe y del tipo de concepto. Corresponde al Importe Neto a cobrar de bolsillo.
Importe de Haberes	Esta medida se calcula a partir del importe y del tipo de concepto. Se suman todos los importes de los conceptos que correspondan al tipo H (Haberes).
Importe Bruto	Esta medida se calcula a partir del importe y del tipo de concepto. Se suman todos los importes de los conceptos de tipo H (Haberes) y se restan los conceptos de tipo D (Descuento) y clasificación RET (Retenciones).

Tabla 2. Medidas del Modelo.

Para los procesos ETL se utiliza la herramienta PDI (Pentaho Data Integration), también conocida como Kettle.

En ella, se realizan todas las tareas de extracción de datos de la base de datos municipal, que se encuentra en Oracle, a una Staging Area (SA) en MySQL Server 5.5. En la SA se procesan y transforman los datos extraídos, para darles un formato acorde a la estructura del Data Mart es decir, se depuran, se crean las tablas dimensiones y la tabla de hechos.

Por último, se copian las tablas de dimensiones y la tabla de hechos en el repositorio del Mart, que también se encuentra en una base de datos en MySQL Server 5.5.
La figura 2, muestra la tarea de transformación para la creación de tabla de dimensiones de conceptos en la SA. La figura 3, en tanto, muestra el trabajo final, donde se ejecutan todas las transformaciones para cargar en el repositorio del Mart, las tablas de dimensiones y la tabla de hechos.
Cuando ya se tienen las tablas componentes del Data Mart de Liquidación de Sueldos en el repositorio, se genera y se publica el esquema del mismo en el servidor de Pentaho, usando la herramienta PSW (Pentaho Schema WorkBench).

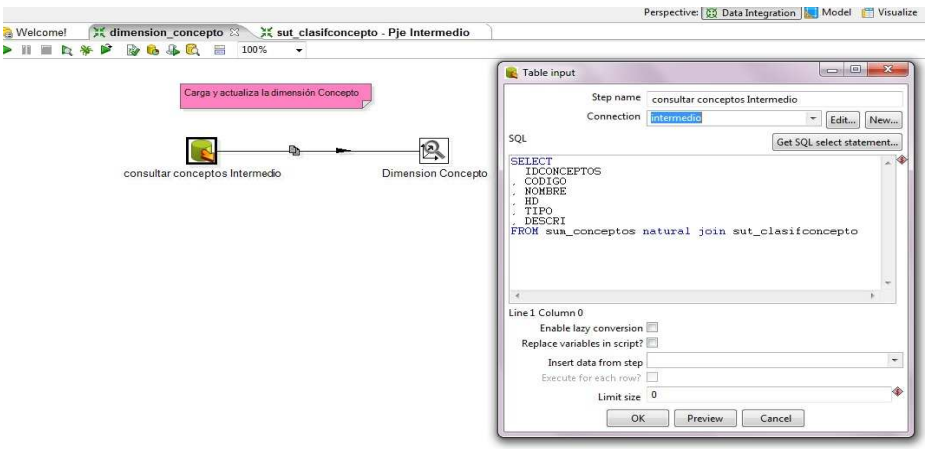


Figura 2. Tarea de transformación para la creación de tabla de dimensiones de conceptos en la SA.

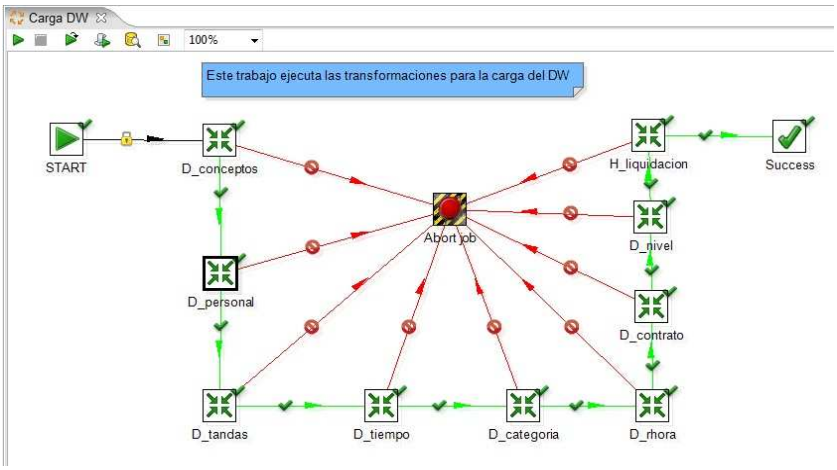


Figura 3. Trabajo para la carga de las tablas de dimensiones y la tabla de hechos en el DW.

La figura 4 muestra el esquema del Data Mart de Liquidación de Sueldos en PSW.

4 Resultados Obtenidos

Una vez desarrollado el Data Mart de Liquidación de Sueldos, para la construcción de vistas de análisis, se utiliza una herramienta OLAP destinada a usuarios finales de Pentaho, Saiku. Esta herramienta es una mejora de la interfaz gráfica del portal web Pentaho, que reemplaza a JPivot. Suministra a los usuarios un sistema avanzado de análisis de información. El usuario puede navegar por los datos, ajustando la visión de los mismos, aplicando filtros de visualización, añadiendo o quitando campos de agregación.

Se presentan algunos resultados obtenidos a partir del análisis de datos del DM. La figura 5 presenta la vista de la evolución de los totales por contratos del personal durante los años 2007 al 2013, en primer instancia se muestran los valores concretos y luego una representación gráfica de los mismo.

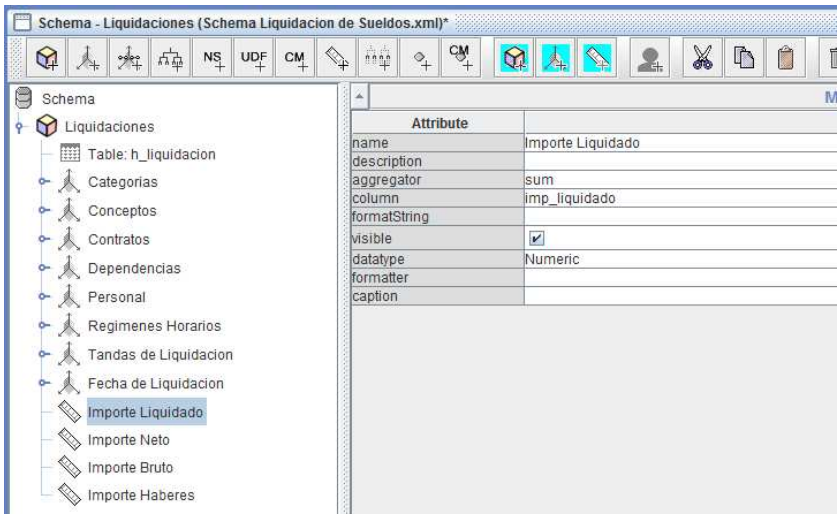


Figura 4. Esquema del Data Mart de Liquidación de Sueldos en PSW.

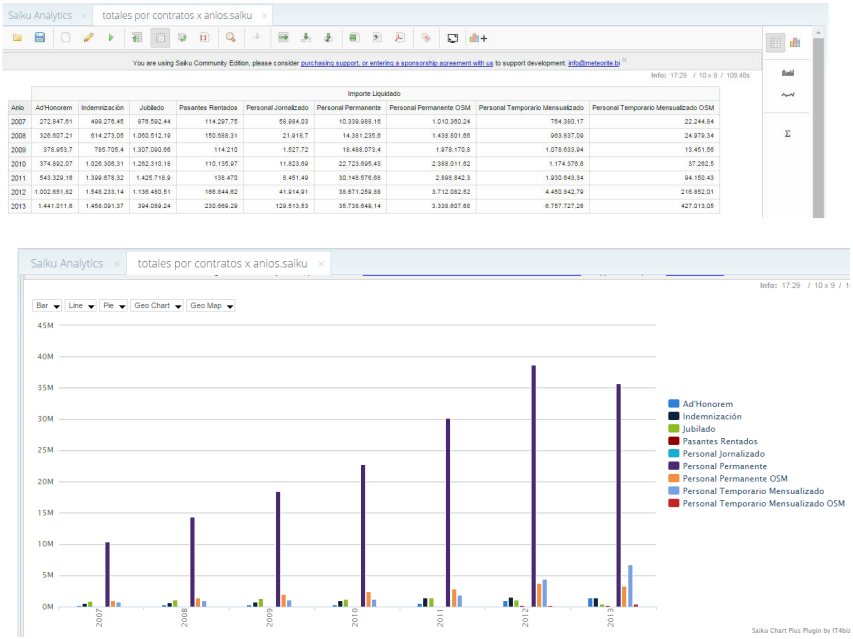


Figura 5. Representación Gráfica de resultados de evolución de contratos de empleados

La figura 6 muestra la vista de la evolución de los gastos de liquidación por categoría de la dependencia de Desarrollo de Sistemas mes a mes durante el año 2010.

		Importe Liquidado			
		DESARROLLO SISTEMAS			
Anio	Mes	JERARQ. II PROF.	JERARQUICO III	PROFESIONAL II	TECNICO I
2010	1	6.756,61	1.326	5.681,29	2.131,03
	2	7.096,61	1.333,88	5.681,29	2.131,03
	3	7.184,04	1.333,88	5.690,61	2.131,03
	4	6.912,04	1.333,88	5.702,66	2.099,96
	5	7.444,25	1.417,38	6.148,08	2.307,6
	6	10.384,84	2.811,91	11.188,16	3.311,4
	7	7.244,25	1.974,61	7.726,3	2.307,6
	8	7.980,25	1.974,61	7.743,37	2.307,6
	9	6.644,25	1.974,61	7.743,37	2.307,6
	10	6.661,56	1.974,61	7.743,37	2.307,6
	11	7.085,69	2.122,58	8.370,63	2.488,23
	12	10.531,24	3.033,87	12.158,75	3.582,34

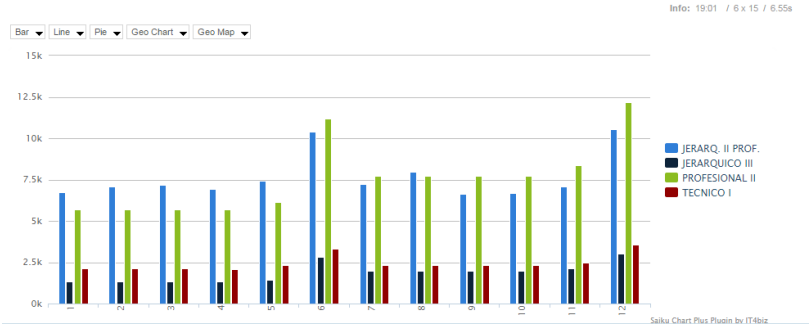


Figura 6. Gastos de liquidación de sueldos por categoría de empleados

La figura 7 representa la vista de la evolución anual por conceptos de la dependencia de Desarrollo de Sistemas desde el año 2007 al 2013.

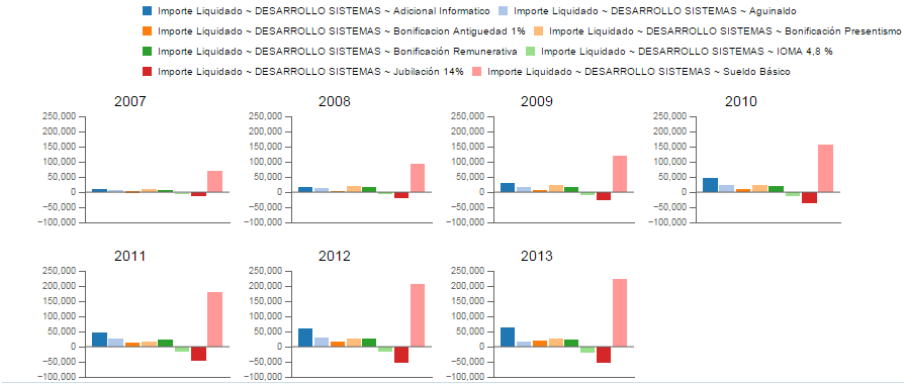


Figura 7. Evolución anual de conceptos de la Dependencia de Desarrollo de Sistemas

Los datos para este trabajo fueron extraídos del sitio http://www.junin.gob.ar.
A modo de ejemplo se presentaron algunos resultados obtenidos, con los que el área de gestión de personal del municipio puede trabajar y realizar diversos cruces de información a fin de tomar las decisiones que políticamente se consideren oportunas.

5 Conclusiones

El trabajo realizado tuvo como objetivo evaluar distintas alternativas OSBI, que permitan seleccionar una herramienta y definir una metodología, que permitan desarrollar el futuro DW de la Municipalidad de Junín.
Se investigaron y compararon diferentes tecnologías, y se llegó a la conclusión de implementar el primer Data Mart con la herramienta OSBI PENTAHO.
Se implementó un Data Mart de Liquidaciones de Sueldos el cual sirve para demostrar las virtudes y ventajas del uso de estas tecnologías.
Con el desarrollo realizado, se brindó al personal jerárquico de la Municipalidad de Junín una potente herramienta para el proceso de BI, con bajos costos.
Esto brinda una alternativa hasta ahora no disponible en el proceso de toma de decisiones, con información más precisa y oportuna para articular nuevas políticas, que permitan realizar una mejor gestión y prestar mejores servicios a los ciudadanos.

Referencias

1. Ana Smail - Taller Pentaho UNNOBA 2012. Anexo 1
2. Leon Blanck Buris. Administración en las Organizaciones: un enfoque estratégico. Centro Editorial Universidad del Valle. Facultad de Ciencias de la Administración. Cali 1990.
3. J Forrester, Dinámica Industrial. Editorial El Ateneo Buenos Aires, 1972
4. Gartner, www.gartner.com, enero 2006.
5. Data Warehousing Institute. El Data Warehousing Institute ofrece educación, capacitación, certificación, noticias, e investigación y tecnologías de la información para ejecutivos. http://tdwi.org/
6. Bill Inmon. Considerado por muchos el padre del Data Warehouse.
7. Ralph Kimball. Considerado el principal promotor del enfoque dimensional para el diseño de almacenes de datos.
8. Nociones de Data Warehousing. http://tdwi.org/
9. Libro “Evaluación y Decisión Multicriterio - reflexiones y experiencias”. EDUARDO MARTÍNEZ Y MAURICIO ESCUDEY. Editorial Universidad de Santiago/UNESCO, Santiago de Chile, 1998.

Sistema de interacción con la ECU de un automóvil empleando procesamiento de lenguaje natural

Martín Agüero, Maximiliano Colman, Fabrice Dubois,
Marcelo Rabadán, Santiago Amarilla

Universidad de Palermo, Facultad de Ingeniería, Buenos Aires, Argentina
aguero.martin@gmail.com, maximilianocolman@gmail.com, kerygme@riseup.net,
mrabadan@dilmi.com.ar, amarilla.santiago@gmail.com

Resumen. En este trabajo se realiza una breve revisión general acerca de las oportunidades y desafíos que presenta la intención de la industria de sumar a Internet objetos de la vida cotidiana. Esta tendencia denominada Internet de las Cosas, implica cambios en el modo como las personas interactuarían con esos objetos-dispositivos-cosas. En base a la disponibilidad de datos a través de la interfaz OBD2 (On Board Diagnostics 2) presente en todos los automóviles y la masividad de las redes sociales, se propone un sistema que permita a las personas interactuar con su auto empleando lenguaje coloquial y a través de Twitter. El prototipo convierte las consultas en comandos que son dirigidos a la ECU (Electronic Control Unit) y la respuesta es traducida a lenguaje humano no experto. Empleando técnicas de procesamiento de lenguaje natural (NLP) y un sistema experto, se confirma la viabilidad de estas herramientas para crear interfaces más humanas para los dispositivos. Para una siguiente etapa, se planea incorporar análisis por distancia semántica y comparación de estructura sintáctica con patrones locales.

Palabras clave: osgi, java, ecu, obd2, nlp, iot, siot, expert system.

1 Introducción

En las redes sociales, las personas comparten opiniones, emociones y todo tipo de contenidos en ámbitos mayormente de acceso público. En la vida cotidiana las personas no solo interactúan con otras personas sino que también lo hacen con los objetos que las rodean, como por ejemplo, libros, autos, películas y todo tipo de dispositivos en general. En la emergente Internet de las Cosas (IoT - Internet of Things), esos dispositivos comenzarán a comunicar tanto su estado como funcionalidad. A través de esos mensajes, un electrodoméstico podrá avisar cuando completó su trabajo o el estéreo del auto comunicará a los seguidores del conductor qué tema está escuchando [1]. El paradigma IoT agrega nuevas dimensiones a la conectividad: objeto-a-objeto, humano-a-objeto y negocio-a-objeto. Estos dispositivos, que poseen capacidad de sensar, actuar y procesar datos, definirán un entretejido omnipresente que modificará el modo como la sociedad interactúa con ellos [2]. Esta realidad está dando lugar a la creación de una IoT Social (SIoT) donde esas *cosas* sociabilizan [3]. El trabajo de Mendes [4] introduce la idea de objetos con

capacidad de participar en conversaciones donde anteriormente sólo los humanos formaban parte. Está claro que ese diálogo entre humanos y objetos deberá ser a través de mensajes en lenguaje natural y coloquial.

El proceso de comprender el significado de esos mensajes, no sólo estará basado en modelos determinísticos o sintácticos. Ese análisis también requerirá de técnicas semánticas para evaluar en contexto cada comunicación de modo tal que esos dispositivos sean capaces de interpretar y responder a otros mensajes emitidos por humanos y otros dispositivos.

Tomando como referencia la arquitectura propuesta para servidor de SIoT [5], este proyecto tiene como objetivo diseñar un prototipo de middleware cuya función principal será la de interpretar consultas en lenguaje humano coloquial y traducirlas en solicitudes a un dispositivo de uso cotidiano. A su vez, las respuestas emitidas por el objeto, serán traducidas a lenguaje humano de modo tal que puedan ser comprendidas por un usuario no experto.

Dado que todos los automóviles actuales están equipados con al menos una unidad de control electrónico (ECU – Electronic Control Unit) que gestiona las principales funciones del motor, está conectada a un gran número de sensores y ofrece como interfaz el estándar OBD2 (On Board Diagnostics 2), se seleccionó al automóvil como dispositivo-cosa-objeto sobre la cual realizar la prueba de concepto. Es por ello que actualmente los vehículos, equipados con tecnología wireless, ya pueden ser considerados como dispositivos IoT móviles [6].

El prototipo se probó con éxito en una SBC (Single-Board Computer) Raspberry Pi conectada a la ECU de un automóvil mediante un scanner de interfaz OBD2 y acceso a Internet por red celular. El canal de comunicación es la red social Twitter.

A continuación, la sección 2 describe brevemente las principales características de la tecnología seleccionada. Luego, la sección 3 describe el prototipo desarrollado. En la sección 4 se presentan posibles casos de uso cotidianos y en la sección 5 se mencionan las actividades de transferencia a la industria efectuadas a partir de este proyecto. Finalmente, la sección 6 presenta las conclusiones y trabajos futuros propuestos.

2 Tecnologías

A continuación se describen brevemente las tecnologías que forman parte del proyecto:

2.1 Plataforma de software: OSGi

A finales de los años 90 la industria detectó la necesidad de contar con un framework que permitiera interconectar pequeños dispositivos para domótica y creó la OSGi Alliance [7]. Este consorcio de empresas surgió como iniciativa marco para el desarrollo de especificaciones que promovieran la estandarización del software a partir de la modularización de sus componentes, facilitando así la reutilización y buscando un equilibrio en el acoplamiento entre sus elementos. OSGi también define un ciclo de vida para sus elementos con el principal propósito de permitir cambios en

sus enlaces, sin la necesidad de detener el sistema completo. El alcance inicial de esta iniciativa se amplió a punto tal que hoy numerosos sistemas de aplicación general han adoptado la especificación. Un software muy difundido y que implementa OSGi como tecnología troncal, es el entorno de desarrollo integrado (IDE) Eclipse, donde cada plugin es un módulo OSGi.

Con el advenimiento de la IoT y por la natural adaptabilidad a dispositivos de limitada capacidad de procesamiento, OSGi resultó ser ideal para el ecosistema IoT. Esta situación derivó en una serie de estándares y frameworks orientados de manera específica a esta nueva tecnología como la comunidad open source IoT dentro de la Fundación Eclipse [8].

2.2 Conectividad por red celular

En la actualidad las redes celulares poseen la capacidad de ofrecer diferentes grados de calidad de servicio dependiendo del nivel de conectividad en servicio de la zona. Las redes celulares y tecnologías disponibles actualmente son: GPRS (80 kbps), EDGE (236Kbps), 3G (2Mbps), HDSPA (14Mbps) y 4G LTE (24Mbps). El módulo GSM de los celulares tiene la capacidad de hacer soft-handover (traspaso de celdas en caliente). Este algoritmo de traspaso se basa en medir la intensidad de señal en la estación base actual y en las cercanas, al pasar un umbral de baja señal, la estación base pasa la comunicación a la nueva y le informa al módulo GSM a través del canal de señalización de la nueva conexión, de esta forma es totalmente transparente para la aplicación que esté utilizando ese canal de datos [9].

2.3 Plataforma de hardware: Raspberry Pi

La Fundación Raspberry es una organización de caridad. Su objetivo es promover el estudio de la ciencia computacional especialmente en las escuelas. El objetivo del Pi fue de crear una pequeña computadora que fuera accesible para cada usuario final. Importante esfuerzo e investigación se puso en el Raspberry Pi, manteniendo los costos a un mínimo removiendo componentes que no eran críticos. El microprocesador para el Pi 2 es el Broadcom BCM2836. Es un system-on-a-chip, tiene CPU y gráficos con 1080p nativo dentro del chip [10]. Uno de los sistemas operativos más empleados en esta computadora es el Raspbian que está basado en Debian y es una distribución de Linux.

2.4 Unidad de Control Electrónico (ECU) a través de OBD2

La Unidad de Control Electrónico o ECU (por sus siglas en inglés: Electonic Control Unit), son unidades de procesamiento que se encuentran presentes en equipos de inyección que gestionan el funcionamiento del motor. Son, por lo general, parte de una red de unidades electrónicas presentes en un vehículo, entre las cuales pueden o no compartir información a partir, de una o varias redes de comunicación. Controlan, en base a lectura de sensores, el funcionamiento del motor variando la apertura de

válvulas, accionamiento de bobinas o apertura de inyectores. Esta gestión electrónica trata mantener la estequiometria de un motor y de esa forma se obtienen emisiones de gases reducidos, potencias elevadas incluso con motores de pequeña cilindrada y considerables aumentos en autonomía.

Con el constante avance la tecnología, los desafíos para diagnóstico de fallas fue creciendo a la par con el avance de las ECU; por lo que las herramientas para diagnostico fueron siendo cada vez más complejas. A modo de estandarizar el acceso a esa información, la industria automotriz acordó desarrollar el protocolo OBD (On Board Diagnostic, por sus siglas en ingles), el cual permite, a través de una interfaz de hardware estándar, verificar el estado de los sensores involucrados en controlar emisiones, así como también emitir alertas de fallas [11].

2.5 Procesamiento de Lenguaje Natural y Sistema Experto

El procesamiento de lenguaje natural (NLP – Natural Language Processing) es un área de la inteligencia artificial que se especializa en la generación automática e interpretación del lenguaje humano [12]. Para realizar este procesamiento se debe contar con herramientas que permitan hacer análisis morfológico, lematización, etiquetado Part-of-Speech (PoS) y desambiguación en texto. El proyecto open source FreeLing [12] ofrece estas capacidades con muy aceptable precisión para el idioma español.

Los Sistemas Expertos habitualmente son empleados como herramientas para la toma de decisiones y también son considerados parte de la inteligencia artificial [14]. Estos sistemas imitan el proceso de razonamiento que los expertos utilizan para resolver problemas específicos. Una implementación muy difundida entre la comunidad open source es JBoss Drools que emplea y extiende el algoritmo Rete [15].

2.6 Medio de comunicación Twitter

Una de las opciones para acceder en a la API de la red social Twitter y a través de Java es con la librería[1] open source Twitter4J [16]. Este software actúa de interfaz simplificada a los servicios de Twitter desde un programa Java. Permite realizar siguientes funciones: autenticación, envío y recepción de tweets, gestión de usuarios, listas, mensajes directos, suscripciones y otros. Ofrece acceso transparente a tres APIs de Twitter: Streaming API, REST API y Search API.

3 Prototipo

A partir de la información disponible en la ECU (Electronic Control Unit) de los automóviles y a través del estándar OBD2 (On Board Diagnostics 2), se integra un sistema de reconocimiento de lenguaje humano que pueda interpretar las solicitudes y

[1] Es una mala traducción de la palabra ‘libraries’ la correcta sería ‘bibliotecas’

responder de forma amigable y fácil de comprender por cualquier persona. El sistema se ejecuta en una mini computadora Raspberry Pi (RPI) ubicada dentro del automóvil y se conecta a Internet a través de la red celular (3G). El canal de comunicación es la red social Twitter. En la Fig. 1 se muestra un diagrama general de los componentes principales del prototipo, donde se representa en contexto el despliegue de la solución.

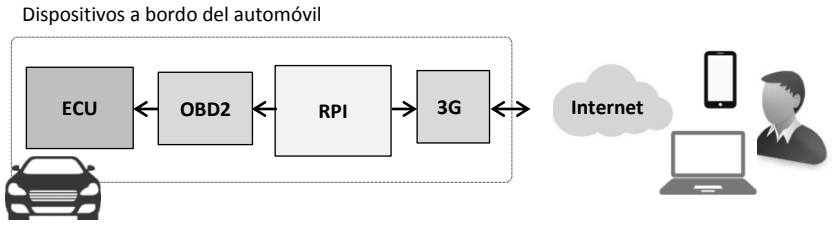


Fig. 1 - Diagrama general

La computadora RPI es la que recibe a través de Twitter las consultas y las traduce en comandos dirigidos a la ECU del automóvil. Prácticamente todo el software se ejecuta en la RPI, no obstante, el procesamiento de lenguaje natural es delegado a un Web Service publicado en la nube. Debido a que FreeLing es software que requiere de prestaciones que superan las capacidades de la RPI, se decidió desarrollar un Web Service (WS) que encapsule la funcionalidad de procesamiento de lenguaje natural (ver Fig. 2).

El software es un conjunto de 5 módulos OSGi donde un módulo Principal atiende y coordina solicitudes recibidas vía Twitter.

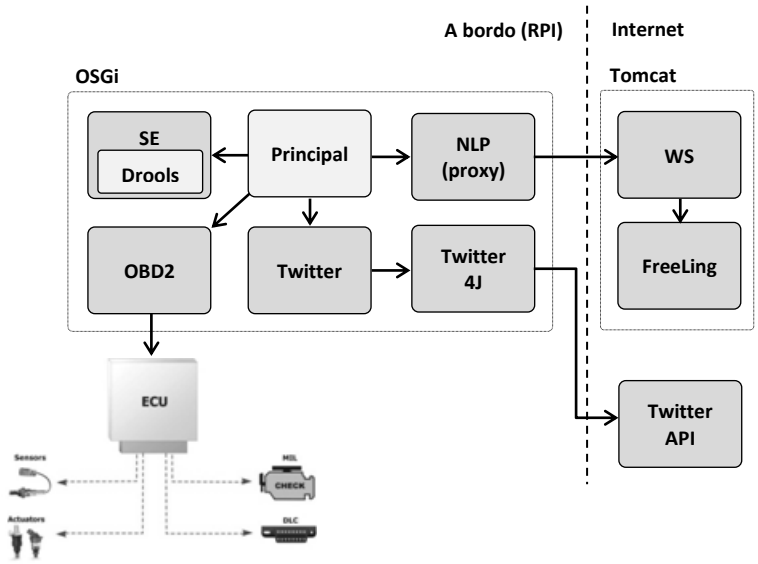


Fig. 2 - Componentes de software del prototipo

Principal: Coordina la ejecución de cada consulta.
Twitter: Accede a través del módulo Twitter4J a la API de Twitter.

Twitter4J: Consulta si existen nuevos mensajes donde se mencione la dirección asignada al automóvil (@irc_up_2015_iot[2]).
NLP: Envía a FreeLing todo el texto del mensaje y recibe los tokens, el etiquetado de cada token (PoS) y la lemmatización de cada uno.
SE: Reglas de Drools Expert analizan el resultado de NLP con el fin de determinar qué dato deberá ser leído a través de OBD2.
OBD2: Consulta a la ECU y devuelve el resultado.

3.1 Secuencia de ejecución

1 Una vez detectado un nuevo mensaje, el módulo principal obtiene el texto completo desde el módulo Twitter.
2 La cadena se envía al módulo NLP que este a su vez la reenvía al WS de FreeLing.
3 Con el resultado de FreeLing, el módulo principal lo envía al módulo SE donde en función de la estructura sintáctica y la presencia de palabras clave se construye dinámicamente la consulta al módulo OBD2.
4 El módulo OBD2 accede a través de la librería Java Simple Serial Connector [17] al puerto USB correspondiente al Scanner OBD2 y transmite la consulta.
5 La respuesta obtenida de la ECU es traducida a lenguaje no técnico por el módulo SE.
6 El módulo Principal envía la cadena resultante al módulo Twitter que este a su vez publica la respuesta anteponiendo la dirección del usuario (@usuario) que realizó la consulta.

Cabe destacar que todo el software de terceros es open source.

3.2 Entornos de desarrollo y pruebas

Una vez que se contó con una versión ejecutable de los módulos, se montó un entorno de pruebas (ver Fig. 3).

[2] https://twitter.com/irc_up_2015_iot/with_replies

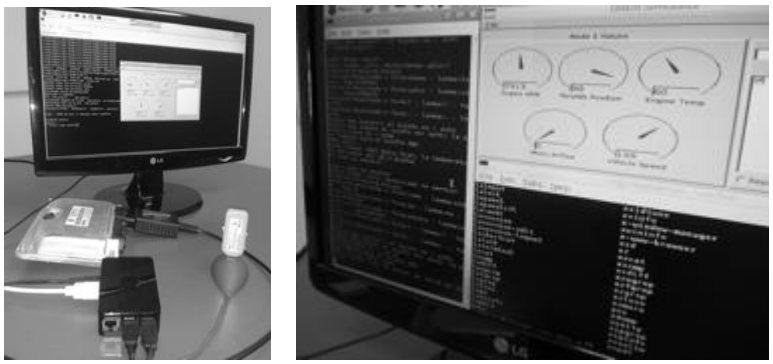


Fig. 3 - Entorno de pruebas

Desde allí se probó el software en la computadora RPI y ejecutando el simulador OBDsim [18]. Luego, como se observa en la Fig. 4, se probó conectado los componentes (Scanner OBD2, RPI y Módem 3G) a dos automóviles.

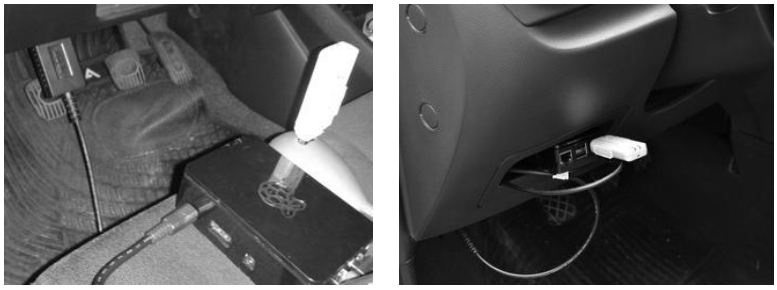


Fig. 4 – Pruebas en Nissan Tiida y VW Gol Trend

7 Casos de uso

La versión inicial del prototipo permite realizar un set limitado de consultas (ver Tabla 1), no obstante es suficiente para validar el concepto.

Tabla 1 - Consultas implementadas

Dato	Consulta
Si hay algún código de error activo (DTC[3])	"Cómo te sentís?", "Cómo estás de salud?"
Motor encendido (RPM)	"Cómo tenés el motor?", "Está encendido el motor?"
Temperatura motor (Coolant temperature)	"Tenés calor?"
Posición acelerador (Throttle position)	"Cómo te tratan?", "Cómo la pasás?"
Posición geográfica (GPS)	"Por dónde andás?", "Cuál es tu ubicación?"

[3] Diagnostic Trouble Codes

Para el caso de la consulta acerca de la ubicación actual, el software solicita a Google Maps una imagen con la posición geográfica y esta es incrustada en el mensaje de respuesta (ver Fig. 5).

Fig. 5 – Respuesta a ubicación con mapa incrustado

8 Transferencia a la industria

Conseguir el apoyo de la industria es el siguiente objetivo del proyecto, por eso se ha presentado en una charla organizada por la CESSI (Cámara de Empresas de Software y Servicios)[4] donde se estableció contacto con una empresa vinculada a la industria automotriz. Actualmente se está en conversaciones para firmar un convenio de colaboración. Asimismo también se postuló el proyecto para el concurso Innovar 2015 organizado por el Ministerio de Ciencia, Tecnología e Innovación Productiva[5]. Se está aguardando novedades respecto al proceso de evaluación.

9 Conclusiones y trabajos futuros

Este trabajo tiene como objetivo crear un prototipo que permita definir bases para avanzar en la creación de una plataforma de integración de dispositivos en Internet. Asimismo el proyecto también propone una interfaz de usuario de lenguaje natural en lugar de botones o comandos. Para ello se seleccionó un conjunto de tecnologías de

[4] http://www.cessi.org.ar/ver-noticias-charla-gratuita-sobre-internet-de-las-cosas-y-oportunidades-de-negocios-1852

[5] ID 18007 Sistema de interacción con la ECU de un automóvil empleando lenguaje natural

soporte a la solución que son OSGi, FreeLing (NLP), Drools (Sistema Experto) y OBD2 como interfaz al dispositivo objetivo (ECU del automóvil). También se emplearon la red celular y Twitter como medios de transmisión y comunicación.

Los resultados del proyecto permiten confirmar la viabilidad de un sistema experto y software NLP para convertir el lenguaje natural en comandos interpretables por un dispositivo y vice versa. Asimismo también se puso a prueba la compatibilidad de OSGi en un entorno de prestaciones reducidas, como es la computadora Raspberry Pi, y el desafío que implica consumir web services (FreeLing y Twitter) a través de una red celular de datos en movimiento.

Como siguiente fase del proyecto se planea ampliar y mejorar la capacidad de interpretar consultas mediante el análisis de la distancia semántica entre palabras y la comparación de la estructura sintáctica con patrones locales.

10 Referencias

1. Kranz, M., Roalter, L., Michahelles, F.: Things That Twitter: Social Networks and the Internet of Things. ETH Zurich (2010)
2. Amir, M., Hu, F., Pillai, P., Cheng, Y., Bibiks, K., Interaction Models for Profiling Assets in an Extensible and Semantic WoT Framework. 10th International Symposium on Wireless Communication Systems. VDE Verlag (2013)
3. Alam, K., Saini, M., El Saddik, A.: Towards Social Internet of Vehicles: Concept, Architecture, and Applications. IEEE (2015)
4. Mendes, P., Social-driven Internet of Connected Objects. Interconnecting Smart Objects with the Internet Workshop (2011)
5. Atzori, L., Iera, A., Morabito, G., Nitti, M., The Social Internet of Things (SIoT), when social networks meet Internet of Things: Concept, architecture and network characterization. Elsevier (2012)
6. Oka, D., Furue, T., Langenhop, L., Nishimura, T., Survey of Vehicle IoT Bluetooth Devices. 7th International Conference on Service-Oriented Computing and Applications. IEEE (2014)
7. OSGi Alliance, http://www.osgi.org/
8. Eclipse IoT, http://iot.eclipse.org/
9. Brunner, C., Caravaglia, A., Mittal, M., Narang, M., Bautista, J.: Inter-System Handover Parameter Optimization. Vehicular Technology Conference. IEEE (2006)
10. RasPi Magazine, Issue 11 (2015)
11. Diseño y Prueba de ECU usando Productos de National Instruments, http://www.ni.com/white-paper/3312/es/
12. Hirschberg, J., Ballard, B., Hindle, D., Natural Languaje Processing. AT&T Technical Journal (1988)
13. Padró, L., Collado, M., Reese, S., Lloberes, M., Castellón, I., FreeLing 2.1: Five years of open-source language processing tools. 7th Language Resources and Evaluation Conference (2010)
14. Rossini, P., Using Expert Systems and Artificial Intelligence for Real Estate Forecasting. 6th Annual Pacific-Rim Real Estate Socienty Conference (2000)
15. JBoss Drools, http://www.drools.org/
16. Twitter4J, http://twitter4j.org/
17. Java Simple Serial Connector, https://code.google.com/p/java-simple-serial-connector/
18. OBDSim, http://icculus.org/obdgpslogger/obdsim.html

EnMa Tool: una herramienta de software en apoyo al estándar ISO 50001 con basamento en Indicadores Clave de Desempeño para la Búsqueda de Eficiencia Energética en Procesos Productivos de PyMEs

Leopoldo Nahuel[1], José Maccarone[2], Javier Marchesini[1],
Marcela Rognoni[1], Gastón Andres[1], Rodrigo María García[1]

[1] Departamento de Ingeniería en Sistemas de Información
[2] Departamento de Ingeniería Eléctrica
Facultad Regional La Plata, Universidad Tecnológica Nacional, Av. 60 esq. 124 s/n CP 1900
{lnahuel, jmacarone, jmarchesini, mrognoni, gandres, rmariag}@frlp.utn.edu.ar

Abstract. En el presente trabajo se analizan las principales causas de ineficiencia energética y se busca definir Indicadores Clave de Desempeño (Key Performance Indicator, KPI por sus siglas en inglés) para su implementación en PyMEs industrializadas de cualquier tipo, con la finalidad de alcanzar la eficiencia energética en los procesos productivos que en éstas se realizan. Se siguen los lineamientos propuestos por la norma internacional ISO 50001, para el diseño de un Sistema de Gestión de la Energía (SGE), con el objetivo de que cualquier organización pueda mejorar su eficiencia energética. Como aporte al campo de la informática se presentará el desarrollo de un software llamado EnMa Tool, una herramienta de soporte a la implementación de Sistemas de Gestión de la Energía bajo norma ISO 50001 en Pequeñas y Medianas Empresas (PyMEs).

Palabras Clave: Indicadores Clave de Desempeño, Sistema de Gestión de la Energía, Herramientas software, Ahorro y Eficiencia Energética.

1. Introducción

Durante los últimos 10 años el consumo industrial global ha estado aumentando significativamente. En el año 2006 se consumieron 51.275 ZW (Miles de trillones de Watts de potencia) y se espera que dicho número crezca a 71.961 ZW para el 2030, con una tasa promedio de crecimiento anual de 1.4%. El sector industrial utiliza más energía que cualquier otro sector de uso final, y actualmente consume cerca del 37% de toda la energía distribuida en el mundo. [1]
En la figura 1 se puede apreciar este crecimiento del consumo energético global del sector industrial, detallando tanto países incluidos o no en la OECD (Organización para la Cooperación y el Desarrollo Económicos).

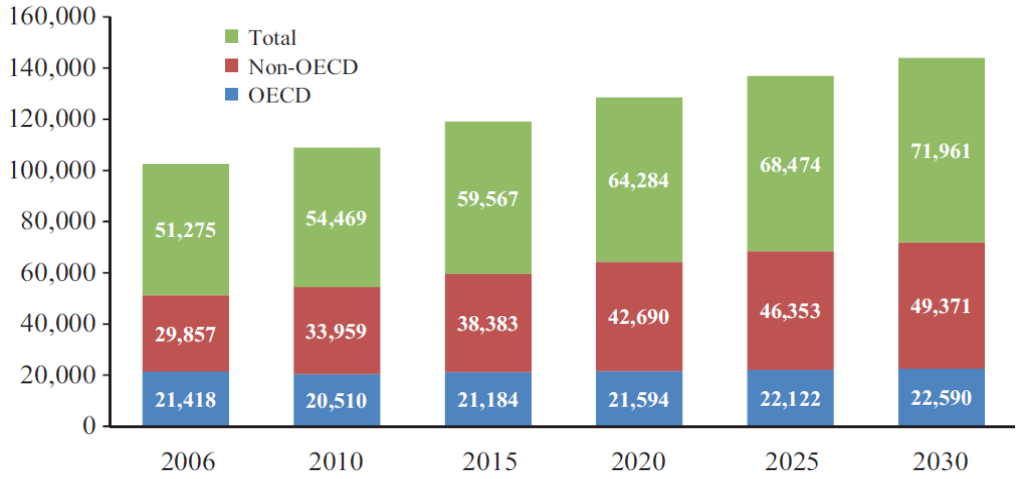


Fig. 1. Consumo de energía del sector industrial entre los años 2006 y 2030 (unidades en ZW: miles de trillones de Watts).

Esta energía es consumida por un diverso grupo de industrias incluyendo manufactura, agricultura, minería y construcción, y por una gran variedad de actividades, como procesamiento y ensamblado, calefacción y refrigeración de recintos, e iluminación.

Si bien existen varias tecnologías para el ahorro de energía, como el uso de motores de alta eficiencia, variadores de velocidad, y sistemas para prevención de fugas, se enfocará el estudio en los Sistemas de Gestión Energética asistidos por herramientas software, dado que se considera que su implementación es más rápida y efectiva, transversal a cualquier tipo de industria y con un menor costo asociado.

Se buscará identificar los principales causales de ineficiencia en una industria, y definir Indicadores Clave de Desempeño Energético, para luego poder realizar un seguimiento que permita identificar desvíos respecto de los valores previamente definidos como objetivos de ahorro.

El estándar ISO 50001 define estos Indicadores Claves de Desempeño como Indicadores de Desempeño Energético, o EnPIs por sus siglas en inglés, y representan un valor de medida del rendimiento energético, el cual debe ser definido por cada organización.

Un SGE que cumpla ISO 50001 debe poder demostrar una mejora en el desempeño energético, por lo cual debe poder medirse este desempeño antes y después de implementar el SGE para comprobar si es eficaz y en qué medida lo mejora.

Debe tenerse en cuenta que el desarrollo industrial en todo el mundo resultará en un mayor consumo de energía y llevará a una mayor concentración de gases de efecto invernadero como el dióxido de carbono, dióxido de azufre y monóxido de nitrógeno, los cuales tienen consecuencias desastrosas para el clima del planeta, como el recalentamiento global, sequía e inundaciones [3]. Además los costos asociados al uso de energía se incrementan permanentemente, lo cual empuja a buscar la eficiencia en el consumo de los recursos energéticos.

2. Requerimientos para un Sistema de Gestión de la Energía con Soporte a la Norma ISO 50001

El punto de partida para encarar un programa de eficiencia energética y de mejora continua es detectar los objetivos que se buscará alcanzar con el mismo. El estándar ISO 50001 exige establecer, documentar, implementar y mejorar continuamente, un sistema de gestión energética. El primer paso en la implementación de un sistema de gestión energética es definir el alcance y frontera de la organización. La organización debe considerar: el uso de energía a través de todo su proceso productivo, las oportunidades de mejora en cuanto a eficiencia y los recursos humanos para soportar un SGE [4]. El desarrollo de una herramienta software de soporte para un SGE debería cubrir o respetar estos requerimientos para estar en línea con la norma ISO 50001.

- *Política energética.*
- *Planeamiento energético.*
- *Requerimientos legales.*
- *Revisión energética.*
- *Equipamiento, tipo de planta y tipo de industria.*
- *Variables afectando a los usuarios significativos.*
- *Estimación de consumo a futuro.*
- *Oportunidades de eficiencia energética.*
- *Línea base e indicadores de desempeño energético.*
- *Objetivos energéticos, metas y planes de gestión de acciones energéticas.*
- *Implementación y operación.*
- *Metodología de comunicación interna.*
- *Documentación: alcance y frontera, política energética, objetivos energéticos, metas y planes de acción.*
- *Control operacional.*
- *Obtención de servicios de energía, productos, equipamiento y energía.*
- *Monitoreo, medición y análisis.*
- *Auditoría interna del SGE: requerimientos, metas, y mejora del desempeño energético.*
- *Disconformidades, corrección, acciones preventivas y correctivas.*
- *Control de registros.*

3. Una Clasificación de las Ineficiencias Energéticas Detectadas

En base a un relevamiento realizado sobre parques industriales locales entre los años 2012 y 2013, se ha estudiado el desempeño energético de varias organizaciones, y en particular se han analizado las condiciones de contratación de los proveedores de energía eléctrica en dichas industrias.

En dicho estudio se han detectado ocho cuestiones, entre necesidades e ineficiencias energéticas, las cuales se detallan a continuación:

1. *Ineficiencia por Factor de Potencia bajo:* el factor de potencia tiene un importante significado técnico-económico, debido a que de su magnitud dependen, en cierta medida, los gastos de capital y explotación, así como el uso efectivo de los equipos de las instalaciones eléctricas. Constituye un indicador cualitativo y cuantitativo de la eficiencia energética en la industria. Todas estas cuestiones tienen una consecuencia negativa y finalmente generan considerables sobrecostos en la facturación del servicio que la organización debe afrontar [5].
2. *Ineficiencia por contratación inadecuada:* una problemática muy común en las organizaciones estudiadas es que efectúan una contratación errónea del servicio de energía eléctrica, contratando capacidades por encima de lo necesario. Este error en general ocurre por falta de conocimiento, o por no contar con las herramientas adecuadas para dimensionar el verdadero consumo de la organización
3. *Ineficiencia por picos de demanda:* los picos de consumo pueden acarrear severas multas y castigos económicos por parte del proveedor del servicio, los cuales pueden prolongarse por los siguientes períodos, aún si los picos no vuelven a ocurrir. Esto conlleva un sobrecosto a pagar que podría ser evitado si se organizan mejor los procesos productivos de cada industria para asegurar que no se incurra en estas demandas excesivas simultáneas, que producen picos por encima de las capacidades contratadas.
4. *Ineficiencia por maquinarias en espera (standby):* muchas organizaciones olvidan considerar los consumos de las maquinarias cuando están en espera (activas, pero no realizando tareas productivas) ni aquellos períodos de demora en los que se está esperando que termine una etapa inmediata anterior de la cadena de producción, que por algún motivo no estuvo lista justo a tiempo para alimentar la entrada del proceso en cuestión. El consumo de una máquina desconectada es cero, pero aquella que está en espera tiene un consumo efectivo mayor, y los tiempos de espera deben ser minimizados para corregir esa ineficiencia.
5. *Necesidad de KPI prácticos, aplicables a nivel de maquinaria, proceso y planta:* es necesario desarrollar métodos que puedan extender el uso de Indicadores de Desempeño Energético (de ahora en adelante EnPIs) a otros niveles de agregación, adaptables a maquinarias, procesos específicos y para cada tipo de planta.
6. *Necesidad de comparación de eficiencia energética entre plantas:* los EnPIs desarrollados deben permitir la comparación entre varias plantas.
7. *Necesidad de sensores inteligentes de tiempo real, de bajo costo:* se debe buscar el desarrollo de herramientas con compatibilidad con tecnologías de medición por sensores inteligentes, de bajo costo y consumo intrínseco de energía.
8. *Dificultad para comprender las relaciones de causa-efecto entre la gestión de la producción y el rendimiento energético asociado:* una parte importante del proceso es identificar las relaciones de causa-efecto entre la gestión de los procesos productivos y la forma en que esto impacta en el consumo de energía.

4. Definición de EnPIs (Indicadores de Desempeño Energético)

Teniendo en cuenta el tipo de industria, las características de la planta y los diferentes procesos productivos que se realizan, se podrán diferenciar entre cuatro tipos distintos de EnPIs:

4.1. EnPIs a Nivel de Organización

Estos indicadores se basan en el consumo total de una organización. Sobre éstos, se calculan los ahorros energéticos año a año y su progreso contrastado con las metas definidas en su Política Energética.

4.2. EnPIs de Consumo de Combustibles Fósiles

Se trata de los indicadores referidos al consumo de combustibles fósiles, en todas sus formas como carbón, gas natural, fuel-oil, comúnmente utilizados en la industria, tanto para procesos químicos y metalúrgicos, como para calefacción de recintos, según los requerimientos de los procesos productivos.

4.3. EnPIs de Consumo de Combustible para Transporte

En los casos en donde las organizaciones incorporen el transporte intensivo y la distribución en sus procesos productivos, se deben definir EnPIs adecuados para su medición y registro.

4.4. EnPIs de Consumo de Electricidad

Estos indicadores se enfocan especialmente en el consumo de electricidad, y son objeto principal de este estudio por ser considerados los de mayor influencia en el ahorro de energía que puede implementar una organización en el corto plazo. Con un mayor potencial para reducir los costos rápidamente y mejorar la eficiencia energética, y siendo el campo en el que se ha detectado una mayor cantidad de ineficiencias, se considera el elemento clave para introducir mejoras en el desempeño energético de las organizaciones.

1. **FP = Coseno (Φ)** donde Φ: ángulo de desfase entre tensión y corriente
   Factor de Potencia: el valor del Coseno de (Φ) puede variar entre 0 y 1. Valores más cercanos a 1 indican una mayor eficiencia (un menor uso de energía reactiva).
   a. FP por máquina: el factor de potencia puede ser corregido para cada máquina individualmente, logrando una reducción en conjunto igual de efectiva.
   b. FP por proceso productivo: al identificar aquellos procesos productivos que mayor influencia tienen en el aprovechamiento de la energía eléctrica, se pueden enfocar adecuadamente las acciones correctivas pertinentes.
   c. FP por franjas horarias: se debe considerar las franjas horarias en las que es más costosa la energía eléctrica y verificar que un factor de potencia bajo no produzca un exceso por encima de la capacidad contratada en horarios pico de consumo.
2. **EC = kW/CU** donde CU: cantidad de unidades producidas, W: potencia activa [watts]
   Eficiencia del Consumo: un mayor valor de ésta medida indicará una mayor eficiencia en el consumo de energía utilizada por cada unidad producida o instancia de servicio brindado.
   a. EC por máquina: mejorar la eficiencia del consumo para cada maquinaria o equipo permite mejorar el desempeño energético prestando especial atención al rendimiento de la producción. Se deberán evaluar los tiempos de standby o reposo, para minimizar los desperdicios de energía.
   b. EC por franja horaria: siempre será más eficiente producir en aquellas franjas horarias con menor costo asociado al consumo energético.
3. **DCC = CC – CP** donde CC: capacidad contratada, CP: consumo instantáneo.
   Desviación entre capacidad contratada y consumo: este indicador se conforma calculándolo a intervalos regulares de tiempo, y brinda información sobre la adecuada contratación del servicio eléctrico, y sobre la posibilidad de expandir la capacidad productiva instalada dentro de los márgenes de energía disponibles según lo contratado.

5. EnMa Tool: Herramienta de Software basada en ISO 50001 para Sistemas de Gestión Energética

EnMa Tool es una herramienta software diseñada para dar soporte a la normativa ISO 50001 en etapas concretas de los procesos que propone, de modo que su objetivo principal será brindar información relevante generada a partir de datos sobre el consumo energético de la organización. [6,7]. El funcionamiento de EnMa se basa en el cómputo de los datos de consumo eléctrico. A medida que se vayan cargando se irán almacenando en una base de datos y estarán disponibles cada vez que se los requiera. En base a estos datos, la herramienta proporcionará informes que asistirán al personal idóneo a tomar decisiones.
La fuente de información para el posterior procesamiento y armado de reportes de EnMa, son las facturas de consumo eléctrico de la organización. Esta carga se hace a través de dos pantallas: una sirve para la carga de los datos de identificación de la factura, como la fecha o el número. Además se ingresan los datos netos de consumo, subsidios, y otros (Figura 2).

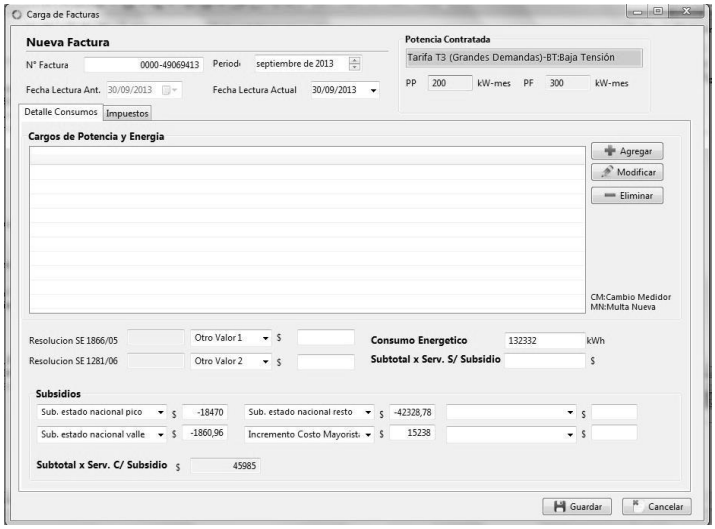


Fig. 2: Ventana para ingreso de datos principales de factura

La otra pantalla es específica para la carga de datos impositivos correspondientes al usuario y su tipo de contratación de servicio (Figura 3). Además de los impuestos fijos (pueden ser configurados a nivel global en la aplicación) también se permite la carga de otros de carácter temporal o específicos de la situación fiscal de la empresa.

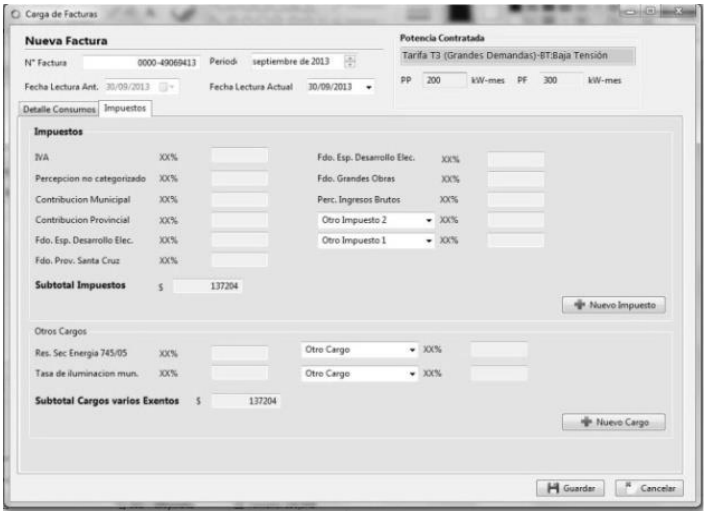


Fig.3: Ventana para ingresar información impositiva relevada de la factura

En la Figura 4 se observa un gráfico de consumo energético desarrollado sobre varios períodos y separados por el momento de consumo (resto, valle y punta), a modo de ejemplo.

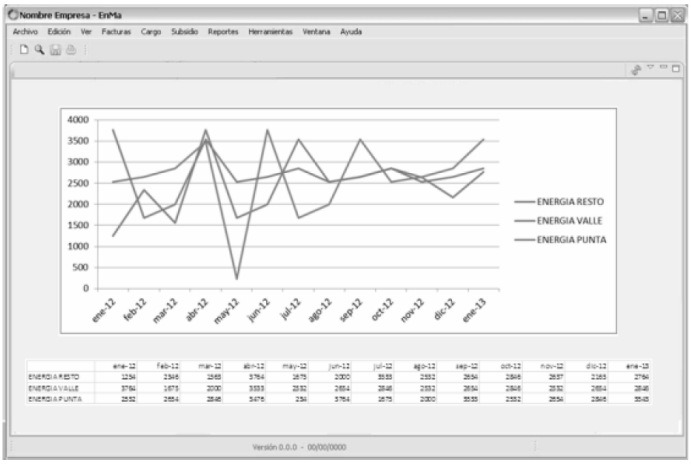


Fig. 4: Ejemplo de salida (reporte gráfico)

6. Conclusiones y Trabajo Futuro

El panorama del uso de energía actual nos indica una fuerte tendencia al aumento del consumo y los costos operativos. Ante ésta situación se torna indispensable la búsqueda de acciones preventivas y correctivas que lleven a mejorar el desempeño energético de las distintas organizaciones. La reducción del consumo de recursos energéticos puede cambiar el futuro de la situación medioambiental, y esto se puede lograr rápidamente si cada organización analiza su propio desempeño energético, y redefine el uso de sus recursos.

Luego de analizar la situación en distintas industrias, se han identificado las principales causas de ineficiencia energética, lo cual ha permitido definir Indicadores Clave de Desempeño que permitan realizar un seguimiento estadístico, buscando mediante su corrección, mejorar la eficiencia energética en las organizaciones, sobre todo aquellas con uso intensivo de energía.

El aporte al campo tecnológico con la herramienta EnMa Tool, busca complementar el desarrollo de un Sistema de Gestión de la Energía bajo norma ISO 50001, cubriendo las necesidades básicas para el gerenciamiento energético, con la posibilidad de incorporar en la brevedad el uso de los EnPIs definidos en esta primer etapa. Actualmente posee indicadores para el consumo de electricidad, como lo son el factor de potencia, eficiencia de consumo e información sobre la desviación entre capacidad contratada y consumo. Esta información se obtiene con la carga de facturas y se generan reportes gráficos que permiten hacer un análisis de negocio para la toma de decisiones. Así se pueden obtener datos anuales de ahorro energético, es decir, utiliza indicadores a nivel organización. EnMa está orientada a la gestión y a la toma de decisiones, y no a la automatización como otras herramientas más sofisticadas, debido a que éstas requieren de funciones de mayor complejidad de implementación que tienen un elevado costo de desarrollo y un aun mayor gasto en implantación de los dispositivos hardware necesarios para esta funcionalidad. EnMa es una herramienta acotada y sencilla pero ha demostrado ser suficiente para verificar resultados y validar los métodos y el diseño empleado. Por último, en etapas más avanzadas del proyecto, se analizará la posibilidad de extender su alcance para la gestión de otros tipos de energía y/o recursos (gas, agua, etc.).

Referencias Bibliográficas

[1] A review on energy saving strategies in industrial sector; 2010; E.A. Abdelaziz, R. Saidur, S. Mekhilef.
[2] Norma ISO 50001 - Energy Management Systems - http://www.iso.org/iso/home/standards/management-standards/iso50001.htm
[3] Mahlia TMI. Emissions from electricity generation in Malaysia. Renewable Energy 2002;27:293–300.
[4] A structured approach for facilitating the implementation of ISO 50001 standard in the manufacturing sector; Bhaskaran Gopalakrishnan, Kartik Ramamoorthy, Edward Crowe, Subodh Chaudhari, Hasan Latif. Research Gate, September 2014.
[5] El factor de potencia y la eficiencia energética. Manuel J. Hernández, Juana M. Téllez Reinoso, artículo: http://www.cubasolar.cu/biblioteca/Energia/Energia29/HTML/articulo06.htm
[6] L. Nahuel, J. Maccarone, J. Marchesini, M. D Ambrosio, L. Cantallops. Métodos y Tecnología Informática aplicada al desarrollo de Sistemas de Gerenciamiento Energético en apoyo a ISO 50001 . XV° Workshop de Investigadores en Ciencias de la Computación - WICC 2013. ISBN 9789872817961. Paraná, Entre Ríos, Argentina
[7] ENMA Tool - producto del PID Desarrollo de Instrumentos de Relevamiento Energético

y Algoritmos necesarios para un Software de Gestión Energética de Organizaciones. Homologado por la Secretaría de Ciencia, Tecnología y Posgrado del Rectorado de la Universidad Tecnologica Nacional (UTN) con código ENIANLP3530 y financiado por SCyT de la UTN - Facultad Regional La Plata y SCTyP del Rectorado UTN.

WhatsINFO: a Gamified Mobile Application Promoting the Integration of Computer Science School Freshmen to the UNLP

Isabel Miyuki Kimura, Matías Brown Barnetche, Claudia Queiruga, Javier Díaz

LINTI (Laboratory of Research in New Information Technologies), Computer Science School, National University of La Plata, 50 and 120 (Second Floor), La Plata (1900), Argentina
{ikimura,mbrown}@linti.unlp.edu.ar, {claudiaq,javierd}@info.unlp.edu.ar

Abstract. This paper presents the issue facing university freshmen due to the complexity implied in acquiring new discourses, contents and habits of the institutional culture of which they aim at becoming a part. Online gaming and socializing through digital media are common among young people – they engage their interest and define new ways of expression and communication. Ubiquity is another requirement of the new generations, with smartphones being the most chosen mobile devices for communication and socialization.
In this context, we present the WhatsINFO mobile application, whose goal is to mitigate the divide existing between new students and university culture through a technological tool that is close to teenage culture. WhatsINFO is a gamified mobile application that accompanies and encourages communication and participation of National University of La Plata (UNLP) Computer Science School freshmen during their entry course. WhatsINFO was tested during the 2014 entry course of the Computer Science School of the UNLP with over a hundred students. This field test yielded encouraging results regarding its incorporation as a complementary device promoting a new learning space.

Keywords: University entry, digital natives, mobile devices, mobile application, Android, QR code, indoor positioning

1 Introduction

For students, entering university constitutes a complex stage that implies acquiring new discourses, contents and habits of the institutional cultura of which they aim at becoming a part (Walker V., 2012).

The passage between high school culture and university culture constitutes a series of transformations on the ways in which students traverse the academic and social world (Carli, S., 2007). Although these problems are normal and unavoidable, we notice that some students can solve this conflict and enter and remain in University, while others face conflictive situations that make it impossible to process and overcome this change.

Our current university students are "digital natives"[1], they have different perceptions and experiences, have developed a different logic and carry with them new ways of

[1] The term "digital native" was coined in the year 2001 by the American autor Marc Prensky and

learning, reading, generating knowledge and doing work. Their vision of the world is traversed by the use of ICTs (Information and Communication Technologies). Taking into consideration the pedagogical potential of information resources coming from ICTs and the technological skills of our current students, it is fundamental to provide an educational use of technological resources in the academic environment.

Mobile telephony has had significant impact on the daily lives of young students, inserting itself deeply in their social practices as it allows them to interrelate multiple communication, socialization and entertaining activities. It is emergently being introduced in academic activities (Druetta D. et al, 2011).

The WhatsINFO application is a mobile application close to teenage culture that complements the activities that the Computer Science School of the UNLP coordinates with its freshmen in order to introduce them into its institutional culture. WhatsINFO is proposed as a technological complement to the activities done within the framework of the entry course of the Computer Science School, giving value to the social and cultural practices of teenagers. WhatsINFO was tested on a group of students of the Computer Science School during the 2014 entry course, yielding encouraging results regarding its incorporation during this first stage as university students.

This paper is organized as follows: first, it explains the problems involved in the university entry process and characterizes current freshmen, afterwards, it presents WhatsINFO, the gamified mobile application that accompanies and encourages communication and participation among Computer Science School freshmen and the activities engaged in with the students using this tool. Finally, we will make some final comments as a conclusion to this work.

2 Background

2.1 Issues Concerning Entering University

The passage between high school culture and university culture constitutes a critical stage for young students, as this process of separation, transition and incorporation into a new social and academic world makes many students experience anxiety and difficulties that can become obstacles for their entry into university (Medrano L et al, 2008).

From the moment in which students enroll in university, they start to undergo learning experiences and relate with teachers, peers and the institution itself, i.e., they begin to incorporate the new "rules of the game".

In order to remain part of the institution, the university requires that students appropriate the institutional dynamics and a series of tools in a short period of time. Generally speaking, public universities are characteristic for their impersonal treatment of students, who are heterogeneous and "anonymous". These characteristics are valued by students –however, they denounce a lack of guidance throughout their academic life. Unlike in high school, where there was a sense of belonging and security, in university students must learn to manage their time and make decisions on their academic path (Carli S., 2012). Many higher education institutions devise plans to contribute to the development of change processes involved in the initiation as a

refers to the dominating worldwide demographic group that describes the generational change in which people are defined by the technological culture with which they are familiar.

university student. In particular, the UNLP has implemented tutorship programs that configurate a pedagogical strategy that enables student entry and permanence in university, guiding freshmen in their insertion into university life (Castagno F. et al, 2013).

University receives increasingly more students accessing information through ICTs, while access to university contents is still ruled by the traditional way of the printed culture.

2.2 Digital natives and mobile devices

Nowadays, ICTs are part of the social culture of the students entering university, of the way in which they bond and present themselves to the world. In general, these subjects arrive in university with digital literacy as a part of their daily lives, which makes them "digital natives".

According to Piscitelli (Piscitelli A, 2006), the "digital natives" are characterized as wanting to receive information immediately, prone to multitasking, favoring graphics over text, being inclined to non-structured information, performing better when working online and preferring gamified education over traditional education.

They have an "innate quality" for developing practices through digital technology and an awareness of progress, which gives them immediate reward and satisfaction similar to what they would obtain from their favourite videogames.

Computers, videogame consoles, and smartphones are some of the devices necessary to thrive in any environment, as they use their devices for most of their daily activities, from obtaining information to communicating and providing information. It is important that activities proposed to freshment are attractive to them as "digital natives" and allow for maximum involvement in relation to what they like.

A study conducted by UNICEF Argentina in October 2013 on "Access, Consumption and Behavior of Teenagers on the Internet" indicates that their main uses for the Web are to socialize, chat with friends and family (82%), followed by a similar percentage of activities such as online gaming (63%), searching for information (61%) and watching movies (59%). This survey also shows that, among the devices used by teenagers, cellphones have won over terrain in socialization through social networks, reaching high percentages (76%).

Mobility has come to be a requirement, both in the way we communicate as in the place where we socialize. Roxana Morduchowicz (Morduchowicz, R., 2013), coordinator of the program Medios y Escuela (Media and the School) of the Ministry of Education of the Nation, points out that the cellphone, due to its portable nature, is the device with which teenagers spend the most time during the day: "Not only does it go with them everywhere, for most of them it is on 24 hours a day. Half of teenagers never turn off their phone, and 30% turn it off to go to sleep". According to the specialist, the cellphone is also a mark of identity: "it gives them a sense of belonging to a group and strengthens their social life, two dimensions that are fundamental to teenagers".

Following these ideas, some universities such as Lewis University (USA), Queen's University (Canada), Amity University (India), Rider University (USA), Queensland University of Technology in Brisbane (Australia) have developed mobile applications to guide their students inside the campus, help them in administrative affairs, keep

them informed regarding activities and events, and aid in communications between students and teachers, among others (Google Play, 2014) (Fitz-Walter, Z., 2013).

2.3 Gamification and Learning Spaces

It is important to integrate habitual practices of "digital natives" into the learning process so the learning environment is not isolated. Motivation and engagement are indispensable prerequisites for learning. In this sense, Prensky (Prensky M., 2014) proposes we "incorporate into the learning process goals and options that are interesting to the participants, immediate feedback and possibilities for growth, as this combination motivates and commits digital natives to their favorite games".

The use of gaming elements to improve participation and commitment from people in non-ludic activities is called gamification. Recent developments in the field of ubiquous and mobile computing technologies and daily use technologies, such as smartphones, allow us to locate gaming elements in most daily activities, enhanced by the use of contextual information provided by sensors within the devices.

For Mariana Maggio (Maggio, M, 2012), a specialist in educational technology, educational proposals must be generated that enable the acquisition significant and long-lasting learning from the acknowledgement of the realities of students and the framework of technological development and knowledge generation in the field to which the teacher belongs.

Today, we must fix our eyes on the students' particularities and complexities if we want to increase their inclusion in the university environment. This approach values social histories and previous experiences brought along by the students (Saccone, J., 2013).

Juan Domingo Farnós makes reference to ubiquitous learning as a way to establish relations between education and work: "learning can take place anywhere, but we must value it, i.e., if learning takes place outside the classroom, what is learned has to have as much value as curricular content –not only that, it has to have more value than curricular content for being learning that attends to what each student wants to learn".

2.4 Entry to the Computer Science School of the UNLP

The Computer Science School of the UNLP has a direct entry system with the additional exigence of an entry course with mandatory attendance and a non-eliminatory diagnostic test. During the entry course, students must attend three disciplinary modules: "Expresión de Problemas y Algoritmos" ("Expressing Problems and Algorithms"), "Concepto de Organización de Computadoras" ("Computer Organization Concepts") and "Matemática 0" ("Mathematics 0"), and the "Taller de Inserción a la Vida Universitaria (TIVU)" ("Workshop for Insertion into University Life"). This workshop was introduced in 2012 in the activities of the entry course and its goal is to support, guide and promote the integration of freshmen with knowledge that will allow them to lead their academic lives the best possible way.

At the Computer Science School, the development of the tutorial activity was linked to the implementation of the national PACENI program (Proyecto de Apoyo para el Mejoramiento de la Enseñanza en Primer Año de Carreras de Grado de Ciencias Exactas y Naturales, Ciencias Económicas e Informática – Support Project for the Improvement of Teaching in the First Year of Computer, Exact, Natural and Economic

Sciences Degree Courses) promoted by the Ministry of Education of the Nation in the year 2009. Peer tutors, who are advanced students of the multiple degree courses of the Computer Science School, support the new students in their transit through academic life. This guidance is a means that enables the creation of social bonds and university affiliation within the institution (Díaz, J., 2013).

3 WhatsINFO

WhatsINFO is a project whose goal is to provide another means to mitigate the existing divide between new university students and university culture, through a technological tool. Although many universities in Argentina have guidance strategies for the first years of the courses, such as the tutorship system, no technological developments exist to complement them. WhatsINFO is proposed as a technological complement to activities undergone in the framework of the entry course to the Computer Science School of the UNLP, giving value to cultural and social practices among teenagers.

WhatsINFO is a mobile application whose distinctive characteristics are gamification and socialization. This references both the use of game elements in the application and the formation of a social network around the group of Computer Science freshmen. Using gaming mechanisms and enabling socialization through digital means gives value to the innate skills and knowledge freshmen bring with them.

The main feature of WhatsINFO is the set of challenges presented to the users that encourage competition. It seeks to motivate students by means of gaming mechanisms and socialization. These challenges provide information relevant to the students and encourage learning of the institutional culture. By sucessfully completing the challenges, users obtain points that will accumulate and allow them to compete in a ranking with other WhatsINFO users.

WhatsINFO challenges students to answer multiple choice questions on topics of their interest, connect with friends in increasing amounts (5 friends, 10 friends, 15 friends), visit facilities such as the Students Department, the library, the canteen, or the students union, attend events and activities in the framework of the entry course and get to know the tutors. By getting students to be informed regarding events of their interest, tour the premises and socialize with their peers, WhatsINFO introduces them slowly into the institutional structure of the School.

3.1 What Information do Freshmen Need

The development of WhatsINFO began in 2013 and the goal set was a pilot test during the 2014 entry course. During the first development stage research was performed on the most common difficulties and questions among freshmen. For this task, during December 2013 and February 2014, a series of interviews were conducted with the people in charge of the institutional areas of the Computer Science School that have contact with the freshmen: Entry Department, Teaching Department, Accessibility Department, Pedagogical Department, Library, Student Affairs Sub-Secretary and Student Union.

The information gathered allowed for the detection of the most frequent problems dealt with by freshmen, the matters most consulted in each area and the information these areas consider fundamental for these students. The most frequently asked

questions are related to exam, enrollment and class dates, use of institutional systems (SIU-Guaraní, Moodle, webUNLP), spaces that hold information on the subjects, their timetables, evaluations, services offered by the School and the UNLP to its students, how to find specific places in the School and everyday entry course matters such as how to: pass subjects, justify absences, contact coordinators, locate classrooms, etc. Based on these questions, all the areas considered that WhatsINFO could become a centralized medium to collaborate in spreading this information. The most commonly used means of institutional communication are the School website, Facebook, Twitter, the School bulletin boards and to a lesser extent email and telephone.

This information was the basis for the ellaboration of activities, setting of challenges and news dissemination through WhatsINFO.

3.2 WhatsINFO Architecture

WhatsINFO has a client-server architecture comprised of an Android mobile client application and a content management server.

One of the features of a client-server architecture is centralized resources due to the server being at the center of the application, able to manage the resources common to all users. The choice of architecture for implementation was influenced by the need to have challenges, events and news load dynamically and to share the ranking and have feedback on the challenges overcome by the users.

The WhatsINFO client only consumes information and it is the server that handles all the processing and content management. When an important event takes place, the server generates an alert that the client consumes and shows the user in the form of a notification in the status bar.

Communications between both applications take place through a REST architecture on a secure connection using a combination of the TLS and HMAC security protocols. This communication is structured by means of a REST API with JSON format. As an indoor positioning method we used a type of symbolic location by means of QR codes.

WhatsINFO was developed with open source code tools and technologies and freeware ones in some specific cases –all of them are broadly used in software development.

Content Management Server Application

This web application was written in PHP using the Symfony 1.4 development framework with a MySQL data base server and HTML5 and CSS 3 for the graphic interface.

This centralized server is used to feed the mobile client application. It helps create and manage user accounts, events, challenges, news, questionnaires and surveys for users to access. Only the content administrator user can access this application.

Fig. 1 shows the statistic screen of the WhatsINFO server application.

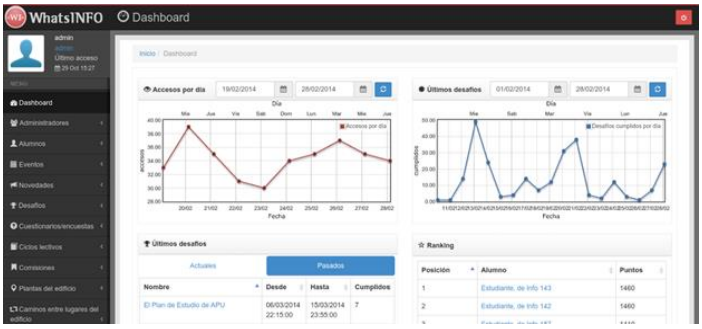


Fig. 1. Screen capture of the statistics of the content management Server Application

The Mobile Client Application

As stated in Queiruga et al. (Queiruga, C., 2013) "Mobile phones have more and more features, bigger and more precise screens, incorporated cameras for pictures and video, they play music and come with multiple sensors, and they have the capability to be always on. This is a challenge to the development of new applications that stop being isolated entities exchanging information through a user interface. The development of innovative applications that allow for the integration of multiple technologies involving mobile devices is a very motivating and current topic for young university students". For a mobile application to be attractive and usable, not only does it have to adapt to a "more reduced" medium, but its presentation must be implemented in a new and creative way.

The Android platform was chosen for the development of the WhatsINFO mobile client application because it is an initiative that is close to freedom in mobile applications and it promotes an ethics controlled by the software user. Moreover, Android is currently the most popular platform, present in over 75% of mobile devices in Argentina (StatCounter Global Stats). Additionally, because Android promotes openness, it makes it posible to create applications free of charge and the developer has the support of a very active community of developers with very complete and up-to-date documentation through the oficial Android developers website (http://developer.android.com/).

The WhatsINFO mobile client allows students to access information on events, news, friends, user profile, WhatsINFO ranking, school maps and challenges, which can be done communicating with the content management server through the Internet.

Fig. 2 shows the list of challenges on the left of the screen and the WhatsINFO ranking on the right.

The Android client application was developed following interaction patterns and design standards present in the currently most popular applications. Its main goal is to provide a good experience to students by means of a friendly and easy-to-use application.

4 The WhatsINFO Experience

In order to evaluate the use of WhatsINFO with real freshmen, a pilot test was performed with a small number of students from the 2014 entry course that expressed

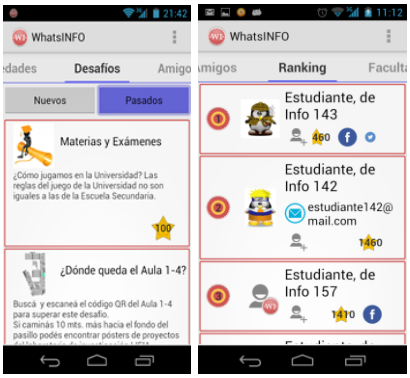


Fig. 2. Screen capture of the list of challenges (left) and the WhatsINFO ranking (right)

interest in participating in the WhatsINFO trial. For this purpose and with the goal of examining the practices in relation to ICT and mobile application use among Computer Science School freshmen, during the 2014 enrollment process, a survey was conducted on applicants to different courses in the School. This survey focused on how many of them have a mobile device (cellphone or tablet) with the Android operating system and whether they would be willing to participate in the experience of using WhatsINFO.

228 surveys were answered, out of which 132 applicants to the 2014 academic year answered they had a mobile device with the Android operating system and they wanted to participate in the WhatsINFO experience.

The WhatsINFO management application was installed in a server provided by the LINTI of the Computer Science School.

WhatsINFO was published in the official Google application store for Android, Google Play, so the applicants could download the application and install it in their cellphones or tablets.

In order to orient the students in the premises of the Computer Science School, QR codes were placed in each classroom and areas frequented by students (20 signs in total) with the goal of allowing them to complete the challenges and enabling them to locate themselves in the facilities.

During the WhatsINFO experience, 183 user accounts, 14 events, 57 news and 36 challenges were created. The frequency with which the students were posed challenges was an average of 5 every week. Out of the 183 users, 94 accessed the application and 58 completed at least one challenge.

The application displayed the daily canteen menu, information on scholarships, means of transport, university diner, and important dates and schedules.

When the entry course was finished, a meeting was organized with the Computer Science School freshmen that used WhatsINFO to hear their opinion about the application. Some awards were given and the attendants participated in further surveys and interviews with the goal of obtaining information to evaluate the experience.

5 Results of the experience

The WhatsINFO pilot test included approximately 30% of the 2014 applicants to the Computer Science School degree courses. The test took place in February and March

and evaluated the applicability of a technological tool close to teenage culture and aimed at guiding new students in their transition from high school to university.

The analysis of the result was performed taking into account: Google Play score, active users registered in the application, the activity of the users, surveys and interviews conducted once the course was over and problems detected during the pilot test.

The statistics provided by Google Play Store showed that over 150 students downloaded and installed the application and rated it very positively, with an average of 4.93 over 5.

The experience took place within a period of two months during which almost a hundred students used the application and solved over thirty challenges. Compared to the amount of user accounts created, the average application use was 50%. It is considered that this shelling was due to problems with some versions of Android, with the WiFi networks of the School and with a lack of use of email accounts by the students, as they were the main initial means of communication with WhatsINFO users.

The maximum score students could obtain by completing all the challenges was 1500. Two students obtained 1460 points, which is 97% of the total.

It is worth nothing that 104 friendships were registered in WhatsINFO, and 20 users completed the challenge "Make friends" (it required the user make at least 5 friends).

The surveys conducted through WhatsINFO and the personal interviews at the end of the experience yielded encouraging comments on the design and usability of the application. Freshmen gave a positive rating to the real time notification system and showed considerate interest in challenges, considering that WhatsINFO "sets itself apart from the seriousness and motivates you" and "provides interesting information and fun by means of the challenges". Some of the suggestions included: incorporating communications like the ones in Facebook or WhatsApp, including information regarding first year subjects and making WhatsINFO available to other mobile platforms.

6 Conclusions

Given the complexity of the transition between high school and university and giving value to the characteristics of digital natives, among which are the freshmen of the Computer Science School of the UNLP, WhatsINFO was built and implemented as a gamified mobile tool to guide them and promote their communication and participation during the Computer Science School entry process.

It is worth noting that this experience constituted an innovative strategy that encouraged students to become involved in the institutional culture through a daily technological medium. This evidences that the use of technology can contribute to solving social problems by becoming a modern complement to traditional practices.

We were able to prove the innate capacity of freshmen in relation to the use of mobile applications and their digital hability to adapt to new applications, not needing specific training or directions to meet the challenges posed by WhatsINFO, befriend the medium, edit the profile information and discover the multiple functionalities.

WhatsINFO fosters the preferences and expectations of freshmen that choose real time communication means such as chats and social networks and value socialization through digital media.

References

Carli, S. (2007): “La experiencia universitaria contemporánea. Transmisión y sociabilidad”. Conferencia organizada por el Programa de Evaluación, Acreditación e Innovación, Subsecretaría de Grado de la Secretaría de Asuntos Académicos de la Universidad Nacional de Córdoba, Córdoba, Argentina.

Carli, S. (2012): “El estudiante universitario. Hacia una historia del presente de la educación pública”. First edition. Serie Educación y Sociedad. Siglo XXI. ISBN 978-987-629-239-9 .

Castagno, F., Ferrer, M., Piretro, A., & Rodríguez, C. (2013): “Ingresantes y entornos virtuales en la formación de comunicadores sociales: acerca de lazos y enlaces”. V Encuentro Nacional y II Latinoamericano de Ingreso Universitario, August 2013, Universidad Nacional de Luján. (2013)

Díaz, J., Queiruga, C., Ungaro, A., Vilches Antao, D., Felipe, C., Gómez, S., Kimura, I (2013).: “Tutoring as an Inclusion Strategy. The UNLP Computer Science School Case - Facultad de Informática - Universidad Nacional de La Plata”. Conferencia Latinoamericana en Informática 2013 (CLEI), Naiguatá, Vargas, Venezuela.

Druetta, D., Garay Cruz, L., López González, R., Portillo Sánchez, M. (2011): “Uso y apropiación de la telefonía móvil. Opiniones de jóvenes universitarios de la UNAM, la UACM y la UPN”. Derecho a Comunicar Magazine. Issue No. 3. Asociación Mexicana de Derecho a la Información A. C. ISSN: 2007-137X

Fitz-Walter, Z., Tjondronegoro, D. & Wyeth, P. (2012): “A Gamified Mobile Application for Engaging New Students at University Orientation”. OzCHI '12 Proceedings of the 24th Australian Computer-Human Interaction Conference. ACM New York, NY, USA. ISBN: 978-1-4503-1438-1.

Google Play (2014): Google Play Store. Available at: https://play.google.com/store.

Maggio, M. (2012): “Enriquecer la Enseñanza”. Paidós. ISBN: 950-12-1535-9.

Medrano, L., Olaz, F (2008): “Autoeficacia Social en Ingresantes Universitarios: su relación con el Rendimiento y la Deserción Académica”. Revista de Psicología General y Aplicada: Revista de la Federación Española de Asociaciones de Psicología, Volume No. 61, Issue No. 4, pp. 369-384. ISSN: 0373-2002.

Morduchowicz, R. (2013): “Los adolescentes del siglo XXI, los consumos culturales en un mundo de pantallas”. Fondo de Cultura Económica. ISBN: 9789505579624

Piscitelli, A. (2006): “Nativos e inmigrantes digitales: ¿Brecha generacional, brecha cognitiva o las dos juntas o más aún?”. Revista Mexicana de Educación Educativa, Volume No. XI, Issue No. 28. ISSN: 1405-6666.

Prensky, M. (2014): “The World Needs a New Curriculum”. The Global Future Education Foundation and Institute, Ltd. ISBN-10: 0578144239

Queiruga, C.; Banchoff Tzancoff, C.; López, F. (2013): “RemoteBot: una Aplicación que Combina Robots y Dispositivos Móviles”. XV Workshop de Investigadores en Ciencias de la Computación (WICC 2013). Paraná, Entre Ríos, Argentina.

Saccone, J., Pacífico, A., Cáneva, C. y Fierro, M. (2013): “Análisis del acceso a la UNL, los dispositivos político-académicos y la normativa institucional como herramienta para la igualdad de oportunidades”. V Encuentro Nacional y II Latinoamericano sobre Ingreso a la Universidad Pública “Políticas y estrategias para la inclusión. Nuevas complejidades; nuevas respuestas”. Available at:http://goo.gl/qrOScs. [Last access: August 2015]

StatCounter Global Stats (2015). “Top 8 Mobile & Tablet Operating Systems in Argentina from Jul 2013 to Jul 2015”. http://gs.statcounter.com/#mobile_os-AR-monthly-201407-201507.

UNLP (2010): “Plan Estratégico de la Universidad Nacional de La Plata”. Available at: http://www.unlp.edu.ar/uploads/docs/estrategia_1___ensenanza_pe_2010_2014.pdf

Walker, V. (2012): “El tránsito de los estudiantes por la universidad”. Editado por la Fundación Universitaria Andaluza Inca Garcilaso para eumed.net. ISBN: 978-84-15547-99-0. Available at: http://www.eumed.net/libros-gratis/ciencia/2012/10/index.htm

VI Workshop Procesamiento de Señales y Sistemas de Tiempo Real (WPSTR)

7441 | Clasificación binaria, desbalanceada y contextual de voxels asociados a series temporales
Javier Giacomantone (UNLP), Armando E. De Giusti (UNLP)

7533 | Automatización del posicionamiento de un detector y adquisición de datos para un Difractómetro de Rayos X
Jorge Runco (UNLP), Alberto Ghisolfi (UNLP)

7548 | Descriptor Based on Spectral Peaks Correlograms
Brenda Lilia Valdez Reyna (UAT), Miroslav Zivanovic (UPNa)

7551 | Diseño Robusto basado en PSO de un Controlador PID para Sistemas con Retardo de Tiempo Variable: Aplicación a AQM
Patricia Noemí Baldini (UTN-FRBB), Guillermo Calandrini (UNS)

7567 | Evaluación de Sistemas Operativos de Tiempo Real sobre Microcontroladores
Santiago Medina (UNLP), Martin Pi Puig (UNLP), Juan Manuel Paniego (UNLP), Matías Dell Oso (UNLP), Fernando Romero (UNLP), Armando E. De Giusti (UNLP), Fernando G. Tinetti (UNLP)

7585 | Optimización de las Señales Acústicas Utilizando Beamforming de Banda Ancha para Personas con Discapacidad Auditiva
Paúl Bernal (EPDE), Amelia Caisapanta (EPDE), Fabián Sáenz (UNLP)

7586 | Sistema de adquisición de datos para medir desorción ó absorción de hidrógeno activadas térmicamente
Jorge Runco (UNLP), Marcos Meyer (UNLP)

7600 | Evaluación de Parámetros Adecuados para una Red Inalámbrica de Sensores
Guillermo H. Reggiani (UTN-FRBB), Guillermo R. Friedrich (UNSL), Sergio Pellegrino (UTN-FRBB)

Clasificación binaria, desbalanceada y contextual de voxels asociados a series temporales

Javier Giacomantone, Armando De Giusti

Instituto de Investigación en Informática (III-LIDI),
Facultad de Informática - Universidad Nacional de La Plata - Argentina.
La Plata, Buenos Aires, Argentina.
{jog,degiusti}@lidi.info.unlp.edu.ar

Resumen. En este artículo se presenta un método computacional para clasificación de regiones 3D en función de sus características dinámicas. La clasificación de voxels atípicos se implementa en función de las series temporales asociadas a los mismos. El método opera en clasificación binaria, clases desbalanceadas y correlación espacial de las series asociadas a cada clase. El método propuesto utiliza máquinas de soporte vectorial y difusión anisotrópica robusta para detectar la estructura subyacente en los datos y clasificar los voxels correspondientes en cada clase. Se presentan resultados experimentales del método propuesto para datos de resonancia magnética funcional e imágenes de rango.

Palabras clave: Reconocimiento de patrones, Series temporales, Máquinas de soporte vectorial, Difusión anisotrópica, fMRI, TOF.

1. Introducción

El problema de segmentar una región en un volumen discreto está presente en diversas áreas de aplicación entre las que podemos mencionar imágenes de tomografía computada, imágenes de resonancia magnética (MRI), imágenes de resonancia magnética funcional (fMRI), imágenes de tiempo de vuelo (TOF), imágenes de ultrasonido, fusión de modalidades y procesamiento de datos multidimensionales discretos en general [1][2]. Las coordenadas de cada elemento de volumen, denominado voxel, normalmente un cubo, son discretas. Así como se puede generalizar un pixel en 2D a un voxel en 3D, también se puede generalizar el término voxel a un elemento de volumen dinámico o doxel en 4D. La cuarta variable independiente normalmente es el tiempo o la distancia. La Fig.1(a) muestra una estructura de datos 3D y un voxel en particular. La Fig.1(b) nos muestra una estructura 4D y un doxel en particular. En este artículo consideramos que todas las etapas de pre-procesamiento ya fueron realizadas tales como co-registro, correcciones de escala, normalización y deriva. Si bien facilita y resulta intuitivo plantear el problema como imágenes 3D o 4D, no necesariamente los datos considerados tienen que corresponder a imágenes. En general

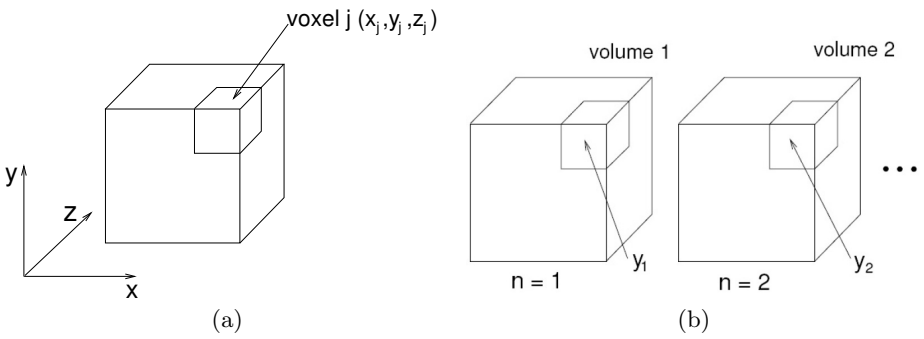


Figura 1. (a) voxel 3D, (b) doxel 4D

no son imágenes naturales, sino datos de distinto tipo visualizados como imágenes o series temporales para facilitar su interpretación, como datos de rango, temperatura o presión entre otros.

El método propuesto (MP) tiene como objetivo la clasificación de voxels en dos clases desbalanceadas. Los voxels de la clase minoritaria pueden considerarse patrones atípicos de acuerdo al nivel de desbalance. La segunda hipótesis del método es que existe correlación espacial entre voxels de cada clase. En particular el objetivo es detectar voxels correlacionados en la clase minoritaria. En este artículo se presentan resultados experimentales con imágenes funcionales TOF y fMRI. La modalidad de imágenes médicas denominada fMRI involucra considerar una imagen estructural 3D y una imagen funcional 4D [3][4]. La cámara TOF MESA SR4000 nos permite obtener secuencias de intensidad y rango [5]. Una imagen fMRI puede ser vista como un conjunto de series temporales donde cada serie, cada doxel, corresponde a un voxel particular en la imagen estructural [6][7]. Las máquinas de soporte vectorial de una clase "One class SVM (OCSVM)" [8][9] han sido también utilizadas en MRI y fMRI [10]. En fMRI del cerebro las series temporales de la mayoría de los voxels son independientes del estímulo experimental y solo pocas series temporales están relacionadas con el paradigma experimental utilizado. Estas últimas pueden ser consideradas como atípicas y por lo tanto satisfacen las hipótesis necesarias para aplicar OCSVM sobre el conjunto de datos fMRI [11]. La hipótesis de correlación espacial sostiene que las series temporales de voxels próximos tienen un estado similar de activación, correlacionado o no correlacionado con el estímulo experimental. Distintos métodos han sido propuestos para reflejar esta última característica de las señales fMRI [12][13][14]. El método propuesto, basado en asumir clases desbalanceadas y correlación espacial, es aplicable en fMRI y en imágenes de tiempo de vuelo. El artículo esta organizado del siguiente modo: las secciones 2 y 3 presentan una breve revisión de los conceptos fundamentales utilizados en el método propuesto. La sección 4 describe el método propuesto. En la sección 5 se muestran resultados experimentales. Finalmente, en la sección 6 se presentan conclusiones.

2. Difusión Anisotrópica Robusta

Perona y Malik [15] definieron difusión anisotrópica como

$$\frac{\partial I(x,y,t)}{\partial t} = \operatorname{div}\left[g\left(\|\nabla I(x,y,t)\|\right)\nabla I(x,y,t)\right], \tag{1}$$

utilizando la imagen original $I(x,y,0) : \mathbb{R}^2 \to \mathbb{R}^+$ como la condición inicial, donde t es un parámetro de tiempo artificial y g es una función que se denomina de parada en los bordes ("edge-stopping function"). La elección adecuada de g determina en que medida, al difundir, las discontinuidades son preservadas. Perona y Malik sugirieron utilizar dos posibles funciones con este objetivo [15]. Black et al. [16] utilizaron teoría de estimación robusta para determinar una mejor función de parada en los bordes denominada "Tukey's biweight".

$$g(x) = \begin{cases} \left[1 - \frac{x^2}{5\sigma^2}\right]^2, & \frac{x^2}{5} \leq \sigma^2 \\ 0, & \text{en otro caso} \end{cases} \tag{2}$$

La función anterior, g, es una versión dilatada y escalada de la función de Tukey, donde $g(0) = 1$ y el máximo local de su función de influencia $\psi(x) = xg(x)$ se localiza en $x = \sigma$. Cuando se utiliza la función de Tukey el proceso de difusión se denomina Difusión Anisotrópica Robusta (RAD).
Perona y Malik [15] propusieron una solución discreta para la ecuación de difusión anisotrópica robusta (1) dada por:

$$I(s,t+1) = I(s,t) + \frac{\lambda}{|\eta_s|}\sum_{p\in\eta_s} g(|\nabla I_{s,p}(t)|)\nabla I_{s,p}(t), \tag{3}$$

donde $I(s,t)$ es una imagen muestreada, s es la posición del pixel en una grilla discreta 2-D o 3-D , $t \geq 0$ son los pasos temporales, la constante λ determina la difusión (normalmente, $\lambda = 1$), y η_s representa el conjunto de vecinos de s. Para imágenes 2-D son los cuatro vecinos y para 3-D los seis vecinos del voxel correspondiente, excepto en las fronteras. La magnitud del gradiente en un voxel en la iteración t es aproximada por:

$$\nabla I_{s,p}(t) = I(p,t) - I(s,t),\ p \in \eta_s. \tag{4}$$

Utilizando la función de Tukey, el proceso de difusión converge mas rápido y genera bordes mas agudos que usando las funciones de Perona y Malik. Black et al. [16] mediante elementos de teoría de estimación robusta, obtienen un valor estimado de la escala robusta σ_e dado por:

$$\sigma_e = 1{,}4826\,\mathrm{MAD}(\boldsymbol{\nabla} I) = 1{,}4826\,\mathrm{median}_I\left[\|\|\boldsymbol{\nabla} I\| - \mathrm{median}_I(\|\boldsymbol{\nabla} I\|)\|\right], \tag{5}$$

donde MAD es la desviación absoluta media.

3. Máquinas de soporte vectorial

Dados l patrones de entrenamiento para dos clases $\{(x_i, y_i), i = 1, ..., l\}$, donde $x_i \epsilon \mathbb{R}^N$ es un patron, un vector de características de dimensión N e $y_i \epsilon \{1, -1\}$ es la etiqueta de la clase a la que pertenece x_i. Decimos que $\mathbf{x}_1, \ldots, \mathbf{x}_l \in \mathcal{X}$ donde $l \in \mathbb{N}$ es el número de observaciones y $\mathcal{X}$ es un subconjunto compacto de $\mathbb{R}^N$. Una SVM determina el hiperplano que maximiza la separación entre ambas clases y es matemáticamente equivalente a maximizar el margen dado por $\frac{2}{||\mathbf{w}||}$ sujeto a $y_i(x_i.w) + b) \geq 1$. El hiperplano $(w.x) + b = 0$ queda determinado por b y w. Se puede relajar la condición de muestras de entrenamiento separables incluyendo variables artificiales que permitan mejorar también la generalización. Para obtener el hiperplano minimizamos

$$\begin{aligned} &\text{mín} && \tfrac{1}{2}||\mathbf{w}||^2 + C \textstyle\sum_i \xi_i \\ &\text{sujeto a} && y_i[(w - x_i) + b]) \geq 1 - \xi_i \end{aligned} \tag{6}$$

donde C controla el compromiso entre complejidad del hiperplano y el error sobre el conjunto de entrenamiento. Resolviendo el Lagrangiano en la forma dual obtenemos la función de decisión como:

$$f(\mathbf{x}) = \text{sgn}[\sum_i \alpha_i y_i (x.x_i) + b]$$

donde α_i son los multiplicadores de lagrange. Si las clases no son linealmente separables es posible extender el método proyectando los patrones en un espacio de mayor dimensión donde si sea posible clasificarlos linealmente, para luego obtener la función de decisión no lineal en el espacio original. Pueden ser utilizados distintos tipos de kernels que corresponden a estimadores no lineales en el espacio de entrada. En particular en este trabajo es utilizado un kernel de base radial $k(x, x_i) = e^{-\gamma \|x - x_i\|^2}$, γ es el parámetro que determina el ancho del kernel.

Han sido propuestos dos algoritmos basados en SVM para operar con una clase, "support vector data description" [8] y "one-class SVM (OCSVM)" [9]. El algoritmo estima una función de decisión f que es positiva para un subconjunto del espacio de características y negativa para el complemento. El mapeo de los datos al espacio de características corresponde al kernel y separa los datos del origen con máximo margen. Sea $\Phi : \mathcal{X} \to \mathcal{F}$ un mapeo de características en $\mathcal{F}$ de tal modo que el producto interno en la imagen de Φ puede ser calculada evaluando un kernel simple.

$$k(\mathbf{x}, \mathbf{z}) = (\Phi(\mathbf{x}) \cdot \Phi(\mathbf{z}))$$

Se puede reescribir como un problema de optimización.

$$\begin{aligned} &\min_{\mathbf{w} \in \mathcal{F}, \xi \in \mathbb{R}^l, \rho \in \mathbb{R}} && \tfrac{1}{2}||\mathbf{w}||^2 + \tfrac{1}{\nu l} \textstyle\sum_i \xi_i - \rho \\ &\text{sujeto a} && (\mathbf{w} \cdot \Phi(\mathbf{x}_i)) \geq \rho - \xi_i, \xi_i \geq 0, \end{aligned} \tag{7}$$

donde $\nu \in (0, 1]$ es un parámetro que controla el término de penalización y ξ_i son variables de holgura. Resolviendo el problema de optimización (7) obtenemos $\mathbf{w}$ y ρ y la función de decisión es -1 para valores atípicos en el conjunto de datos y +1 para el resto de los patrones en el conjunto de datos.

$$f(\mathbf{x}) = \text{sgn}(\mathbf{w} \cdot \Phi(\mathbf{x})) - \rho) \tag{8}$$

Introduciendo multiplicadores de Lagrange $\alpha_i, \beta_i \geq 0$, obtenemos

$$L(\mathbf{w}, \xi, \rho, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \tfrac{1}{2}||\mathbf{w}||^2 + \tfrac{1}{\nu l}\textstyle\sum_i \xi_i - \sum_i \beta_i \xi_i - \rho$$

$$- \textstyle\sum_i \alpha_i(\mathbf{w} \cdot \Phi(\mathbf{x}) - \rho + \xi_i)$$

Fijando las derivadas con respecto a las variables primarias $\mathbf{w}, \xi, \rho$ igual a cero obtenemos

$$\mathbf{w} = \textstyle\sum_i \alpha_i \Phi(\mathbf{x}_i),$$

$$\alpha_i = \tfrac{1}{\nu l} - \beta_i \leq \tfrac{1}{\nu l},$$

$$\textstyle\sum_i \alpha_i = 1.$$

La función de decisión puede ser escrita como

$$f(\mathbf{x}) = \text{sgn}(\sum_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) - \rho)$$

Los multiplicadores α_i se puede resolver en el problema dual:

$$\min_{\alpha} \qquad \tfrac{1}{2}\textstyle\sum_{ij} \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j)$$

$$\text{sujeto a } 0 \leq \alpha_i \leq \tfrac{1}{\nu l}, \ \textstyle\sum_i \alpha_i = 1.$$

El parámetro ρ puede ser obtenido notando que para cualquier α_i y el correspondiente patron $\mathbf{x}_i$ se satisface que:

$$\rho = (\mathbf{w} \cdot \Phi(\mathbf{x}_i)) = \sum_j \alpha_i k(\mathbf{x}_i, \mathbf{x}_j). \tag{9}$$

4. Método Propuesto

Sea I' una imagen funcional. El primer paso consiste en remover el valor medio de I', obteniendo así:

$$I = I' - \overline{I'} \tag{10}$$

y de este modo se evita difundir información estructural. Las series temporales correspondientes a cada voxel son consideradas como un patron. Los $\mathbf{x}_{i,j,k}$ son los patrones que corresponden la fila i, la columna j y la imagen k, identificando

una serie temporal particular o un punto del espacio de características. Los $\mathbf{x}_{i,j,k}$ son directamente los valores de entrada del problema de optimización (7). La solución optima dada por $\mathbf{w}$ y ρ se puede obtener resolviendo el problema dual y (9). Luego para cada $\mathbf{x}_{i,j,k}$ un valor de decisión primario $\mathbf{y}_{i,j,k}$ es obtenido,

$$\mathbf{y}_{i,j,k} = (\mathbf{w} \cdot \Phi(\mathbf{x}_{i,j,k})) - \rho \tag{11}$$

que representa la distancia entre un punto $\Phi(\mathbf{x}_{i,j,k})$ y un hiperplano en el espacio de mayor dimensión que el kernel mapea $(\mathbf{w} \cdot \Phi(\mathbf{x})) - \rho = 0$.

Sea la imagen funcional en la iteración $t \geq 0$ del proceso de difusión, $I(s, n, t)$, donde $I(s, n, 0)$ es la imagen inicial con la corrección de valor medio ya aplicada en la posición correspondiente al voxel s y volumen n, y $T(s, t)$ el mapa de activación formado a partir de $\mathbf{y}_{i,j,k} \forall i, j, k$ donde s es una posición particular i, j, k.

1. Sea $t \leftarrow 0$.
2. Calcular el mapa de activación $T(s, 0)$ utilizando OCSVM (11).
3. Calcular los coeficientes de difusión. El coeficiente de difusión entre un voxel s y su voxel vecino p en el instante t es:

$$g(|\nabla T_{s,p}(t)|), \text{ donde } \nabla T_{s,p}(t) = T(p, t) - T(s, t). \tag{12}$$

4. Con estos coeficientes aplicar difusión en $I(s, n, t)$, generando la imágen funcional difundida, $I(s, n, t + 1)$, en la iteración $t + 1$:

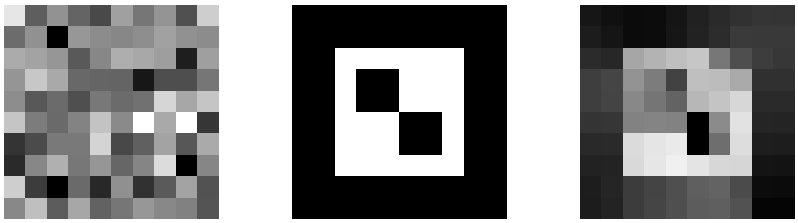
$$I(s, n, t + 1) \leftarrow I(s, n, t) + \frac{\lambda}{|\eta_s|} \sum_{p \in \eta_s} g(|\nabla T_{s,p}(t)|) \nabla I_{s,p}(t), \tag{13}$$

   donde $\nabla I_{s,p}(n, t) = I(p, n, t) - I(s, n, t)$.
5. Asignar $t \leftarrow t + 1$ y repetir los pasos 2 a 5 un número predefinido de veces o hasta que el valor medio de los valores difundidos (segundo término de la ecuación (13)) se encuentre por debajo de un determinado umbral.
6. Clasificar cada voxel aplicando la función de decisión dada por la ecuación (8).
7. Considerar los resultados los resultados anteriores como prototipos, conjunto de patrones de entrenamiento para un clasificador SVM de dos clases.
8. Re-clasificar todos los patrones del experimento con la máquina de soporte vectorial final obtenida.

La difusión anisotrópica es controlada por el número de iteraciones y el parámetro de escala de la función de parada (2), σ. La escala óptima en el método propuesto es entre dos y tres veces mayor que la escala robusta estimada. Tsiotsios y Petrou ([17]) analizan la elección adecuada de parámetros para imágenes en general.

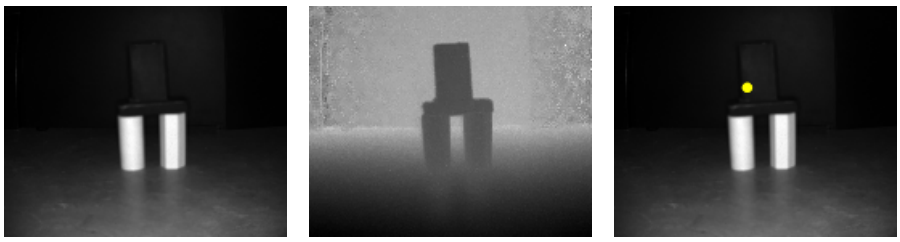
5. Resultados Experimentales

Para evaluar el método propuesto utilizamos imágenes fMRI e imágenes TOF. Se generaron dos imágenes sintéticas basadas en la imagen artificial (IA) propuesta en [18], con diferentes niveles de activación. El modelo fMRI utilizado tiene las siguientes características 10x10x3 voxels por volumen y 84 volúmenes. Todos los voxels tienen sumado ruido gaussiano de media 16000 y desvío estándar 4000. El valor de los voxels activados ha sido incrementado entre 600 y 1500 con incrementos de 100 generando así 10 imágenes sintéticas fMRI denominadas IA1 a IA10. Otras 10 imágenes sintéticas fueron generadas con ruido Rayleigh denominadas IA11 a IA20. El experimento fMRI que la imagen sintética utilizada representa está formado por bloques alternados de seis volúmenes activados y seis no activados. Los volúmenes activados tienen una región de 6x6x3 voxels activada y una de 2x2x3 voxels no activada. La Fig. 2 muestra una imagen 2-D activada correspondiente a la IA, la imagen de activación de referencia y el mapa de activación sin segmentar aplicando en método propuesto.



(a) (b) (c)

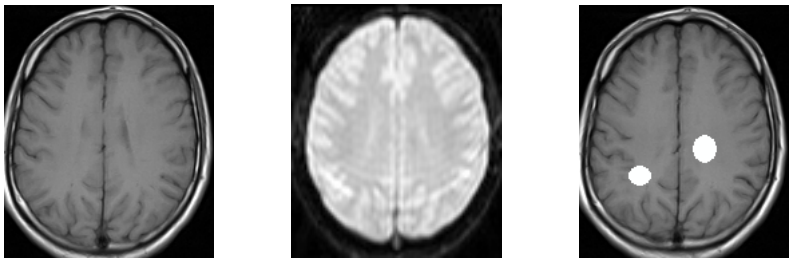
Figura 2. (a) IA-activada, (b) Imagen de referencia, (c) Mapa de activación



(a) (b) (c)

Figura 3. TOF

Se generaron 15 imágenes TOF semi-sintéticas sobre imágenes de intensidad y secuencias de 60 imágenes de rango correspondientes. Se utilizo una cámara ME-



(a) (b) (c)

Figura 4. fMRI con activación controlada

SA SwissRanger SR 4000, generando un paradigma experimental en bloques de 3 imágenes de 144x176 pixels. La Fig. 3 muestra una imagen de intensidad, una imagen de rango y el área generada artificialmente sobre la imagen de intensidad. En esta primer etapa de evaluación también fue generada una imagen semi-sintética, denominada ISSf, con dos estímulos esféricos artificiales sobre una imagen fMRI que después de ser interpolada, re-alineada y normalizada tiene dimensiones 79x95x68 voxels y una estructura en bloques como la de las imágenes totalmente sintéticas propuestas. La Fig. 4 presenta tres imágenes, un corte de la imagen MRI estructural de referencia (157x189x136) 4(a), denominada ISSe, un corte de la imagen funcional 4(b) y una representación del estímulo generado artificialmente sobre la imagen estructural 4(c).
La Fig. 5 muestra curvas ROC ("Receiving operating Characteristics"). Las curvas permiten comparar el método propuesto para el modelo fMRI artificial en tres métodos para IA4 y en dos para IA8, con menor relación señal a ruído, en las figuras 5(a) y 5(b). La curva ROC de la figura 5(c) corresponde al experimento TOF dinámico correspondiente a la Fig. 2. En la Fig. 5(d) las pruebas son realizadas sobre la imagen ISSf mostrando las curvas ROC correspondientes a MP y OCSVM. La tabla 1 presenta los parámetros correspondientes al punto óptimo de operación para las pruebas realizadas sobre IA8, ISSf y TOF.

6. Conclusiones

En este artículo se ha presentado un método alternativo de post-procesamiento de imágenes funcionales utilizando máquinas de soporte vectorial y difusión anisotrópica. La detección de regiones activadas mejora como consecuencia de dos mecanismos: la capacidad de detectar valores atípicos y la capacidad de reconocer la presencia espacio-temporal no aislada de los mismos. El análisis presentado muestra resultados que permiten considerar el método propuesto como viable y comparable con métodos alternativos. Han sido planificadas dos etapas futuras. En primer término analizar el comportamiento en distinto tipo de imágenes funcionales y paradigmas experimentales. Una segunda etapa involucra evaluar las

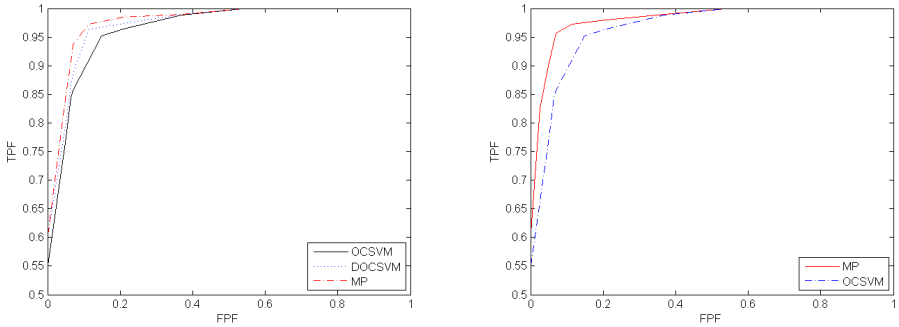
Modalidad	Método	Area	d_{po}	TPF_{po}	FPF_{po}
fMRI IA8	$OCSVM_{\nu=0,37}$	0.9615	0.5685	0.9521	0.1483
	$MP_{\sigma=1,8,t=3,\nu=0,37}$	0.9789	0.6275	0.95679	0.0691
fMRI ISSf	$OCSVM_{\nu=0,4}$	0.9659	0.5812	0.8914	0.0694
	$MP_{\sigma=2,t=3,\nu=0,4}$	0.9835	0.6343	0.9452	0.0481
TOF	$OCSVM_{\nu=0,38}$	0.9665	0.5821	0.8984	0.0684
	$MP_{\sigma=1,9,t=3,\nu=0,38}$	0.9866	0.6545	0.9552	0.0475

Tabla 1. Métricas de rendimiento

condiciones de ruido y variaciones entre los distintos tipos de imágenes funcionales.

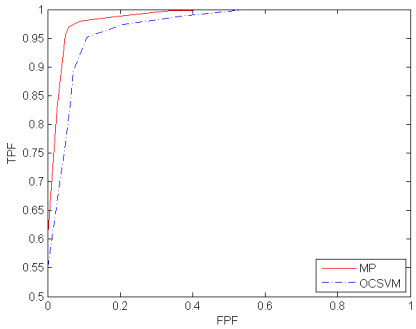
Referencias

1. J.A Noble. Ultrasound image segmentation and tissue characterization. Journal of Engineering in Medicine, vol. 223, pp. 1-10, 2009.
2. A. Salomon et. al. Simultaneous Reconstruction of Activity and Attenuation for PET/MR. IEEE Transactions on Medical Imaging, 30(3):804-813, 2011.
3. S. Ogawa et. al. Functional brain mapping by blood oxygenation level-dependent contrast magnetic resonance imaging. Biophysics Journal, 14(3):803-812, 1993.
4. N. Lange et. al. Plurality and Resemblance in fMRI Data Analysis. NeuroImage 10:282-303, 1999.
5. F. Chiabrando, D. Piatti, F. Rinaudo. R-4000 TOF Camera: Further Experimental Tests and First Applications to Metric Surveys. Proceedings of V Symposium on Remote Sensing and Spatial Information Sciences, 149-154, 2010.
6. B. Ardekani, I. Kanno. Statistical Methods for Detecting Activated Regions in Functional MRI of the Brain. Magnetic Resonance Imaging 16(10):1217-1225, 1998.
7. J. Tian, L. Yang, J. Hu. Recent advances in the data analysis method of functional magnetic resonance imaging and its applications in neuroimaging. Progress in Natural Science, 16(8):785-795, 2006.
8. D. M. J. Tax, R. P. W. Duin. Support vector data description. Machine Learning, 54:45-66, 2004.
9. B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Samola, R. C. Williamson. Estimating the support of a high dimensional distribution. Neural Computation, 13(7):1443-1471, 2001.
10. J. Zhang, K. Ma, M. H. Er, V. Chong. Tumor segmentation from magnetic resonance imaging by learning via one-class support vector machine. Proc. 2004 Int. Conf. Intell. Mechatronics Automation, 207-211, 2004
11. X. Song, A. M. Wyrwicz. Unsupervised spatiotemporal fMRI data analysis using support vector machines. NeuroImage, 47:204-212, 2009.
12. H. F. Chen, D. Z. Yao, S. Becker, Y. Zhou, M. Zeng, L. Chen. A new method for fMRI data processing: Neighborhood independent component correlation algorithm and its preliminary application. Science in China Series, 45(5):373-382, 2002.

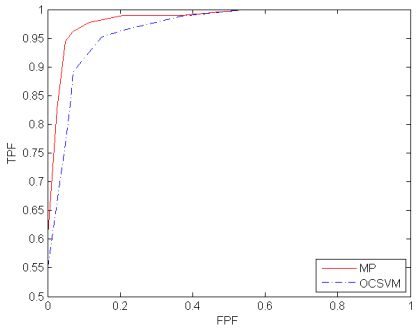


(a) fMRI:IA4

(b) fMRI:IA8



(c) TOF



(d) fMRI:ISSf

Figura 5. Curvas ROC

13. A. F. Sole, S. C. Ngan, G. Shapiro, X. P. Hu, A. Lopez. Anisotropic 2D and 3D averaging of fMRI signals. IEEE Transactions on Medical Imaging, 20(2):86-93, 2001.
14. J. Yang, N. Zhong, P. Liang, J. Wang, Y. Yao, S. Lu. Brain activation detection by neighborhood one-class SVM. Cognitive Systems Reasearch, 11:16-24, 2010.
15. P. Perona, J. Malik. Scale space and edge detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(7):629-639, 1990.
16. M. J. Black, G. Shapiro, D. H. Marimont, D. Hegger. Robust anisotropic diffusion. IEEE Transactions on Image Processing, 7(3):421-432, 1998.
17. C. Tsiotsios, M. Petrou. On the choice of the parameters for anisotropic diffusion in image processing. Pattern Recognition, 46:1369-1381, 2013.
18. H. Y. Kim, J. Giacomantone. A New Technique to Obtain Clear Statistical Parametric Map By Applying Anisotropic Diffusion to fMRI. IEEE International Conference on Image Processing,724-727, 2005.

Automatización del posicionamiento de un detector y adquisición de datos para un Difractómetro de Rayos X

Jorge Runco[1], Alberto Ghisolfi[2]
[1]Prof. Titular Ded. Exclusiva
[2]Profesional Principal - CONICET
[1,2]Depto.de Física – Facultad de Cs.Exactas– UNLP - IFLP
{ runco, ghisolfi }@fisica .unlp.edu.ar

Resumen

En este trabajo se describe la implementación del control y automatización del posicionamiento del detector y adquisición de datos, de un difractómetro de rayos x.

El sistema forma parte del equipamiento de la cámara de reacción de un acelerador de iones de 400 KV que posee el Depto. de Física de la UNLP, y está montado de manera de poder hacer difractometría [1] sobre muestras que están simultáneamente siendo tratadas con el acelerador. Esta cámara de reacción se encuentra en vacío y el generador de rayos x es externo a la cámara, entrando a la misma a través de una ventana de berilio. El detector está solidario a un sistema mecánico conectado a un motor paso a paso que permiten llevarlo a distintas posiciones.

Se propone un sistema de adquisición [2] automático para la colección de datos y para ser utilizado en una segunda etapa para una experiencia de correlación con otras variables, como por ej. el seguimiento automático del pico de un difractograma para variaciones de temperatura de la muestra.

Palabras claves: cristalografía, procesos, difracción, adquisición de datos.

1. Introducción

La cristalografía [1] es una técnica experimental que permite estudiar materiales, observando la dispersión y difracción de rayos x que inciden sobre el mismo.

Los rayos X interaccionan con la materia a través de los electrones que la forman y que se están moviendo a velocidades mucho menores que la de la luz. Cuando la radiación electromagnética X alcanza un electrón cargado éste se convierte en fuente de radiación electromagnética secundaria dispersada.

Según la longitud de onda y de las relaciones de fase de esta radiación dispersada, nos podemos referir a procesos elásticos, o inelásticos (dispersión Compton), dependiendo de

que no cambie, o cambie, la longitud de onda, y de coherencia o incoherencia según que las relaciones de fase se mantengan en el tiempo y en el espacio, o no.

Los intercambios de energía y momento que se producen pueden incluso dar lugar a la expulsión de un electrón fuera del átomo, seguido de la ocupación del nivel de este electrón por electrones de niveles superiores.

Todos estos tipos de interacciones dan lugar a diferentes procesos en el material como pueden ser: refracción, absorción, fluorescencia, dispersión Rayleigh, dispersión Compton, polarización, difracción, reflexión.

La figura 1 muestra el esquema de la experiencia para estudiar materiales mediante difracción de rayos x.

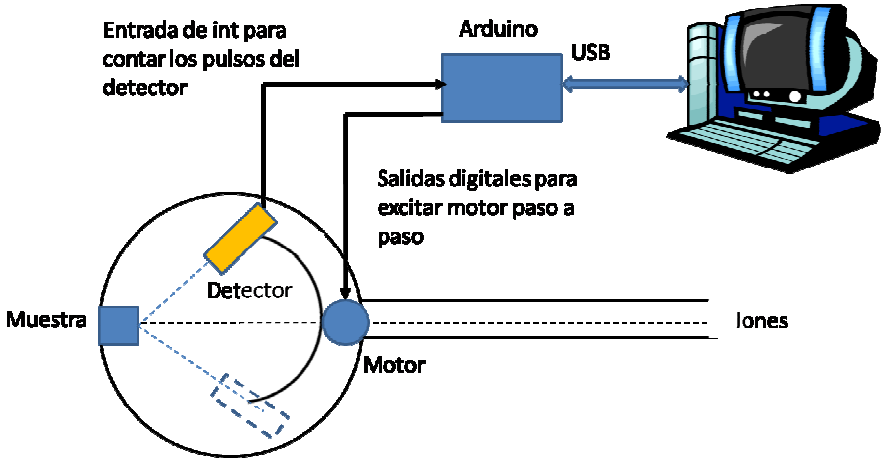


Figura 1. Disposición esquemática del equipamiento

Los rayos x inciden sobre la muestra y son difractados en distintas direcciones, observando como son dispersados estos rayos al impactar con el material, se puede estudiar la estructura cristalina del mismo. El detector va cambiando de posición y va contando la cantidad de rayos (pulsos) que fueron dispersados por la muestra en las distintas direcciones (posiciones).

La cámara de reacción se encuentra en vacío y el generador de rayos x es externo a la misma, conectada a ella a través de una ventana de berilio. El detector está montado sobre un goniómetro dentro de la cámara y es movido por un motor paso a paso, donde la posición es sensada a través de un encoder angular (sensor de posición).

La geometría está determinada entre otras cosas por el soporte del material a irradiar, el cual limita el ángulo de barrido aproximadamente a 270^{o}.

El detector, montado sobre el goniómetro, es un tubo fotomultiplicador [3] con su correspondiente base (polarización de electrodos), fuente de alta tensión y preamplificador, estos dos últimos fuera de la cámara.
En la figura 2 se muestra una imagen de la cámara de reacción y generador de iones.

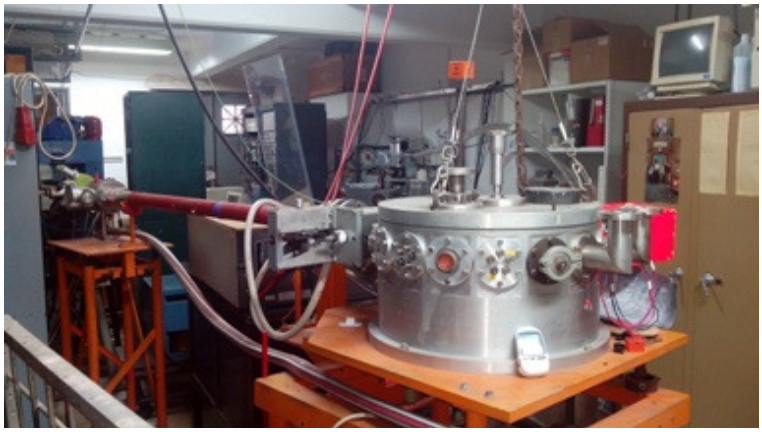


Figura 2

2. Descripción del sistema de medida

Con el fin de automatizar la medida, las distintas funciones a realizar son:

- Posicionar el detector: llevarlo a una posición determinada, cambiar de posición en décimas de grado ó en grados.
- Adquisición de datos: contar la cantidad de pulsos recibidos del detector una cantidad de tiempo pre-determinada en cada posición
- Almacenamiento de datos para posterior procesamiento.
- Graficar los pulsos contados en función de la posición del detector.
- Repetir lo anterior en varios ciclos.

Una experiencia típica es: posicionar al detector en una posición inicial, medir la cantidad de pulsos que llegan del detector, almacenar el resultado, cambiar la posición y repetir un número determinado de ciclos la experiencia graficando la evolución de los parámetros medidos, en este caso número de pulsos contados en función de la posición.

Para tal fin se desarrolló en software que automatiza la experiencia sobre un sistema de cómputo, notebook ó PC, ofreciendo al usuario una interfaz gráfica característica de los lenguajes de alto nivel. A través del puerto usb se envían comandos que son interpretados por un sistema arduino[4] que actúa como interfaz de hardware con el resto del equipo.

Los pulsos del detector, rayos x dispersados, ingresan por la entrada de interrupciones del arduino y son contados por éste, y sumados al valor contado en el ciclo anterior. Por medio de las salidas digitales se genera la excitación del motor paso a paso del detector y su sentido de giro. Se utiliza el conversor A/D del arduino para la medida de temperatura con el agregado de electrónica que adapta la señal para ser medida.

3. Conclusiones

Este equipo fue desarrollado a pedido de un grupo de Investigación del IFLP (Instituto de Física de La Plata) – Depto. de Física (Fac. Ciencias Exactas – UNLP) que investigan en el tema.

La cámara de reacción, el generador de rayos x y la etapa de potencia que excita al motor, forman parte del equipo existente en el Depto. de Física de la UNLP y se implementó la interfaz con componentes comerciales para interactuar con este equipo. Se desarrolló el software "a medida" que automatiza y controla toda la experiencia, permitiendo cumplir los requerimientos del grupo de investigación en cuanto a los distintos parámetros a medir durante la realización de la experiencia.

El Laboratorio de Electrónica atiende los requerimientos de los distintos grupos de investigación participando en el desarrollo de equipamiento auxiliar para aplicaciones específicas, requerimientos de nuevas configuraciones experimentales y apoyo en la puesta en marcha de nuevos experimentos. Dentro de este marco fue desarrollado el sistema presentado en este trabajo.

4. Referencias

[1] The Basics of Crystallography and Diffraction: Third Edition (International Union of Crystallography Texts on Crystallography) 3rd Edition. Christopher Hammond.

[2] Data Acquisition and Control Handbook. Keithley

[3] Photomultiplier tubes. Principles & applications. Philips

[4] http://dfists.ua.es/~jpomares/arduino/page_03.htm

DESCRIPTOR BASED ON SPECTRAL PEAKS CORRELOGRAMS

Brenda Lilia Valdez Reyna, Miroslav Zivanovic.
Ingeniería en Sistemas Computacionales, Ingeniería Elétrica y Eletrónico
Universidad Autónoma de Tamaulipas Campus Mante, Universidad Pública de Navarra
Blvd.E.Cárdenas Glz. 1201, 89840 Tamaulipas, México.
blvaldez@uat.edu.mx, miro@unavarra.

Abstract. We have presented in this report a new peak descriptor SD based on the lag plot and discussed its relationship to the NBD. The SD, defined as the slope of a linear regression model constructed through the data samples belonging to a time sequence, shows to be a good candidate for describing sinusoidal and noise peak classes.

A proper choice of data set is crucial for discerning correctly between the peak classes. The SD makes use only of the spectral peak shape and consequently this information can be explained by just two parameters, namely the root mean square BWrms and absolute bandwidth L. This is similar to the way the shape factor explains the filter spectral shape in the circuit theory. We have shown that BWrms and L, initially defined in the spectrum domain, hold relationship to the time duration of the data set and its sampling rate respectively.

Keywords: DESCRIPTOR, LAG PLOT, SPECTRAL

1 Introduction

This work presents the development of the application of descriptors, which allow them to know the characteristics of the signal, will be explored here the following analysis of descriptors:

1. LAG PLOT – BASICS, 2. LAG PLOT AND SPECTRAL PEAKS: THE SLOPE DESCRIPTOR (SD), 3. CHOOSING THE TIME-DOMAIN REPRESENTATION OF A SPECTRAL PEAK, 4. THE SLOPE DESCRIPTOR AND THE PARAMETERS BWrms AND L.

1. LAG PLOT – BASICS

The lag plot is a method in statistics that aims to determine whether a discrete data set is random or deterministic. For a data set x(n) the lag plot is conceived as a 2D representation of x(n) versus x(n-P), where P is a lag expressed in samples. Although P can be an arbitrary value, the lag plots are most commonly generated with P = 1. namely, the Gaussian noise and a pure tone of 100 samples each. In the case of the noise, the knowledge about the sample x(n-1) provides almost no knowledge about the following sample x(n). As for the sinusoid, the elliptical structure aligned with the diagonal (unitary slope) allows to precisely predict the next point in the data set and thus characterize the data set as deterministic.

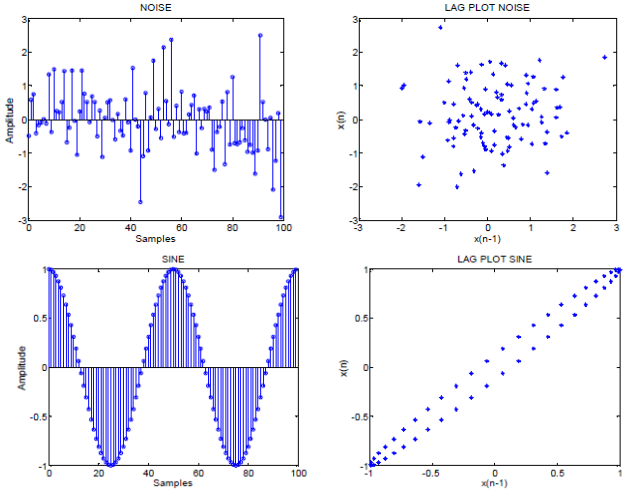


Figure 1. Lag plot representation for Gaussian and sinusoidal noise-free pattern.

2. LAG PLOT AND SPECTRAL PEAKS: THE SLOPE DESCRIPTOR (SD)

By applying a lag plot to the time domain representation of a spectral peak, we may eventually distinguish sinusoids from noise. The lag plot for the sinusoidal signal shown on Figure 1 suggests that a linear regression (least-squares) model might be appropriate for characterizing the data set. The linear regression models are simple to implement and for our case, we would need only one coefficient (parameter) that would represent the slope of the straight line fitted to the data set.

For a given data set x(n) we define the Slope Descriptor (SD) in the following way:

$$x(n) = ax(n-1) + b + e(n)\ ,\ a = SD\ . \quad (1)$$

The parameters a and b are the coefficients of the model while e(n) is the modeling error. Wemay expect a priori that the sinusoidal patterns will have the SD close to one while the noise patterns will present a wide range of values (including SD = 1). This means that this simple linear model cannot achieve a complete sine/noise separation, so we have to adapt it somehow in order to reduce/eliminate the overlap between the peak classes. Hence, the key issue is to correctly choose the time-domain representation of the spectral peaks i.e. to adapt the data set in such a way to be correctly handled by the SD.

3. CHOOSING THE TIME-DOMAIN REPRESENTATION OF A SPECTRAL PEAK

Approach 1: Complex narrow-band spectral peak and its time signal

In order to preserve the peak's central frequency we first set to zero all the bins in the spectrum except those belonging to the peak. Then we calculate the IDFT and use its real part as a data set for the lag plot. The problem with this approach is that the time domain pattern of any spectral peak would exhibit a strong deterministic behaviour. Let us recall our previous research where

we discovered that any spectral peak or, in general, a part of the spectrum with the bandwidth much smaller than the Nyquist frequency, can be considered a narrow-band process represented by a strong carrier with slowly varying amplitude and phase:

$$N(n) = A(n)\cos[\Omega_o n + \theta(n)]. \quad (2)$$

Therefore, even though the modulation parameters A(n) and $\Omega_o(n)$ are random variables, the presence of the carrier assures a deterministic behaviour of N(n). Hence, the lag plot for both sinusoidal and noise peaks will produce a tight clustering of the points along the diagonal and consequently there will be no way of distinguishing between the peak classes. As an example, we show on Figure 2 various narrow-band noise spectral peaks centred around 1.5 kHz and their corresponding real IDFTs. We observe that even though the time-domain patterns are quite different, the lag plots order the points as if they belonged to a purely deterministic signal.

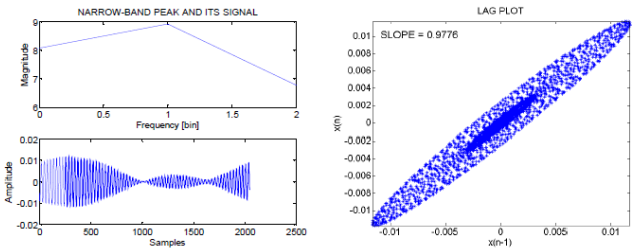


Figure 2. Various complex narrow-band noise peaks centered around 1.5 kHz, their real IDFTs and corresponding lag plots. The window is Hanning, the size of the widow is 1000 and the size of the FFT/IFFT is 2048. The IDFTs are computed by setting to zero all the bins in the FFT not belonging to the peak

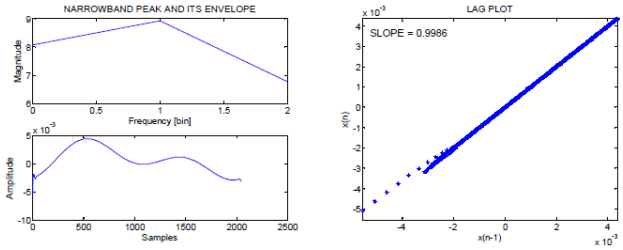


Figure 3. Various complex narrow-band noise peaks down-shifted to the origin, their real IDFTs and corresponding lag plots. The window is Hanning, the size of the widow is 1000 and the size of the FFT/IFFT is 2048. The IDFTs are computed by setting to zero all the bins in the FFT not belonging to the peak.

Approach 2: Complex narrow-band spectral peak and its time envelope

Another approach consists in first down shifting the peak to the origin (base-band) by typically multiplying the original signal with a corresponding complex exponential term. Then, like in Approach 1, the rest of the bins in the spectrum that are not included in the peak are set to zero

and the IDFT is calculated. The resulting pattern (the real part of the IDFT) is the amplitude envelope while the carrier is suppressed. For pure tones the real part of the IDFT will be exactly the analysis window. For modulated peaks it will be a product between the analysis window and the modulator of the sinusoidal signal. For out worst case sinusoidal signal the resulting pattern will still resemble the shape of the analysis window (e.g. see the windowed longest and shortest pattern for the NDD thresholds determination in our previous work). The noise peaks will produce a wide variety of patterns, which may seem beneficial for distinguishing noise from sinusoids. Unfortunately, this approach is neither good because the envelope from a noise peak is a low-pass function, i.e. the transition between the contiguous samples is quite smooth, which will again produce very "deterministic" lag plots. The same noise peaks from Figure 2 are represented on Figure 3 together with respective amplitude envelopes and lag plots. We can see at glance that the lag plots show very strong deterministic behaviour.

Approach 3: Complex broad-band spectral peak and its time signal

Therefore, we have to somehow "randomize" the time-domain representation of the noise spectral peaks. A way to achieve this is to describe spectral peaks by general broad-band, rather than narrow-band processes, i.e. the peak bandwidth is close to the Nyquist frequency. With this approach, the time-domain representations of spectral peaks would include many frequencies.

This is particularly beneficial for the noise peaks because this would allow to correctly describing randomness. For the sinusoidal peaks this approach is also good, because the deterministic signals are always correctly represented, either the model is narrow-band or broadband process.

A practical implementation of the above approach can be carried out by simply considering a spectral peak as a whole DFT, i.e. neglecting the presence of the rest of the bins in the spectrum. Then, we calculate the IDFT only over the bins in the peak and take its real part as

a data set for the lag plot.

If we take a look at Figure 4 we observe that the transitions between the contiguous samples in the real IDFTs of the noise peaks are abrupt, that is, the patterns are more "random". As a consequence, the corresponding lag plots show no apparent point clustering and the SD

will thus have a wide range of values. Unfortunately, the same effect is detected in the sinusoidal peaks. As an illustration, compare Figure 4 with Figure 5, where various spectral peaks belonging to our worst-case sinusoidal signal for the carrier frequency equal to 1.5 kHz are dealt with.

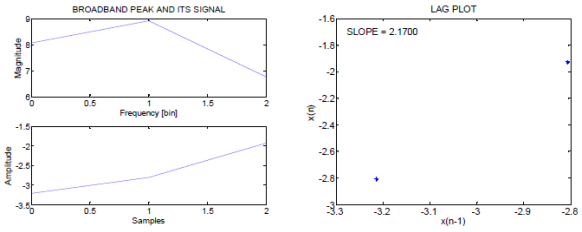


Figure 4. Various complex broad-band noise peaks, their real IDFTs and corresponding lag plots. The window is Hanning, the size of the widow is 1000 and the size of the FFT is 2048. The size of the IDFT is equal to the number of bins in the peak.

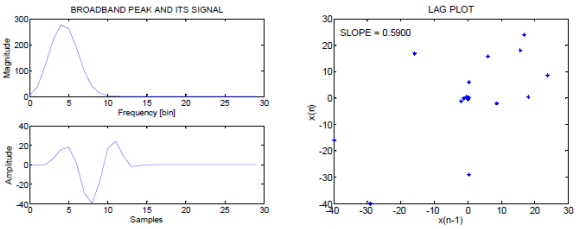


Figure 5. Various complex broad-band sine peaks from the worst case signal, their real IDFTs and corresponding lag plots. The window is Hanning, the size of the widow is 1000 and the size of the FFT is 2048. The size of the IDFT is equal to the number of bins in the peak.

Approach 4: Complex broad-band spectral peak and its time envelope

Hence, we now have to assure smoother transitions between the contiguous samples in thesinusoidal time-domain patterns, while preserving the randomness in

the noise patterns. One way to do it is to deal with the envelope of the analytical signal, rather then its real part.

Namely, an isolated spectral peak can be considered unilateral spectrum of an analytical sequence (we recall that any real sequence will give a conjugated even absolute spectrum, while analytical sequences produce unilateral absolute spectra). Then, the IDFT of a complex broadband spectral peak is an analytical sequence which may be written as:

$$y(n) = x(n) + jx\hat{}(n) \quad (3)$$

where x^(n) is the Hilbert transform of real sequence x(n). The expression (3) can also be written in the polar form:

$$y(n) = A(n)e^{j\phi(n)}, \qquad A(n) = \sqrt{x^2(n) + \hat{x}^2(n)}, \qquad \phi(n) = a\tan[\hat{x}(n)/x(n)]. \quad (4)$$

The term A(n) is the low-pass component or envelope of the analytical signal i.e. it varies more slowly than the original signal x(n). In fact, this approach is similar to Approach 2 in the sense that the carrier is suppressed and the resulting pattern is a low-pass sequence. The difference is that herein the computational resolution of the IDFT is much smaller (the size of the IDFT), because we consider only the bins in the spectral peak. Therefore, we can say that the envelope has been sampled at a very low rate. How is this fact related to the lag plot?

A lag plot describes deterministic signals properly only if the sample rate is high enough to assure smooth transitions between adjacent samples. If this condition is satisfied, all the points in the lag plot will be clustered along the diagonal. If the sampling rate is reduced, the lag plot will exhibit a number of outliers and thus the slope of the linear model is reduced. According to the lag plot the signal is less deterministic, but this is not true! It is simply that we have fewer points per sinusoidal cycle and the transitions between the samples in the data set are more abrupt.

An extreme example would be a cosine sampled at exactly Nyquist frequency, that is, 2 samples per cycle. In this case all the points in the lag plot are concentrated in only 2 outliers: (-1, 1) and (1, -1). Figure 6 illustrates this trend for a sinusoidal signal sampled at different rates.

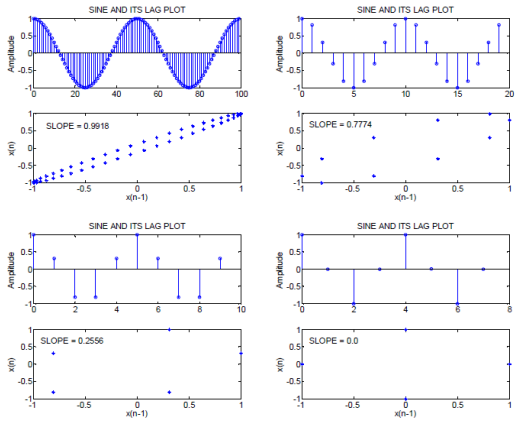


Figure 6. A sinusoid sampled at different rates: a) 50 samples/cycle, b) 10 samples/cycle, c) 5 samples/cycle, d) 4 samples/cycle, and respective lag plots.

Returning now to the issue of the time envelope, we may say that a low sampling rate will definitely randomize it up to some point. This sampling rate reduction affects

principally the noise peaks, because they are in general narrower than the sinusoidal peaks and consequently the size of the corresponding IDFT is smaller. Therefore, the IDFT from a noise peak will put

fewer points on the corresponding time envelope. The sinusoidal peaks are in general broader, so the IDFT from a sinusoidal peak will put more samples on the corresponding envelope, thus rendering it smoother and more deterministic in the sense of the lag plot. Figures 7 and 8 illustrate the benefits of this approach for the same noise and sinusoidal peaks as those in Figure 4 and 5.

In spite of the advantages of this approach, there is a shortcoming that must be dealt with. While the values of the SD for sinusoidal peaks are close to one, the values for the noise peaks may be greater or lesser than one. As a consequence, we may thus expect some overlap between the sinusoidal and noise classes in the SD domain. Therefore, we still need to improve the data set in order to best separate the peak classes.

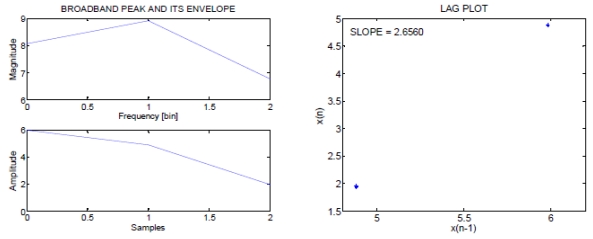


Figure 7. Various complex broad-band noise peaks, their absolute IDFTs and corresponding lag plots. The window is Hanning, the size of the widow is 1000 and the size of the FFT is 2048. The size of the IDFT is equal to the number of bins in the peak.

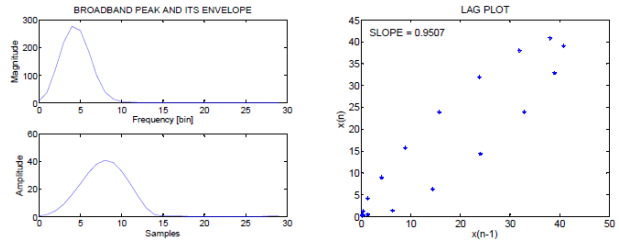


Figure 8. Various complex broad-band sine peaks from the worst case signal, their absolute IDFTs and corresponding lag plots. The window is Hanning, the size of the widow is 1000 and the size of the FFT is 2048. The size of the IDFT is equal to the number of bins in the peak.

Approach 5: Real broad-band spectral peak and its time envelope

Regarding Approach 4, we need to tighten the range of values of the SD for the noise peaks in order to reduce as much as possible the overlap between the peak classes. In fact, if we can reduce the variety of A(n) shapes for the noise peaks, it would automatically reduce the range of possible corresponding SD values. That is to say, we would let A(n) vary from peak to peak but always within some general pattern.

One way to do it is to process only the modulus of the spectral peak disregarding its spectral phase. As the spectral modulus is a real sequence, its IDFT will yield a complex sequence whose A(n) is conjugate even. The sequence A(n) is now periodic with period equal to the size of the IDFT. This periodicity will emphasize the

deterministic aspect of the timedomain representation for the noise peaks and consequently the range of possible values for the SD reduces. If we compare the SD values from Figures 7 and 9 for the same noise peaks, we see how efficiently this approach delimits the SD values to a small range close to zero. Initially, we might have thought that this approach would put the noise peaks closer to the sinusoidal peaks in the SD domain. Recall, however, Approach 4 where we saw that due to larger IDFT the sinusoidal peaks yield smoother A(n) and thus more deterministic lag plots. Hence, the sinusoidal peaks will still have larger SD values (as an illustration, compare Figures 8 and 10) although the range of values is somewhat enlarged.

In view of the aforementioned discussion, we choose this approach as a reference for generating the data sets for the spectral peaks' lag plots. In the following sections we will prove analytically the hypothesis posed herein and show the benefits of this approach for the peak classification.

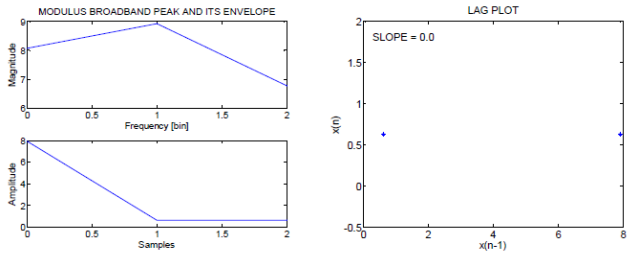


Figure 9. Various real broad-band noise peaks, their absolute IDFTs and corresponding lag plots. The window is Hanning, the size of the widow is 1000 and the size of the FFT is 2048. The size of the IDFT is equal to the number of bins in the peak.

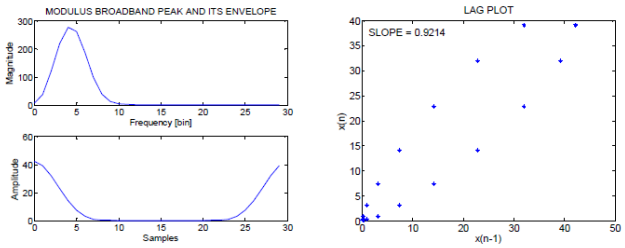


Figure 10. Various real broad-band sine peaks from the worst case signal, their absolute IDFTs and corresponding lag plots. The window is Hanning, the size of the widow is 1000 and the size of the FFT is 2048. The size of the IDFT is equal to the number of bins in the peak.

4. THE SLOPE DESCRIPTOR AND THE PARAMETERS BWrms AND L

Let us first recall the nomenclature we used in the previous research, because we are going to apply it herein too. The parameter BWrms was referred to as the root mean square bandwidth of a spectral peak expressed in bins. We called the parameter L the absolute bandwidth of a peak, being equal to the number of bins in the peak. We can now proceed to the discussion about the relationship between the SD and these spectral peak parameters.

The SD is the slope of a linear regression model which we have already defined in Section 2. For the sake of clarity we repeat herein the equation (1):

$$x(n) = ax(n-1) + b + e(n)\ ,\ a = SD\ . \quad (5)$$

The expression (5) represents a first order linear predictor i.e. a first order AR model. Recalling the origin of x(n), we may say that a real broad-band spectral peak is the spectral estimate of the following LTI constant coefficient model H(z):

$$H(z)=\frac{1}{1-az^{-1}} \quad , \qquad z=e^{j\Omega} \quad . \tag{6}$$

In the frequency domain the expression (6) becomes:

$$H(e^{j\Omega})=\frac{1}{1-ae^{-j\Omega}}=\frac{1}{1-a\cos\Omega+ja\sin\Omega} \quad . \tag{7}$$

The last expression says that a real broad-band spectral peak can be represented as a LTI lowpass system with one real pole which can be inside (a < 1), outside (a > 1) or onto the unitary circle (a = 1). The parameter a determines the sharpness of the peak and thus the bandwidth of the system. Therefore, we may expect the smallest bandwidth when the pole is on the unitary circle.

In order to estimate the bandwidth BWrms of the system, we can use the well-know 3dB criterion:

$$H(\Omega_c)=\frac{1}{\sqrt{2}}H(0)=\frac{1}{|1-a|\sqrt{2}}\times\frac{1}{\sqrt{1-2a\cos\Omega_c+a^2}} \quad , \tag{8}$$

where Ω_c is the cut-off frequency. From (8) we can express the cut-off frequency as:

$$\Omega_c=\cos^{-1}\left(-\frac{a^2-4a+1}{2a}\right). \tag{9}$$

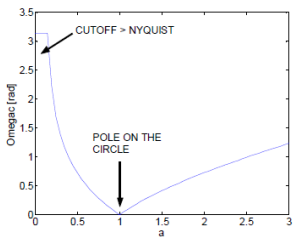


Figure 11. The cut-off frequency Ω_c versus the slope *a*. For a < 0.15 approximately, the bandwidth is greater than the Nyquist interval: we have aliasing.

If we put $BW_{rms}=2\Omega_c$ and $a=$ SD in (9) then we get straightforwardly an approximate relationship between the bandwidth and slope descriptor of a spectral peak. The expression (9) is plotted on Figure 11 showing the dependence of a on Wc. We observe that, according to our expectations, for *a = 1* the bandwidth is theoretically zero while otherwise the bandwidth increases as the pole steps off the unitary circle. For small *a* the bandwidth spreads out of the Nyquist interval, giving rise to the aliasing effect. So, how is all this related to the peak classification?

Let us recall that in order to yield a deterministic characterization from the lag plot, the transitions between the contiguous samples in a time sequence must be smooth enough. We have already seen in the previous section that this condition can be met by having the sampling rate (the size of the IDFT) large enough. According to Approach 5, it means that larger peaks (L large) will have large SD independently of the shape of its time envelope.

However, a large SD can also be achieved with smaller L but larger time envelope, if we recall from the Fourier theory that a small BWrms correspond to a large time response and vice

versa. This is conceptually similar to the process of decimating. For example, if a time sequence is decimated with a factor 2, then its bandwidth enlarges two times. But at the same time this enlargement is compensated by cutting down the number of bins in the DFT by two (assuming that we don't use zero padding). As a result, the number of DFT bins within the bandwidth remains the same.

We may now take a look at Figure 12 where various real broadband spectral peaks from the worst case signal and the corresponding time envelopes are shown. The envelopes are represented versus the normalized length, so that the effect of L is suppressed and only the effect of BWrms remains. We observe that both small BWrms and L produce almost the same shape of the envelope as larger BWrms and L, thus yielding similar SD. This issue is very important for the description of the sinusoids, because this approach always tends to compress the sinusoidal class by reducing the SD of the larger peaks (more modulation) and enlarging the SD of the narrower peaks (less modulation).

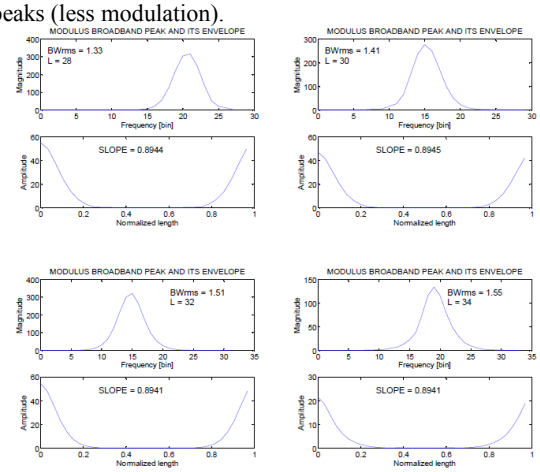


Figure 12. Various real broadband peaks from the worst-case signal and the corresponding envelopes. Normalized length is given as a sample index divided by the length of the envelope sequence.

However, this trend is also present in the noise peaks, because for any peak in general a large L corresponds to a large BWrms and the other way around. Fortunately, the relationship between those parameters is quite different for sinusoids and noise. Figure 13 shows the joint BWrms-L histograms for both peak classes (the axis are the same in both plots in order to facilitate the visual comparison).

We see that the noise histogram exhibit very strong correlation between the parameters, because the dependence region follows a steep linear trend and thus the parameters change proportionally. This is in fact very beneficial for the peak classification, because it will not allow the noise peaks go beyond a certain value of the SD. Regarding the sinusoidal peaks, the correlation is much weaker and doesn't seem to obey a linear trend. As a consequence, the beneficial trend (increments in L) is much more emphasized than the non-beneficial trend (increments in BWrms). Having in mind that the sinusoidal peaks are in general larger than the noise peaks, this approach will tend to push the sinusoidal peaks towards larger values of SD, that is, away from the noise peaks. In the next section we will show the histograms for the peak classes in the SD domain and establish a comparison between the SD and NBD.

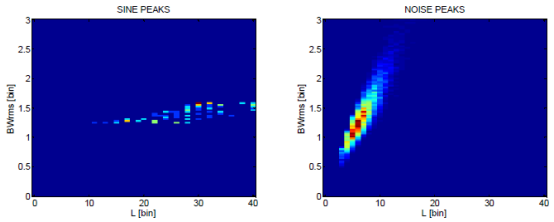


Figure 13. Joint BW_{rms}-L histograms for sinusoidal and noise peaks. For the sinusoidal peaks we used our noise-free worst case signal and for the noise peaks the Gaussian noise was used. The analysis window was the Hanning.

6. CONCLUSIONS

We have presented in this report a new peak descriptor SD based on the lag plot and discussed its relationship to the NBD. The SD, defined as the slope of a linear regression model constructed through the data samples belonging to a time sequence, shows to be a good candidate for describing sinusoidal and noise peak classes.

A proper choice of data set is crucial for discerning correctly between the peak classes. The SD makes use only of the spectral peak shape and consequently this information can be explained by just two parameters, namely the root mean square BWrms and absolute bandwidth L. This is similar to the way the shape factor explains the filter spectral shape in the circuit theory. We have shown that BWrms and L, initially defined in the spectrum domain, hold relationship to the time duration of the data set and its sampling rate respectively.

We have next shown that the mutual action of those parameters determines the smoothness of the data set and consequently its lag plot interpretation. In order to yield a deterministic interpretation, BWrms should be as small as possible and L as large as possible. This however does not correspond to a realistic scenario because those parameters are mutually coupled. However, this relationship is less strict for the sinusoidal peaks and thanks to this fact the sinusoidal class can be well distinguished from the noise.

Finally, we have shown that, regarding BWrms and L, the NBD behaves in a similar way the SD does but in the opposite direction. By inverting the roles of the parameters in the NBD we obtain a new description of the peak classes that resembles the SD. This shows that two apparently very different spectral peak descriptors, depending on the same parameters in a similar way, provide for a similar description of the peak classes. Their dependence on BWrms and L is of course not the same but it follows the same trends.

Referencias

[1] F. Plante, G. Meyer and W.A.Ainsworth, “Improvement of Speech Spectrogram Accuracy by the Method of Reassignment”, IEEE Transactions on Speech and Audio Processing. Vol.6 No.3, Mayo 1998.

[2] L. Cohen, “Time-frequency analysis”, Prentice Hall Signal Processing Series, 1995.

[3] S. Orphanidis, “Introduction to signal processing”. Prentice Hall International Editions,1996.

[4] C.Hory, N. Martin and A. Chehikian, “Spectrogram Segmentation by Means of Statistical Features for Non-Stationary Signal Interpretation”, IEEE Transactions on Signal Processing, Vol. 50, No.12, Diciembre 2002.

[5] C.Hory, N. Martin, “Mixture densities formulation of a spectrogram segmentation task”, Proceedings of EUSIPCO 2002, pp. 427-430, 3-6 September 2002.

[6] Miroslav Zivanovic, “Detection of Non-Stationary Sinosoids by Using Joint Frequency Reassignment and Null to Null Bandwith”, Proceedings of EUSIPCO 2008, Agosto 2008.

Diseño Robusto basado en PSO de un Controlador PID para Sistemas con Retardo de Tiempo Variable: Aplicación a AQM

Patricia N. Baldini[1,2], Guillermo L. Calandrini[2]
[1] Departamento deElectrónica, Facultad Regional Bahía Blanca , Universidad Tecnológica Nacional
11 de Abril 461- Bahía Blanca

[2] Departamento de Ingeniería Eléctrica y de Computadoras, Universidad Nacional del Sur
Av. Alem 1253-Bahía Blanca
pnbaldi@frbb.utn.edu.ar; calandri@criba.edu.ar

Resumen. En este trabajo se plantea el control robusto de sistemas modelados con retardos de tiempo variables. Los parámetros del controlador del tipo proporcional-integrativo-derivativo (PID) se determinan en forma automática. La metodología adoptada integra el algoritmo heurístico de enjambre de partículas al procedimiento de diseño robusto de la Teoría de Realimentación Cuantitativa (QFT) para sistematizar la etapa de *loop shaping*. El objetivo del control es lograr un buen compromiso entre estabilidad robusta, seguimiento robusto y rechazo de perturbaciones, con mínimo esfuerzo de control. El algoritmo seleccionado resulta de fácil implementación, con características de convergencia estable y buena eficiencia computacional. En particular, se presentan los resultados del diseño de un sistema de gestión activa de colas (AQM) para mejorar el control de congestión y la calidad de servicio en redes de comunicación TCP.

Palabras Clave: Optimización Heurística, PSO, Respuesta en Frecuencia, Control PID, QFT, AQM.

1 Introducción

En muchas aplicaciones de ingeniería se encuentran sistemas con retardos de tiempo producto ya sea de la transmisión de material o de información. Algunos ejemplos típicos son procesos químicos, sistemas de comunicaciones, sistemas de potencia y, en general, todo sistema de control basado en redes de comunicación. Los retardos de tiempo tienen un impacto negativo en el comportamiento del sistema y comprometen la estabilidad [1]. Su tratamiento es complejo en especial cuando los retardos varían en el tiempo o tienen un valor incierto. En general es posible describirlos mediante modelos equivalentes de naturaleza estadística o determinista que facilitan su estudio. Para el caso determinista en particular, la teoría de control robusto permite representarlos utilizando el concepto de incertidumbre y diseñar de controladores que

garantizan el cumplimiento de los criterios de desempeño cualquiera sea su valor dentro del rango de variación esperado.

En este trabajo se propone el uso de una metodología robusta de sintonización automática de controladores PID para sistemas con retardos de tiempo variables o inciertos representados mediante el modelo de función transferencia. Se combina la Teoría de Realimentación Cuantitativa (QFT) [2] con el algoritmo de optimización heurística de Enjambre de Partículas (PSO) [3] para la determinar el controlador PID robusto que permite cumplir con objetivos múltiples de un modo óptimo en el sentido de Pareto [4]. Se especifican estabilidad robusta, propiedades de seguimiento, rechazo de perturbaciones y reducción de la sensibilidad a la variación de las condiciones de operación. Mediante PSO se automatiza la etapa denominada *loop shapping*, es decir, la sintonización propiamente dicha de los parámetros del controlador sin sobrediseño. Se logra mayor eficiencia del controlador y, por lo tanto, mejores resultados con estructuras simples como la del PID. Las características y la complejidad del problema no permiten el uso de técnicas de optimización tradicionales de modo que los algoritmos de búsqueda inteligente inspirados en la naturaleza, se presentan como una alternativa efectiva. Tanto QFT como PSO fueron seleccionados siguiendo el criterio de mantener la claridad, simplicidad y versatilidad de los procedimientos adoptados y por los buenos resultados que se han reportado en diversas aplicaciones de control [5]-[7]. El objetivo de QFT es la síntesis de un controlador lo más simple posible, con ancho de banda mínimo, que satisfaga las especificaciones con el menor costo de realimentación teniendo en cuenta la incertidumbre del modelo.

El control PID representa la estrategia de control con mayor aplicación industrial. A pesar de su simplicidad, permite resolver satisfactoriamente una gran variedad de problemas complejos [6],[8].

El método presentado se aplica al diseño de un esquema alternativo de AQM para la prevención de congestión y la optimización de la calidad de servicio (QoS) en redes de comunicación basadas en TCP. El objetivo, desde el punto de vista de la teoría de control, es optimizar la utilización del enlace independizando al sistema de la carga de trabajo y reduciendo el efecto de los retardos variables de transmisión. La robustez del control se verifica en base al modelo no lineal, no estacionario, basado en flujo de fluido propuesto por Misra *et al.* [9] con variación aleatoria de parámetros. Se consideran variaciones en las características de operación de la red como número de sesiones activas, capacidad de enlace, RTT.

1.1 Conceptos de QFT

La Teoría de Realimentación Cuantitativa introducida por Horowitz [2] propone una metodología de diseño robusto en el dominio de la frecuencia basada en la arquitectura de realimentación estándar de la Fig. 1 (a). Permite cumplir las especificaciones de comportamiento preestablecidas dentro de una región de incertidumbre determinada a priori durante el modelado del sistema. En relación a otros métodos de control robusto presenta una mayor transparencia en el proceso de diseño que permite relacionar de antemano la complejidad del controlador con la factibilidad de los objetivos. Por otra parte, tiene en cuenta cuantitativamente el costo de la realimentación en relación a la incertidumbre.

A partir del modelo de función transferencia del proceso o sistema, QFT contempla el hecho de que los parámetros de la misma, ganancia, polos y ceros, pueden variar dentro de rangos finitos conocidos. Esto conduce a considerar una región dentro del espacio paramétrico asociada a la incertidumbre, Θ, y una familia de Funciones Transferencia, $\{ P(j\omega,\boldsymbol{\theta}),\boldsymbol{\theta}\in\Theta\}$. Una de ellas se adopta como el modelo de la planta nominal, P_0 ($j\omega$), y se utiliza como la referencia para del diseño. La incertidumbre se incluye en los objetivos a alcanzar [5].

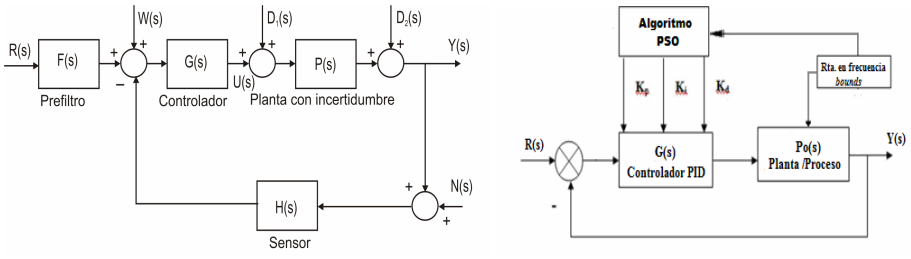
En la Fig. 1 (a) se muestra una configuración típica de control con dos grados de libertad, donde $P(s)$ representa la planta con incertidumbre y $H(s)$ representa la dinámica del sensor. Las perturbaciones están modeladas por los procesos W, D_1 y D_2. La existencia de ruido se incorpora con N, la señal de referencia es R y el objetivo del control es la salida Y.

En QFT el controlador $G(s)$ debe compensar el efecto de la incertidumbre. El precompensador $F(s)$ puede ser incluído para un ajuste final.

Para la planta P_0 (s), su *template* es definido como el conjunto de respuestas en frecuencia posibles asociadas al espacio de incertidumbre paramétrica. Las especificaciones cuantitativas de estabilidad, comportamiento temporal y rechazo o atenuación de perturbaciones se expresan analíticamente mediante restricciones, en el rango de frecuencias de operación, sobre el módulo de las familias de funciones trasferencia que relacionan distintas variables del lazo de la Fig. 1 (a). Algunos ejemplos se muestran en la Tabla 1, donde $L(j\omega+\boldsymbol{\theta}) = P(j\omega+\boldsymbol{\theta})G(j\omega)H(j\omega)$ es la familia de funciones transferencia de lazo abierto. Estas restricciones se representan como regiones admisibles en un plano complejo módulo-fase conocido como carta de Nichols, para la función de lazo abierto nominal buscada, $L_0(j\omega)\, e^{-j\omega\tau} = G(j\omega)\, P_0(j\omega)\, H_0(j\omega)\, e^{-j\omega\tau}$ .Estas regiones quedan limitadas, a cada frecuencia de trabajo, por un conjunto de curvas denominadas *bounds.*

El diseño consiste en determinar el controlador, $G(j\omega)$, tal que $L_0(j\omega)\, e^{-j\omega\tau}$ satisfaga las restricciones establecidas por los *bounds* lo más ajustadamente posible para evitar el sobrediseño, manteniendo la ganancia de alta frecuencia y al ancho de banda lo menores posibles. Si esto se logra, se garantiza que también lo harán todas las funciones de lazo correspondientes al *template.*

Desde el punto de vista del cumplimiento óptimo de las especificaciones, el problema es de objetivos múltiples con más de una solución posible en el sentido de Pareto [4].



(a) (b)

Fig. 1: Esquemas: (a) del sistema de control general y (b) del método de sintonización del PID

El uso de los *bounds* tiene la ventaja de que se mantienen constantes durante el proceso de optimización iterativa, por lo que se reducen los cálculos.

Tabla 1. Tipo de especificaiones incluídas en el diseño robusto mediante QFT

Tipo de Especificación	Restricción
Estabilidad Robusta	$\left\| \frac{1}{1+L(j\omega+\theta)e^{-j\omega(\tau+\Delta\tau)}} \right\| \le \delta_1 \,, \omega \in \Omega_1$
Seguimiento Robusto	$\delta_m(\omega) \le \left\| \frac{P(j\omega+\theta)G(j\omega)}{1+L(j\omega+\theta)e^{-j\omega(\tau+\Delta\tau)}} \right\| \le \delta_M(\omega) \,, \omega \in \Omega_1$
Rechazo de perturbaciones D_1	$\left\| \frac{P(j\omega+\theta)}{1+L(j\omega+\theta)e^{-j\omega(\tau+\Delta\tau)}} \right\| \le \delta_2(\omega) \,, \omega \in \Omega_2$

1.2 Revisión del Algoritmo de Optimización por Enjambre de Partículas

El control robusto implica una formulación compleja, altamente no lineal y con un espacio factible de soluciones no convexo. Para lograr resultados óptimos en este tipo de problemas se han propuesto numerosas metodologías heurísticas de búsqueda inteligente. Entre ellas se cuentan los algoritmos que imitan fenómenos de la naturaleza, basados en poblaciones. Desde un punto de vista general, se considera un conjunto de individuos distribuidos dentro del espacio de parámetros factibles. En las sucesivas iteraciones los individuos se desplazan intentando llevar adelante de modo efectivo la exploración del espacio en la búsqueda del óptimo. Esto se logra contemplando tres pasos en cada iteración realizados en la mayoría de los casos en forma estocástica: auto-adaptación o mejora de su propia performance, cooperación donde todos los miembros contribuyen en la transferencia de información y competición según el suceso alcanzado.

Dentro de estas opciones, se ha comprobado que PSO resulta de gran utilidad en el diseño de controladores. Gracias a la claridad de su funcionamiento y el reducido número de parámetros propios se logra un buen de la control sobre la convergencia [10]-[11]. Propuesto por Kennedy and Eberhart [3], representa un procedimiento de búsqueda que reproduce la interacción social entre integrantes de un grupo de una misma especie para cumplir un objetivo, tal como sucede en las bandadas de pájaros o enjambre de abejas. Ese comportamiento social se basa en la transmisión del suceso de cada individuo al resto del grupo, lo que resulta en un proceso sinérgico que permite a los integrantes lograr de la mejor manera posible un objetivo común.

En PSO se asocia a cada individuo de una población de tamaño fijo, una posición que representa un valor posible del vector de parámetros incógnita dentro del espacio multidimensional de búsqueda. Las posiciones inicialmente se asignan en forma aleatoria y se van modificando con una velocidad que se ajusta dinámicamente teniendo en cuenta la experiencia individual y la información compartida por el resto del grupo. En todo caso, la mejor posición alcanzada, ya sea por el conjunto o por cada integrante, representa el conjunto de parámetros con el que se obtuvo el menor valor de la función objetivo a minimizar.

Si $\boldsymbol{X}$ y $\boldsymbol{V}$ definen los vectores de posición y velocidad en el espacio paramétrico *n*-dimensional de búsqueda, *N* el tamaño de la población, $\boldsymbol{P}_{best}$ y $\boldsymbol{G}_{best\text{las}}$ las mejores posiciones alcanzadas por cada individuo y por el grupo, respectivamente, la evolución dinámica o actualización de las posiciones y velocidades en la *k*-ésima iteración se describen mediantes las siguientes expresiones vectoriales:

$$\boldsymbol{V}^i_{k+1} = w_k\, \boldsymbol{V}^i_k + C_1\, r_1\, (\boldsymbol{P}^{\,i}_{\,best} - \boldsymbol{X}^i_k) + C_2\, r_2\, (\boldsymbol{G}^{\,i}_{\,best} - \boldsymbol{X}^i_k) \qquad (\mathbf{1})$$

$$\boldsymbol{X}^i_{k+1} = \boldsymbol{X}^i_k + \boldsymbol{V}^i_{k+1} \qquad (\mathbf{2})$$

$$w_{k+1} = w_{máx} - (w_{máx} - w_{mín})\, k/k_{máx} \qquad (\mathbf{3})$$

donde: $1 \le i \le N$, w_k es la inercia. Este parámetro rige la capacidad de exploración en el espacio de búsqueda. En este caso se lo considera variable linealmente entre un máximo inicial $w_{máx}$ y un mínimo w_{min} al alcanzarse el máximo de las iteraciones permitidas, $k_{máx}$. Los dos últimos términos en la expresión (1) representan las inteligencias individual y colectiva, con C_1 y C_2 los factores cognitivo y social, r_1 y r_2 factores de peso aleatorios distribuidos uniformemente en (0,1).

1.3 Optimización Multiobjetivo en el Marco de QFT

La estructura general del controlador PID a considerar incluye un término proporcional, otro integrativo y otro derivativo al que se le incorpora un polo de alta frecuencia para evitar amplificación de ruido. Su función transferencia resulta

$$G_{PID}(s) = K_p + \frac{K_i}{s} + \frac{K_d\, s}{\frac{K_d}{T} s + 1} \qquad (\mathbf{4})$$

La dimensión del espacio paramétrico es n = 4 y la posición $\boldsymbol{X} = [K_p\ K_i\ K_d\ T\ ]^{\mathrm{T}}$ es el vector de parámetros. La función a minimizar o función objetivo incluye propiedades de estabilidad robusta, seguimiento robusto y rechazo de perturbaciones, a través de la distancia de la función transferencia de lazo abierto a los *bounds*, y restricción del ancho de banda, mediante la ganancia de alta frecuencia, según la expresión (5):

$$f(\boldsymbol{X}) = \gamma_1\, 20\, \log(K_{HF}) + \gamma_2 \sum_{k=1}^{n_f} f_{bdb}(\omega_k) + \gamma_3 \sum_{k=n_{LIM}}^{n_f} f_{UHF}(\omega_k) \qquad (\mathbf{5})$$

$$f_{bnd}(\omega_k) = d_k \quad , \quad k = 1,2,\cdots,n_f \ , \qquad f_{UHF}(\omega_k) = d_{Lk}\, \delta \quad , \quad k = n_{\lim}, n_{\lim}+1, \cdots, n_f$$

$$\delta = \begin{cases} 0 & \text{si se cumple con la condición} \\ 1 & \text{si no se cumple} \end{cases}$$

donde d_{k} y $d_{L\mathrm{k}}$ denotan las distancias entre la función trasferencia de lazo abierto nominal, $L_0(j\,\omega_k)$, y el *bound* correspondiente a ω_k, y entre la función trasferencia de lazo abierto nominal para $\omega_k \ge \omega_{Lim}$ ($k \ge k_{lim}$) y el denominado *bound* de alta frecuencia universal (UHB) respectivamente. Esta condición garantiza un buen comportamiento a alta frecuencia. K_{HF} es la ganancia de alta frecuencia o costo de

realimentación, es decir, $\lim_{\omega\to\infty} |L(j\omega)| = \lim_{s\to\infty} \frac{K_{HF}}{\omega^{m+2-r}}$; γ_i son factores de peso ; n_f es el número de frecuencias consideradas dentro del rango de trabajo.

Se incluyen un par de restricciones, g_1, g_2 que garantizan la estabilidad durante el proceso y el cumplimiento de los *bounds*. Como el denominador de la función transferencia de lazo cerrado resulta un cuasi polinomio, se recurre, para simplificar la verificación de estabilidad, a aproximar la exponencial compleja por un cero en el semiplano derecho, (1-$s\tau$). De este modo, se define g_1 teniendo en cuenta que un sistema lineal e invariante en el tiempo es estable si las raíces de su polinomio característico son reales y negativas o complejas con parte real negativa,

$$g_1(\boldsymbol{X}) = \max_{1\le i\le m+2-r} \left(Re(p_i),0\right) \le 0$$
$$g_2(\boldsymbol{X}) = \max_{1\le i\le n_f} \left(d_k,0\right) \le 0 \qquad (\mathbf{6})$$

El tratamiento de las restricciones y los límites de búsqueda se realiza mediante el método conocido como penalización y el criterio para la convergencia tiene en cuenta la invarianza del mejor valor hallado de la función objetivo, dentro de la tolerancia numérica considerada como aceptable, durante un número dado de iteraciones. Un esquema de la metodología propuesta se muestra en la Fig. 1 (b).

2 Diseño de un PID para Control de Congestión Basado en AQM

Para la gestión de colas en los routers TCP con el objetivo de evitar la congestión sin esperar la información remota, se han propuesto diversas técnicas. El problema frecuente es que cada configuración solo resulta adecuada para condiciones de tráfico determinadas. De modo que el diseño de un algoritmo de control PID en el marco de la teoría de control robusto se presenta como una buena alternativa ya que tiene en cuenta condiciones de operación variables a través de la incertidumbre [12]-[15]. A continuación se presentan los detalles del diseño desde este enfoque, con la metodología QFT-PSO propuesta en este trabajo.

2.1 Modelo dinámico del flujo en TCP

Un modelo basado en flujo de fluidos y representado por ecuaciones diferenciales estocásticas fue propuesto en por Misra *et al.* Describe la dinámica en TCP durante el modo de prevención de congestión. En este trabajo se adopta la versión simplificada que ignora los mecanismos de inicio lento y tiempo de espera. Este modelo relaciona el valor medio de las variables fundamentales de la red y se describe mediante un sistema de dos ecuaciones diferenciales no lineales acopladas variantes en el tiempo. El detalle del modelo puede verse en [9] y un diagrama en bloques, en la Fig. 2.

Un diagrama en bloques se muestra en la Fig. 3 donde: W *:*tamaño promedio de la ventana TCP (paquetes); q *:* longitud promedio de la cola (paquetes); R *:* RTT (seg); C *:* capacidad de enlace (paquetes/seg); T_p *:* retardo de propagación (seg); N *:* factor

de carga; p : probabilidad de marcado de paquetes ($p \in (0,1)$).

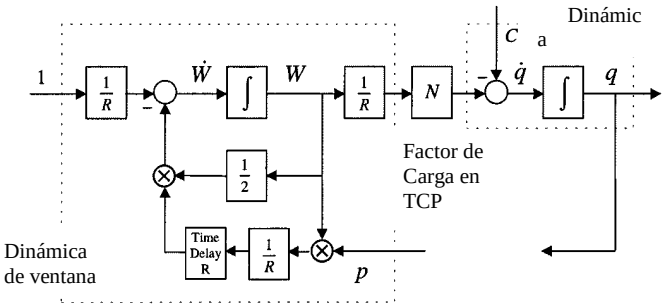


Fig. 2: Diagrama en bloques del modo de prevención de congestión en TCP

Para la linealización del modelo en torno a un punto de operación, primero se obtienen ecuaciones invariantes en el tiempo bajo las siguientes hipótesis: los retrasos temporales se suponen constantes e iguales a R_0; el punto de operación satisface las ecuaciones no lineales variantes en el tiempo; el número de sesiones TCP y la capacidad de enlace son constantes, es decir, $N(t) = N$ = cte. y $C(t) = C$ = cte. [9]

El punto de operación en estado estacionario (W_0, q_0, p_0), se determina haciendo $\dot{W} = 0$, $\dot{q} = 0$ resultando las siguientes relaciones: $W_0^2\, p_0 = 2$, $R_0 = \frac{q_0}{C} + T_p$, $W_0 = \frac{R_0 C}{N}$

El modelo lineal resultante se esquematiza en la Fig. 3 (a) donde: $\delta W(t) = W(t) - W_0$, $\delta q(t) = q(t) - q_0$, $\delta p(t) = p(t) - p_0$ son perturbaciones respecto del equilibrio y δp es la acción de control.

El modelo lineal se descompone en una parte considerada como la planta nominal del sistema de control conteniendo el retardo y las dinámicas de cola y de ventana. La otra se considera incertidumbre de alta frecuencia y se la incorpora como rechazo de perturbaciones [9], [18] según las expresiones (8), como se muestra en la Fig. 3 (b).

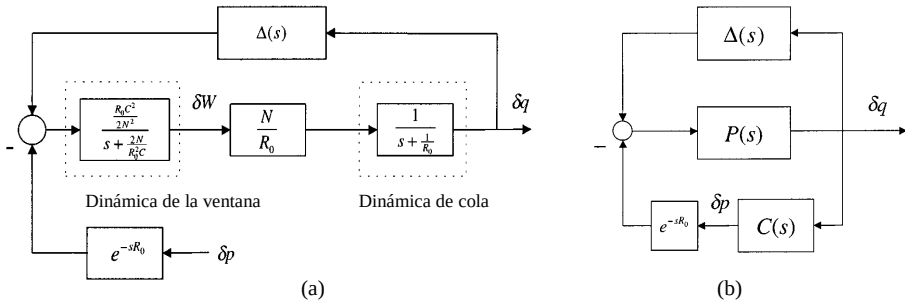


Fig. 3: Diagramas en bloques del sistema linealizado con incertidumbre de alta frecuencia.

$$P(s)e^{-Rs} = \frac{\frac{RC^2}{2N}}{\left(s + \frac{2N}{CR^2}\right)\left(s + \frac{1}{R}\right)} e^{-Rs} \qquad \Delta(s) = \frac{2N^2 s}{R^2 C^3}\left(1 - e^{-Rs}\right) \qquad (8)$$

La familia de plantas con diferente ganancia y polos acorde a las condiciones de

operación de la red, determina la incertidumbre paramétrica:

$$100 \le N \le 150\,,\ 3650 \le C \le 3850\,,\ 0.150 \le R \le 0.246\,.$$

Las especificaciones impuestas en este trabajo según la Tabla 1, con $H_0(j\omega) = 1$ y teniendo en cuenta que $\left|e^{-j\omega\tau}\right| = 1$, fueron :

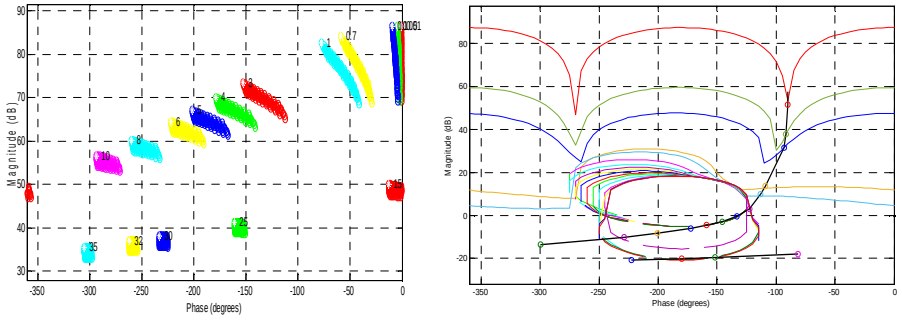
$$\Omega_1 = \{\omega: 0.01 \le \omega \le 35\}, \qquad \Omega_2 = \{\omega \in \Omega_1 : \omega \ge \omega_{LIM} = 15\}\,, \qquad \delta_2 = 1.2,$$

$$\delta_m(\omega) = \left|\frac{2}{(s+0.5)(s+1)(s+4)}\right|, \delta_M(\omega) = \left|\frac{0.7\,(s+1)}{s^2+1.306s+0.7}\right|, \ \delta_1 = \alpha\left|\Delta(j\omega)\right|^{-1} < 0.1\left|\frac{C^3R^3}{4N^2}\frac{(s+10)}{s}\right|$$

Los parámetros para PSO adoptados fueron: tamaño de la población= 60, C_1 =1.2, C_2 = 0.5, $0.4 \le w \le 0.9$. El número de iteraciones promedio para la convergencia resultó de 320 para una tolerancia de 10^{-8}.

En la Fig. 4 (a) se observan los *templates* mostrando el grado de dispersión en las respuestas en frecuencia producto de la incertidumbre, y (b) la respuesta en frecuencia de lazo abierto nominal resultante después del diseño, sobre el conjunto de *bounds*.

Los parámetros del controlador y medidas de desempeño temporal y frecuencial logrados son: $[K_p \ \ K_i \ \ K_d \ \ T\ ]^T$ = $[1.8999\text{x}10^{-4} \ \ 17983\text{x}10^{-4} \ \ 4.2472\text{x}10^{-5} \ \ 100]^T$; Los márgenes de fase y ganancia y la frecuencia de cruce resultaron: MF = 50.8 °, MG = 6.7 dB, ω_0 = 2.8 *rad/seg*; el tiempo de respuesta y el sobrepico : t_s = 5.7 seg, M_p=11%



(a) (b)

Fig. 4: (a)*Templates* y (b) respuesta en frecuencia de lazo abierto con el PID sobre *bounds*

2.2 Comprobación de Robustez

La robustez se verifica a partir del modelo no lineal, no estacionario, implementado en Simulink®. Se contemplan condiciones de variación aleatoria en los parámetros que describen la operación de la red siguiendo las establecidas en [17]-[18] y graficadas en la Fig. 6. Se considera que el buffer es suficientemente grande para evitar overflow y un tamaño medio de paquete de 500 Bytes. Los resultados se comparan con los obtenidos con el control PI propuesto en [18]. Se observa una respuesta más rápida y que los valores promedio de cola y del RTT se mantienen más estables en el valor establecido con el control propuesto.

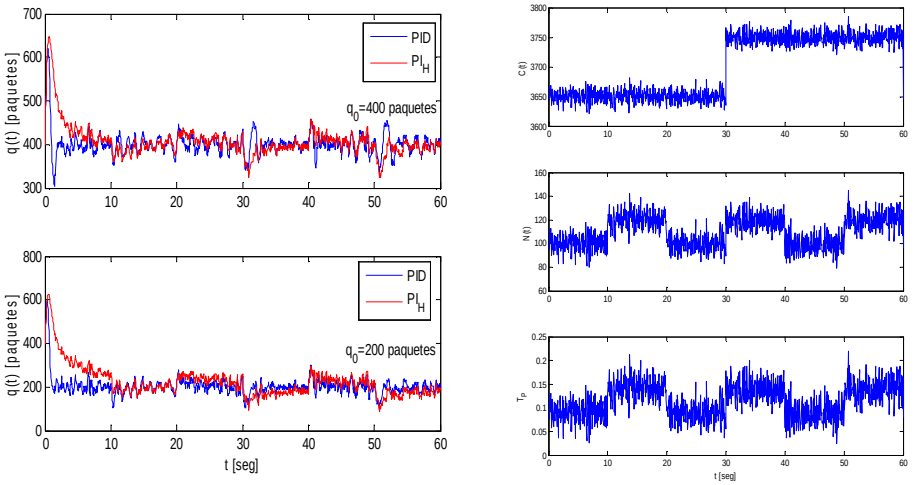


Fig. 6: Longitud de cola instantánea con el controlador para q_0= 200 y q_0= 400 paquetes

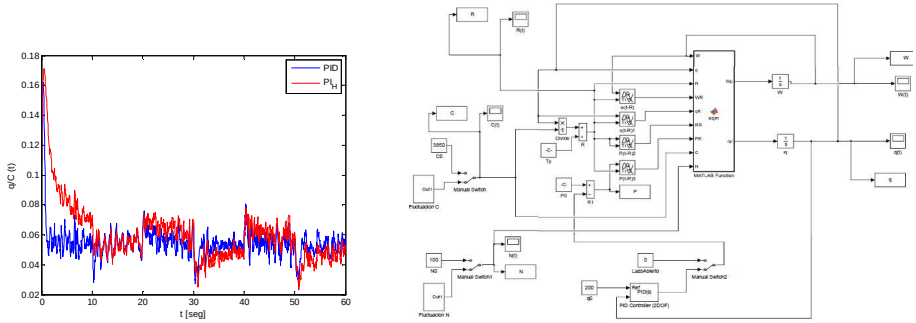


Fig. 7: RTT instantáneo con el controlador para q_0= 200 y modelo no lineal en Simulink

3 Conclusiones y trabajos futuros

En este trabajo se propuso una metodología robusta de diseño de controladores PID para sistemas con retardo de tiempo variable y acotado, basada en QFT y automatizada con PSO. Esta combinación permite obtener las mejores prestaciones del controlador adoptado, cumpliendo las especificaciones de diseño aún en los peores casos determinados por la incertidumbre. El algoritmo de optimización heurística converge a una de las posibles soluciones óptimas con una dispersión reducida medida por la varianza del valor de convergencia del vector de parámetros. La aplicación a un esquema de control activo de colas en redes condujo a buenos resultados frente a variaciones esperables de las condiciones de operación. Se tiene en cuenta tanto los criterios de desempeño propios de la teoría de control como los de análisis de redes. La comparación con resultados publicados permite concluir la efectividad de la metodología. El comportamiento fue verificado por simulación

usando el modelo no lineal variable en el tiempo, usando SIMULINK®. La naturaleza estocástica del proceso se tiene incluye en la variación aleatoria de los parámetros de la red. Se espera obtener resultados igualmente satisfactorios con el uso de simuladores de redes tales como *ns*2.

Referencias

1. Saadaoui,K.; Testouri,S.; Benrejeb ,M.: Robust stabilizing first-order controllers for a class of time delay systems. Instrumentation, Tran. Systems and Automation (ISA), Vol. 49, pp. 277--282 (2010).
2. Houpis, H.; Rasmussen, S.; García Sanz, M.: Quantitative Feedback Theory: Fundamentals and Applications. 2da. Ed, CRC Press, Florida (2006).
3. Eberhart, R.; Kennedy, J.: New Optimizer Using Particle Swarm Theory, Sixth Int. Symposium on micro machine and human science, Nagoya, Japan, pp. 39–43 (1995)
4. Hu, X.; Everhart, R.: Multiobjetive Optimization using Dynamic Neighborhood PSO. In: Proc. Evolutionary Comp., (CEC '02). IEEE Press, Honolulu, Vol.2, pp.1677—1681 (2002)
5. García Sanz, M.: Quantitative Robust Control Engineering: Theory and Applications. RTO-En-SCI- Vol.166, pp. 1—44 (2006)
6. Hussein, T.: A Genetic Algorithm for Optimum Design of PID Controller in Load Frequency Control. W. Academy of Science, Eng. and Tech., Vol.70, pp. 956--960 (2012).
7. Khandani, K.; Jalali, A.A.; Alipoor, M.: Particle Swarm Optimization Based Design of Disturbance Rejection PID Controllers for Time Delay Systems. In: Conf. Intelligent Computing and Intelligent Systems (ICIS 2009), IEEE Press, Vol. 1, pp.862--866 (2009).
8. Ämtrom, K.; Hägglund, T.: Control PID Avanzado. Pretice Hall, Madrid (2009)
9. Hollot, C.V.; Misra, V.;Towsley,D.;Gong,W.: Analysis and Design of Controllers for AQM Routers Supporting TCP Flows. IEEE Tr. Aut. Control, Vol.47, No.6, pp. 945--959 (2002)
10. Tian, D.P.:A Review of Convergence Analysis of Particle Swarm Optimization. Int. Jou. Grid and Distributed Computing, Vol.6, No.6, pp.117--128 (2013).
11. Ye, H.; Luo, W.; Li, Z.: Convergence Analysis of Particle Swarm Optimizer and Its Improved Algorithm Based on Velocity Differential Evolution. Comp. Intell. Neuroscience, 2013. http://www.hindawi.com/journals/cin/2013/384125/. Accedido Junio 2015.
12. Gu, D.; Zhang, W.: Design of an H_∞ Based PI Controller for AQM Routers Supporting TCP Flows. In: 48th Conference on Decision and Control and 28th, Chinese Control Conference, IEEE Press, Shanghai, P.R. China, pp. 603--608, (2009).
13. Sun, J.; Zukerman, M.: RaQ: A Robust Active Queue Management Scheme based on rate and queue length. Computer Communications, Vol. 30, pp. 1731-–1741 (2007)
14. Rouhani, M.; Tanhatalab, M.R.; Shokohi-Rostami, A.: Nonlinear Neural Network Congestion Control Based on Genetic Algorithm for TCP/IP Networks. In: 2nd Int. Conf. Computational Intelligence, Comm. Systems and Networks (CICSyN), pp. 1--6 (2010)
15. Voicu, M.: Robust Controller Including a Modified Smith Predictor for AQM Supporting TCP Flow. CEAI, Vol. 14, No. 3, pp. 3-8 (2012).
16. Sundaram, R.K.; Padhy, P.K.: GA-Based PI-PD Controller for TCP Routers. Int. Jou. Machine Learning and Computing, Vol. 3, No. 4, pp. 361--364. (2013)
17. Quet, P-F.; Ozbay, H.: On the Design of AQM Supporting TCP Flows Using Robust Control Theory. In: 42nd Conf. Dec. and Control, IEEE Press, USA, pp. 4220--4224 (2003).
18. Hayes, M.J.; Mahdi Alavi, S.M.; Van de Ven, P.: Robust Active Queue Management using a Quantitative Feedback Theory Based Loop-Shaping Framework. In: 2007 American Control Conference, New York , USA, pp.3077-3082 (2007)

Evaluación de Sistemas Operativos de Tiempo Real sobre Microcontroladores

Medina Santiago[1], Pi Puig Martín[1], Paniego Juan Manuel[1], Dell'Oso Matías[1], Romero Fernando[1], De Giusti Armando[1,2], Tinetti Fernando G.[1,3]

[1]Instituto de Investigación en Informática LIDI (III-LIDI), Facultad de Informática, Universidad Nacional de La Plata, 50 y 120 2do piso, La Plata, Argentina.
[2]CONICET – Consejo Nacional de Investigaciones Científicas y Técnicas
[3]CIC – Comisión de Investigaciones de la Pcia. de Buenos Aires

{smedina, mpipuig, jmpaniego, mdelloso, fromero, degiusti, fernando}@lidi.info.unlp.edu.ar

Abstract. En el presente trabajo se presentan mediciones sobre diferentes sistemas operativos que soportan las características de tiempo real, instalados sobre un sistema de microcontrolador. Dichas evaluaciones caracterizan los posibles tiempos de respuesta, es decir, un límite que determina el tipo de aplicación en la cual pueden ser utilizados, dentro del marco del desarrollo de aplicaciones de Sistemas Embebidos de Tiempo Real. Una de las principales métricas es el tiempo de latencia, el cual representa el tiempo que transcurre desde que se produce una entrada hasta que se emite la salida de respuesta. Luego, no solo se evalúa este tiempo, sino la variabilidad del mismo, ya que en tiempo real es fundamental el determinismo. Si bien no es la única condición que debe cumplir un sistema operativo para ser considerado de tiempo real, es un requisito necesario. Los tiempos de latencia y su variabilidad, determinan las cotas de requerimientos temporales que brinda el sistema.

Keywords: Tiempo Real, Microcontroladores, Sistemas Operativos, Sistemas Embebidos, Latencia.

1 Introducción

Los Sistemas de Tiempo Real (STR) [1] [2] [3] [4] son aquellos que permiten proveer en forma garantizada un servicio dentro de un intervalo de tiempo de respuesta limitado. Esta restricción exige un cuidadoso diseño tanto del hardware como del software.

Los STR están conformados por un conjunto de dispositivos electrónicos acompañados de un Sistema Operativo de Tiempo Real (SOTR). En un enfoque ascendente, estos sistemas pueden estar implementados mediante:

- Sistema electrónico, o sea solo hardware, diseñado especialmente para dar solución a los requerimientos de control.
- Sistemas electrónicos programables. En general, el hardware está diseñado para un conjunto generalizado de requerimientos, y se personaliza a partir del software. Solo se utiliza un programa con instrucciones para lograr la funcionalidad deseada, sobre un hardware que no cuenta con sistema operativo. También son conocidos como entorno de metal desnudo (Bare-metal environment). Son sistemas en los que una máquina virtual se instala directamente en el hardware en lugar de sobre un sistema operativo host [6].
- Sistema electrónico con un Sistema Operativo de Tiempo Real (SOTR) con un programa que logre la funcionalidad deseada. Dentro de estos sistemas los hay de complejidad variada, desde microcontroladores con un SOTR elemental (FreeRTOS, MQX), hasta complejas computadoras corriendo SOTR mayormente derivado de UNIX (QNX) o Linux (RTLinux, RTAI, Linux RT-Preempt).

Las ventajas en los sistemas más complejos, con SOTR, son:

- Tiempo de desarrollo mínimo: debido a que se utiliza el soporte proporcionado por el SOTR.
- Flexibilidad: ya que puede cambiarse fácilmente el software.
- Consecuencia del punto anterior es que estos sistemas son de fácil mantenimiento.
- Cumplir con compromisos temporales estrictos: brinda al programador herramientas para que, una vez ocurrido un evento, la respuesta ocurra dentro de un tiempo acotado. Esto implica que una aplicación mal diseñada puede fallar en la atención de eventos aún cuando se use un RTOS.
- Manejo del tiempo: el RTOS proporciona manejo de temporizadores y esperas.
- Tarea Idle: cuando ninguna de las tareas requiere del procesador, el sistema ejecuta una tarea llamada idle u ociosa, que permite contabilizar el nivel de ocupación del CPU, poner al mismo en modo de bajo consumo o correr cualquier tarea que pudiera ser de utilidad para el sistema cuando no debe atender ninguno de sus eventos.
- Multitarea: simplifica la programación de sistemas con varias tareas.
- Escalabilidad: al tener ejecución concurrente de tareas se pueden agregar más, teniendo la precaución de insertarlas correctamente en el esquema de ejecución del sistema.
- Mayor reutilización de código: si las tareas se diseñan bien, con nula o mínima dependencia, es más fácil incorporarlas a otras aplicaciones.

Entre las desventajas se pueden nombrar:

- Mayor consumo: entre las diferentes aplicaciones de STR se encuentran sistemas móviles como también desconectados de la red eléctrica, que utilizan baterías para el suministro de energía. Su autonomía dependerá de su consumo.
- Mayor probabilidad de fallas: por contener mayor cantidad de componentes, electrónicos o de software, un STR requiere de funcionamiento continuo y sin desperfectos, por lo que muchas veces se prefiere sistemas lo más simples posibles.
- Mayores latencias: todo componente agregado, principalmente de software, implica demoras debido a su ejecución (ejemplo: planificador de tareas).

En este trabajo se aborda la medición de latencia, es decir, el tiempo que se demora entre la ocurrencia de un evento y la ejecución de la primera instrucción en el código del programa que da servicio a la interrupción.

Conforme se asciende en la complejidad del procesamiento, los elementos intervinientes suelen incrementar esta métrica. En este trabajo se analizan los tiempos de latencia de un sistema de desarrollo TWR-K70 que forma parte de la familia Kinetis creada por Freescale. Las diferentes muestras obtenidas sin SOTR se comparan con las evaluaciones sobre el mismo sistema pero utilizando los SOTR FreeRTOS y MQX.

Un Sistema de Tiempo Real puede requerir tiempos de respuesta desde el orden de los segundos, al orden de los milisegundos o microsegundos. La corrección de un sistema de tiempo real, a diferencia de los sistemas convencionales, involucra tanto a las salidas que produce como al tiempo que demora en producirlas. Si no se satisfacen estas restricciones temporales en las respuestas, se arriesga a la falla completa del sistema. De allí la necesidad de conocer las latencias que tengan las plataformas de hardware y SOTR sobre las cuales se piensa desarrollar una determinada solución.

2 Objetivos y metodología

Por tratarse de sistemas embebidos interactuando con el medio físico, los eventos que se produzcan en este medio deben generar respuestas del sistema. Estas respuestas deben ser correctas y emitidas dentro de los límites temporales. La demora en producirlas, también llamada latencia, debe ser conocida. Es objetivo de este trabajo obtener una medida de dicha latencia, tanto utilizando un SOTR como sin él (Bare-metal). La metodología consiste en generar un evento, tomar la medida de tiempo en el que se produjo, y medir nuevamente cuando se elabora la respuesta en la interrupción generada. La diferencia entre estas dos medidas determina el tiempo de latencia. El proceso que se ejecuta como respuesta al evento de una interrupción se lo conoce como Rutina de Servicio de Interrupción (en inglés, Interrupt Service Routine, ISR).

Como pueden seguir produciéndose eventos de interrupción en forma periódica, situación típica en los SOTR, es deseable que dicha ISR se ejecute en el menor tiempo posible. Por tanto, en general, se implementa la respuesta a la interrupción en dos partes:

una dentro de la ISR, lo más pequeña posible, y el resto en una rutina diferida. En este trabajo se analiza la metodología para implementar esta técnica.

2.1 Experimentación

Para llevar a cabo las pruebas, se trabajó con una placa de desarrollo propietaria de Freescale utilizando el IDE Kinetis Design Studio. Se realizaron experimentos sin SOTR y con SOTR, empleándose dos de ellos, uno de código libre y el otro propietario.

El método de medición consistió en tomar el instante de inicio antes de interrumpir al sistema y el tiempo final al captar el evento y atender la interrupción. Para ello, se generó un pulso por un determinado puerto del sistema, el cual se configuró para desencadenar una interrupción al momento de recibir el evento. Esta situación se repitió varias veces para lograr mejores aproximaciones del tiempo final.

De igual manera, los tiempos obtenidos se enviaron por puerta serie, mediante la placa anexa TWR-SER, hacia un host local para realizar el posterior tratamiento de los datos.

Por otra parte, se realizaron diferentes pruebas sobre casos dónde parte de la tarea de elaborar la respuesta se realiza en forma diferida, con lo que el planificador del SOTR interviene de manera directa. Dichas pruebas fueron ejecutadas sobre el SOTR FreeRTOS.

Las tres situaciones contempladas fueron las siguientes:

- Caso A: La tarea principal genera el pulso de salida, el sistema capta el evento generado y llama al manejador de la interrupción, donde se señaliza mediante un semáforo a la tarea principal que, posteriormente, detiene el timer.
- Caso B: La tarea principal genera el pulso de salida, el sistema capta el evento generado y llama al manejador de la interrupción, donde se señaliza mediante un semáforo a una tarea secundaria que, posteriormente, detiene el timer. La prioridad de la tarea principal es igual a la prioridad de la tarea secundaria.
- Caso C: La tarea principal genera el pulso de salida, el sistema capta el evento generado y llama al manejador de la interrupción, donde se señaliza mediante un semáforo a una tarea secundaria que, posteriormente, detiene el timer. La prioridad de la tarea principal es menor a la prioridad de la tarea secundaria.

2.2 Hardware utilizado

Se utilizó un kit de desarrollo TWR-K70 que forma parte de la familia Kinetis creada por Freescale [12]. Está compuesto por un microcontrolador ARM-CortexM4 de 32bits, con una frecuencia de Reloj de 120Mhz. Cuenta con memoria SRAM de 1GB, memoria Flash de 2GB, Acelerómetro, cuatro Led's de uso general, cuatro sensores touch pad, potenciómetro, slot para tarjeta Micro SD, Puerto USB para programación y debugging.

La placa permite la programación de una gran diversidad de aplicaciones ya que tiene un hardware muy variado y permite agregar placas de complemento conformando una estructura de torre. Dentro del hardware adicional, existen módulos de comunicación, módulos de sensado y módulos multimedia. Cabe aclarar que, para el desarrollo del proyecto, se utilizó además el modulo de comunicación TWR-SER, que permite la conexión a una interfaz Ethernet y una Serie.

2.3 Sistemas Operativos Evaluados

Para llevar a cabo las pruebas se emplearon los SOTR FreeRTOS y MQX. Ambos son denominados Sistemas Operativos Embebidos, en los cuales hay una fusión entre el Sistema Operativo y la Aplicación.

2.3.1 FreeRTOS

FreeRTOS [7] [8] [9] [10] [11] es un kernel de sistema operativo de tiempo real para sistemas embebidos que se ha vuelto muy popular, ha sido portado a 35 arquitecturas de microcontroladores y plataformas de hardware, incluso a computadoras personales. Se distribuye bajo licencia GPL con una excepción: permite que el código propietario de los usuarios siga siendo código cerrado, manteniendo el núcleo en sí como código abierto, lo que facilita el uso de FreeRTOS en aplicaciones propietarias.

FreeRTOS está diseñado para ser simple, el kernel está formado por un conjunto de archivos en lenguaje C. A su vez, en algunas arquitecturas, incluye algunas líneas de código ensamblador, mayormente en las rutinas del planificador.

FreeRTOS provee:

- Poco uso de memoria
- Bajo nivel de sobrecarga
- Ejecución rápida
- Métodos para múltiples hilos o tareas
- Mutexes y semáforos
- Timers de software
- Reducción del tick para aplicaciones de bajo consumo.
- Prioridades para las tareas
- Cuatro esquemas de asignación de memoria
- Es de código abierto: está bajo continuo desarrollo, y no hay costo de implementación.
- Amplia comunidad de usuarios: ayuda a resolver problemas que surgen.

El método de múltiples hilos cambia tareas en función de la prioridad y un esquema de planificación round-robin. Luego, el planificador puede ser configurado para operar en

forma apropiativa o colaborativa. Las tareas tienen una prioridad de ejecución, donde 0 es la menor prioridad (se recomienda usar referencias a tskIDLE_PRIORITY +1,+2, etc).

2.3.2 MQX

Es un sistema operativo de tiempo real propietario pero de distribución gratuita. Tiene dos partes:

- PSP: Plataform Support Package, es el subsistema de MQX que está relacionado al core sobre el que va a correr, ej: ARM, ColdFire, PowerPC,etc.
- BSP: Board Support Package, es el subsistema de MQX que contiene el código para soportar una placa determinada de un procesador determinado. Este código tiene los drivers de los periféricos disponibles en una placa dada.

Este sistema operativo provee:

- Comportamiento de tiempo real.
- Tareas.
- Múltiples hilos.
- Drivers para dispositivos.
- Comunicaciones y sus correspondientes pilas de protocolo.
- Kernel multitarea con programación preventiva y respuesta de interrupción rápida.
- Comunicación inter-tarea y sincronización.
- Sistema de archivos.
- Se puede configurar tan sólo 6 KB de ROM, incluyendo núcleo, interrupciones, semáforos, colas y administrador de memoria.
- Incluye pila TCP/IP (RTC).
- Integrado sistema de archivos MS-DOS (MFS).
- Pila de host/dispositivo USB.
- Soporte disponibles para varias plataformas, incluyendo Kinetis, ColdFire, Vybrid, i.MX y Power Architecture.

2.4 Sistema de Pruebas

Para obtener los tiempos de latencia se utilizaron dos herramientas diferentes:

- Reloj del sistema: Se utiliza el reloj del core ARM Cortex M4, denominado Systick. Es un contador decreciente de 24bits que oscila a una frecuencia de entrada de 24MHz. Luego, su configuración se realiza a través de cuatro registros del procesador.
- Componente “TimerUnit” de Processor Expert (PE): En dicho método, se utiliza la componente TimerUnit_LDD propia del sistema procesador experto que provee Freescale. La misma hace uso del módulo FlexTimer de la placa de desarrollo. El FlexTimer es un contador de 16bits, el cual permite configurar el sentido y la

frecuencia de la cuenta. En este caso, se personalizó el módulo para utilizar una frecuencia de 20.9MHz. Su control se realiza a través de funciones de alto nivel de PE.

La ventaja de usar el reloj del core Systick es que se puede unificar dicho método sobre los tres escenarios planteados (BareMetal, FreeRTOS, MQX).

Por otra parte, el método que utiliza el componente de PE, es solo aplicable sobre BareMetal y sobre FreeRTOS, pero no así en MQX. Esto se debe a que Freescale MQX no provee soporte para la creación de proyectos de Processor Expert sobre la placa TWR-K70.

De todas maneras, el tiempo medido utilizando el método de PE corroboró el resultado obtenido en las pruebas con Systick.

3 Resultados

En la Tabla 1 se pueden observar los resultados obtenidos para cada uno de los escenarios.

Tabla 1. Tiempos de latencia

Tiempo Bare-metal	6.9 μs
Tiempo FreeRTOS	6.9 μs
Tiempo MQX	9.0 μs

Como se puede observar, el tiempo de latencia correspondiente a la placa de desarrollo en un entorno Bare-metal es realmente pequeño. Este valor es la referencia principal para las posteriores comparaciones con los SOTR. Además, el tiempo obtenido decreta un límite impuesto por la placa para el desarrollo de aplicaciones que requieran respuestas en intervalos menores al obtenido.

En el caso de FreeRTOS, no se percibe ningún tiempo extra añadido por la utilización del mismo. Esto se debe a que el planificador del SOTR no se ejecuta, ya que en la rutina de interrupción propia del evento no se realiza ningún llamado a alguna API de FreeRTOS. En conclusión, el manejador de la interrupción generada por el pulso, no introduce overhead alguno en el tiempo final de latencia, resultando el mismo idéntico al obtenido en el escenario sin SOTR.

Sin embargo, al evaluar el sistema de microcontrolador sobre el SOTR de Freescale, se advierte un mínimo incremento del tiempo de latencia. Al utilizar las interrupciones estándares que provee MQX, se genera un tiempo extra de aproximadamente 2 μs [5]. Asimismo, al igual que en el caso anterior, el código correspondiente al manejador de la interrupción no realiza llamado alguno a una determinada API del SOTR. En caso de requerirse tiempos de latencia menores al impuesto por MQX, se debe recurrir a interrupciones en modo kernel, las cuales son soportadas completamente por el SOTR de Freescale. Para ello, solo basta con instalar la interrupción al evento requerido con

funciones exclusivas para el kernel que provee MQX. De todas maneras, si ese fuera el caso, la aplicación no dispondría de ciertos beneficios que otorgan las interrupciones clásicas de MQX, como por ejemplo, utilizar los flags y eventos del SOTR como una API para otros procesos MQX.

Como se analizó anteriormente, el manejador de la interrupción de ambos SOTR no ejecuta ningún llamado a una determinada API del mismo. En consecuencia, el tiempo de latencia varía mínimamente o no lo hace, respecto de la utilización de un escenario de placa desnuda.

Luego, se realizaron diferentes pruebas donde se hace un llamado explícito a ciertas funciones del SOTR para, de esta manera, observar el posible overhead introducido por el mismo.

Por tanto, en la Tabla 2 se detallan los resultados obtenidos para el escenario donde la placa TWR-K70 incorpora el SOTR FreeRTOS, con la modificación de que se utiliza una sección diferida en la gestión del evento de entrada.

Tabla 2: Resultados obtenidos utilizando la sección diferida sobre FreeRTOS

Caso	T. mín.	T. max.
A	22 µs	32 µs
B	55 µs	75 µs
C	40 µs	50 µs

Como consecuencia del llamado explícito a una determinada API del SOTR, los tiempos de respuesta se incrementan sustancialmente.

En el primer caso, el manejador del evento desbloquea el semáforo que espera la tarea principal, generando así un overhead propio del llamado a la API de semáforos de FreeRTOS. Como se puede observar, este procesamiento diferido agrega un tiempo extra de entre 15 y 25 µs al tiempo obtenido en la Tabla 1.

Luego, en el caso B, la interrupción desbloquea el semáforo que espera una tarea secundaria, diferente a la tarea principal que generó el pulso de entrada. No obstante, las dos tareas presentan la misma prioridad. En esta situación, el tiempo obtenido se incrementa en mayor medida al igual que su variabilidad. Esto se debe a que el SOTR genera diferentes situaciones al momento de decidir cuál de las dos tareas se ejecuta, luego de terminada la gestión de la interrupción.

Por último, el caso C es idéntico al anterior, con la diferencia de que la tarea secundaria posee una prioridad mayor a la de la tarea principal. Por tanto, el tiempo medido disminuye en comparación a la situación anterior pero se incrementa respecto al caso A. Esto se debe a que, al finalizar la ejecución de la rutina de interrupción, FreeRTOS evalúa si la tarea desbloqueada por el semáforo tiene mayor prioridad que la tarea anteriormente interrumpida. Si esta condición se cumple, se ejecuta el planificador de tareas introduciendo un overhead producido por la utilización de una API del SOTR. Por otra

parte, la variabilidad es idéntica que en el primer escenario, ya que no existen tareas con la misma prioridad.

4 Conclusión

El propósito principal de este trabajo consistió en verificar, mediante el tiempo de latencia, el desempeño de un STR, empleando dos SOTR diferentes y comparando los resultados obtenidos en dichos sistemas con los obtenidos ejecutando aplicaciones directamente sobre el hardware en la modalidad Bare-metal. En las diferentes pruebas se constató que los tiempos de latencia sobre Bare-metal y FreeRTOS son iguales si el procesamiento de la respuesta se realiza dentro de la ISR, donde el SOTR no introduce overhead alguno. En el caso de ejecutarse una rutina diferida con llamadas a API's del SOTR, aparecen tiempos extras con variabilidades mayores, debido a las tareas que lleva a cabo el SOTR, principalmente por el planificador del mismo.

En otro orden, el SOTR propietario MQX, introduce un overhead mínimo en el tiempo de latencia del sistema, siempre y cuándo se haga uso de las, generalmente utilizadas, interrupciones clásicas.

5 Bibliografía

1. N.C. Audsley , A. Burns , M. F. Richardson , A. J. Wellings: Hard Real-Time Scheduling: The Deadline-Monotonic Approach. Proc. IEEE Workshop on Real-Time Operating Systems and Software (1991).
2. L. Buhr. "An Introduction to Real Time Systems". Prentice Hall (1999).
3. A. Burns, A. J. Wellings: Designing hard real-time systems. Proceedings of the 11th Ada-Europe international conference on Ada. ISBN:0-387-55585-4 (1992).
4. A. Burns & A. Wellings: Real-Time Systems and Programming Languages. Addison Wesley, ISBN 90-201-40365-x.
5. L. Prokop, Freescale Semiconductor Application Note. Motor Control Under the Freescale MQX Operating System.
6. A. Silberschatz, Peter Galvin, Greg GAGNE. Operating System Concepts.
7. R. Barry. Using the FreeRTOS Real Time Kernel.
8. C. Becker. RTOS en sistemas embebidos, disponible en http://ieee.eie.fceia.unr.edu.ar/PDF_RTOS.pdf
9. A. Celery. Sistemas Operativos de Tiempo Real, disponible en http://www.sase.com.ar/2011/files/2010/11/SASE2011Introduccion_RTOS.pdf
10. FreeRTOS, disponible en http://en.wikipedia.org/wiki/FreeRTOS
11. FreeRTOS, disponible en http://www.freertos.org/
12. TWR-K70F120M: Kinetis Tower System Module, disponible en http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=TWR-K70F120M

OPTIMIZACIÓN DE LAS SEÑALES ACÚSTICAS UTILIZANDO BEAMFORMING DE BANDA ANCHA PARA PERSONAS CON DISCAPACIDAD AUDITIVA

Paúl Bernal, Fabián Sáenz *, and Amelia Caisapanta

Departamento de Eléctrica y Electrónica DEEE,
Universidad de las Fuerzas Armadas - ESPE
Sangolqui Ecuador
{cpbernal,fgsaenzavcaisapanta}@espe.edu.ec
http://www.espe.edu.ec

Abstract. La voz es la principal forma de comunicación que posee el ser humano, el presente trabajo busca ayudar a las personas con problemas de audición, mediante el realzado de la voz. Para esto se utilizan las técnicas de *beamforming* y *Direction of Arrival* (DOA) en un arreglo de micrófonos semiesférico. Mediante el análisis de los algoritmos adaptativos *Linearly Constrained Minimum Variance* (LCMV) con sus tres familias: *Constrained, Generalized Sidelobe Canceler* (GSC) y Householder, utilizando los algoritmos *unconstrained*: *Least Mean Squares* (LMS), *Normalized Least Mean Squares* (NLMS), *Recursive Least-Squares (RLS)* y *Conjugate Gradient* (CG). Se pretende verificar las cualidades y desventajas de cada uno de los algoritmos tratando de optimizar sus desempeños variando sus respectivos parámetros de adaptación, a fin de obtener una rápida convergencia sin comprometer el *Mean Square Error* (MSE) ni aumentar el costo computacional. Los algoritmos adaptativos se comparan y en base a los resultados obtenidos se selecciona el algoritmo más idóneo, tomando en cuenta la velocidad de convergencia, costo computacional, todos los resultados son analizados a fin de obtener las conclusiones y recomendaciones.

Keywords: Mejoramiento de la voz, Beamforming, Direction of Arrival, Algoritmos Adaptativos, LCMV

1 Introducción

El procesamiento digital de señales de voz posee una amplia base teórica y experimental que se encuentra bien adaptada a las demandas computacionales del procesamiento digital de señales de voz [1]. El estudio de las señales de voz es un tema de importancia, dado que una gran parte de la sociedad presenta

* Estudiante del Doctorado en Ciencias Informáticas de la Universidad Nacional de La Plata, Argentina.

algún tipo de discapacidad auditiva, por tanto la presente trabajo pretende ser la base matemática para la implementación de soluciones que mejoren la calidad de vida de dichas personas.

Existen varias técnicas para el procesamiento de señales sonoras, dependiendo de la aplicación que se realice, en este caso se utiliza la metodología de *beamforming* [2] y la estimación de la dirección de arribo de las señales (*Direction of Arrival*) DOA [3], que tradicionalmente eran estudiados en la teoría de antenas, substituyendo las ondas electromagnéticas por señales acústicas y el arreglo de antenas por un arreglo de micrófonos, presentando varias aplicaciones en diferentes campos. Los algoritmos son los encargados de realizar tales tareas, dentro de los cuales se puede nombrar: *Delay and Sum DAS* , Capon, *Multiple Signal Classification* MUSIC, Root-MUSIC, *Estimation of Signal Parameters via Rotational Invarence Technique* ESPRIT [4] y *Linearly Constrained Minimum Variance* LCMV [2].

De los algoritmos se pretende escoger uno que tenga buenos resultados y que su costo computacional sea razonable, para que después se implementen en un FPGA (*Field-Programmable Gate Array*) . El presente articulo se encuentra compuesto de cinco secciones: En la Sección II se expone los conceptos básicos que se usan para la elaboración de este proyecto. En la Sección III, analiza el arreglo geométrico semiesférico. En la Sección IV se realiza un análisis de los resultados mas relevantes encontrados y en la Sección V se concluye el trabajo y se presentan trabajos futuros.

2 Conceptos Básicos

Para el desarrollo del presente articulo se inicio implementando un arreglo lineal ULA con un $M = 4$, esto pensado en futuras implementaciones con recursos de harward y de economías limitadas. Como se indico en la Sección I, existe un sinnúmero de algoritmos que permiten solventar el problema como escoger el mas adecuada.

Para determinar el algoritmos a implementar se escogió como limitación el costo computacional, proyectado lo para una posterior implementación, utilizando el número de multiplicaciones, como medida, la cual se detalla en la Tabla 1 a continuación:

Table 1. Costo computacional de los algoritmos

Algoritmo	Número de Multiplicaiones
Delay and Sum	$\frac{180^\circ}{\Delta^\circ}(M^4+2M^3+4M^2+4M+2)$
Capon	$\frac{180^\circ}{\Delta^\circ}(M^4+3M^3+4M^2+4M+2)$
MUSIC	$\frac{180^\circ}{\Delta^\circ}(M^4+2M^3+7M^2+4M+2)$

Donde M es el número de micrófonos y Δ° es la escala para el barrido del ángulo. Estos costos computacionales es un aproximado y está basado en cada uno de los códigos que se utilizan para estimar el DOA [4]. Como se observa el costo computacional es alto esto es debido a que existe inversión de matrices en los procesos. Los algoritmos LCMV no son presentados en la Tabla 1 puesto que tienen un gran cantidad de estructuras, que tienen costos computacionales variables; por lo cual son los más adecuados para su implementación, con resultados aceptables.

2.1 Algoritmos de Filtraje Adaptativo

Los algoritmos adaptativos intentan minimizar una función de costo $\xi[k]$ y que generalmente se relaciona con el error de salida [5]. En la Figura 1 se muestra la estructura de un *beamforming* adaptativo *broadband* con M sensores y N taps (número de coeficientes del filtro). La salida de la estructura de la Figura 1 se expresa como $y[k] = \mathbf{w}^H(k)\mathbf{x}(k)$:

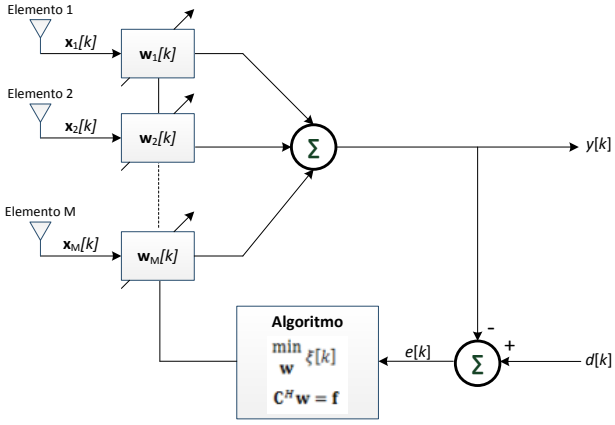


Fig. 1. Estructura de *Beamforming* Adaptativo *Broadband*.

$$\min_{\mathbf{w}} \xi(k) \quad \text{sujeto a } \mathbf{C} = \mathbf{wf} \tag{1}$$

donde $\xi[k] = E[|e[k]|^2]$. Con la matriz de restricción $\mathbf{C}$ de dimensión $MN \times p$, siendo p el número de restricciones y $\mathbf{f}$ el vector de ganancia de dimensión $p \times 1$ y corresponde a la ganancia unitaria en la fuente deseada y ganancia nula para las direcciones indeseadas, si fuera la aplicación a utilizar. Utilizando la función de costo y relacionando la señal de error $e[k] = d[k] - y[k]$, se obtiene la solución

de Wiener sujeto a $\mathbf{C}^H\mathbf{w} = \mathbf{f}$.

$$\mathbf{w}_{opt} = \mathbf{R}^{-1}\mathbf{p} + \mathbf{R}^{-1}\mathbf{C}(\mathbf{C}^H\mathbf{R}^{-1}\mathbf{C})^{-1}(\mathbf{f} - \mathbf{C}^H\mathbf{R}^{-1}\mathbf{p}) \tag{2}$$

Donde $\mathbf{R} = E[\mathbf{x}[k]\mathbf{x}^H[k]]$ y $\mathbf{p} = E[d^*[k]\mathbf{x}[k]]$ siendo $\mathbf{R}$ es la matriz de autocorrelación de la señal de entrada y $\mathbf{p}$ es el vector de correlación cruzada entre la señal deseada y la señal de entrada. Es fácil verificar que en ausencia de la señal deseada $d[k] = 0$ el vector $\mathbf{p} = \mathbf{0}$ y la solución óptima se reduce a la siguiente expresión:

$$\mathbf{w}_{opt} = \mathbf{R}^{-1}\mathbf{C}(\mathbf{C}^H\mathbf{R}^{-1}\mathbf{C})^{-1}\mathbf{f} \tag{3}$$

La solución de la Ecuación (3) es conocida como la ecuación óptima LCMV, donde la función de costo se denomina *Minimum Output Energy* (MOE). Los algoritmos adaptativos LCMV como ya se indicó requiere de un conjunto de restricciones que deben ser cumplidas en cada interacción. Mediante la imposición de restricciones lineales sobre el filtro adaptativo, la necesidad de una señal deseada a menudo puede ser descartada. Este tipo de algoritmos adaptativos se dividen en tres familias: con Restricciones Lineales (*Linearly Constrained*), de estructura GSC (*Generalized Sidelobe Canceler*) y los de estructura Householder [6].

2.2 Algoritmos Linearly Constrained

La actualización de los coeficientes en los algoritmos adaptativos *Constrained* se realiza en un subespacio que es ortogonal al subespacio generado por la matriz de restricción. La dirección de la actualización está dada por el vector de la señal de entrada premultiplicado por una matriz de proyección [7]. La matriz de restricción $\mathbf{C}$ como se indicó es de dimensión $MN \times p$. Esta matriz tiene la forma siguiente:

$$\mathbf{C} = \begin{bmatrix} 1 \cdots 1\ 0 \cdots 0\ 0 \cdots 0 \\ 0 \cdots 0\ 1 \cdots 1\ 0 \cdots 0 \\ 0 \cdots 0\ 0 \cdots 0\ 1 \cdots 1 \end{bmatrix}^T \tag{4}$$

Al utilizar la matriz $\mathbf{C}$ de la Ecuación (4) los algoritmos que se estudian en la presenta investigación poseen un desempeño bajo, puesto que el valor del *Mean Square Error* MSE es alto, como se puede observar en la Figura 2, en la cual el valor promedio del MSE de cada uno de los algoritmos es de aproximadamente -7dB, por lo que se presume que al converger los algoritmos se encuentran en un mínimo local, puesto que a pesar de que se varía el tamaño del parámetro de adaptación de los algoritmos no se obtiene una disminución en el valor del MSE y un aumento exagerado en los valores de los parámetros de adaptación puede hacer que el algoritmo diverja. Para que los algoritmos mejoren su desempeño se precondiciona la estructura de la matriz $\mathbf{C}$, transformando esta matriz en una matriz toeplitz [8] y como se puede observar en la Figura 3 los algoritmos poseen un valor menor en el MSE en comparación a la Figura 2, con un valor promedio del MSE de $-28dB$ para CLMS, $-30dB$ para NLMS, $-33dB$ para CRLS y $-35dB$ para el algoritmo CCG.

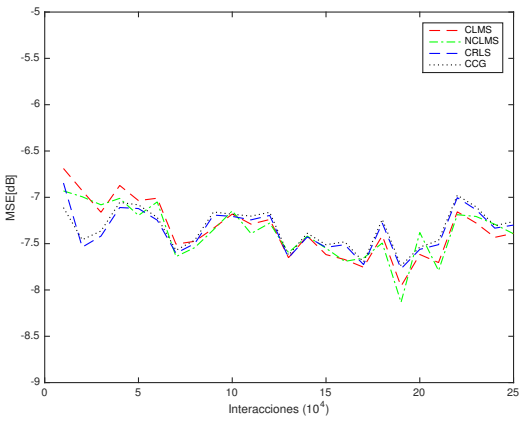


Fig. 2. Algoritmos Constrained para un arreglo ULA (**C** original).

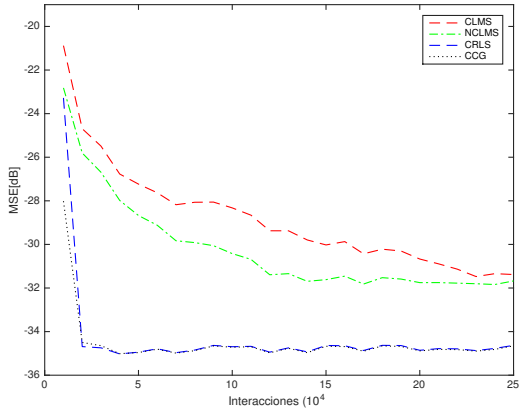


Fig. 3. Algoritmos Constrained para un arreglo ULA (**C** modificada).

2.3 Estructura Generalized Sidelobe Canceler (GSC)

En la Figura 4 se presenta la estructura GSC. La estructura GSC es una alternativa diferente para implementar filtros adaptativos *linearly constrained*. En la estructura GSC, la dimensión del subespacio de adaptación se reduce adecuadamente a través de una matriz de *blocking* **B**, permitiendo que cualquier algoritmo unconstrained pueda ser empleado como *constrained* [9].

Sea **B** la matriz *blocking* de dimensión $MN \times (MN - p)$ que abarca el espacio nulo de la matriz de restricción **C**. Existe una limitación para la matriz *blocking* y es que sus columnas deben ser ortogonales a la matriz de restricciones [6]. Una matriz de transformación **T** de longitud $MN \times MN$, relaciona la estructura GSC con el filtro LCMV.

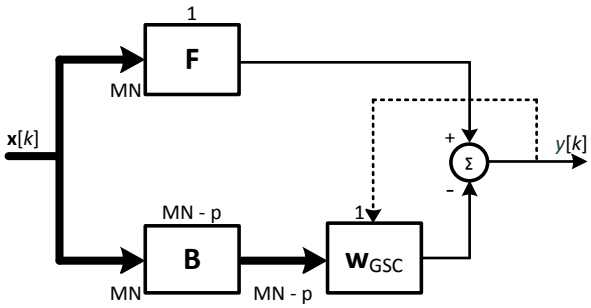


Fig. 4. Estructura Generalized Sidelobe Canceling (GSC).

$$\mathbf{T} = [\mathbf{C} \vdots \mathbf{B}] \tag{5}$$

La Figura 5 muestra el MSE de los algoritmo GSC, de los que se observa los algorimtos GSC-RLS y GSC-CG tienen el mejor desempeño

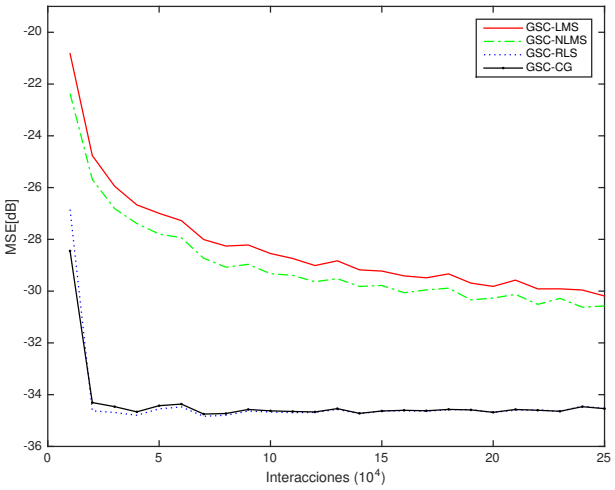


Fig. 5. MSE de los algoritmos GSC.

2.4 Transformada de Householder

La multiplicación de la matriz de *blocking* por el vector de señal de entrada en la estructura GSC puede ser computacionalmente alta y para muchas aplicaciones, no es práctico. La estructura Householder es una solución a este problema

mediante la matriz $\mathbf{Q}$ y sus reflectores de Householder que cumple como la estructura GSC, pero computacionalmente son más eficientes [9].

La obtención del algoritmo HCLMS se inicia con la rotación en el vector $\mathbf{Px}[k]$ con el fin de asegurar que el vector de coeficientes no se perturbe en otra dirección no excitado por $\mathbf{Px}[k]$. Esto es posible si una matriz de rotación ortogonal $\mathbf{Q}$ se utiliza como la transformación que va a generar un vector de coeficientes modificados $\hat{\mathbf{w}}[k]$ que se relaciona a $\mathbf{w}[k]$ [9].

$$\hat{\mathbf{w}}[k] = \mathbf{Q}\mathbf{w}[k] \tag{6}$$

La matriz $\mathbf{Q}$ se debe escoger de tal manera que $\mathbf{Q}\mathbf{Q}^H = \mathbf{I}$. Los resultados de las simulaciones se observan en la Figura 6, destacando se nuevamente los algoritmos HCRLS y HCCG.

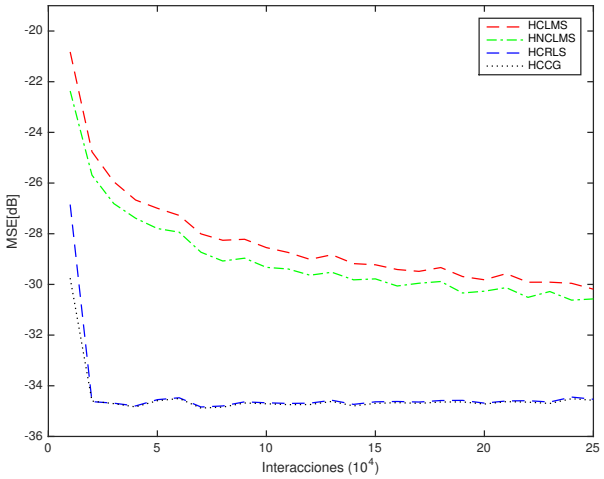


Fig. 6. MSE de los algoritmos Householder.

3 Análisis de los Algoritmos en la Geometría Semiesférica

En la literatura existen tres opciones para la configuración de un arreglo esférico [10]. La geometría que se utiliza en la presente investigación se basa en la configuración de L anillos con igual espaciamiento, donde los M_L micrófonos tienen una distancia de separación uniforme. El arreglo de micrófonos para la presente investigación consta de un semianillo $l = 1$ con 4 micrófonos espaciados uniformemente con una apertura entre cada micrófono adyacente de 60°, donde r es el radio con dimensión de 15 cm, posicionado en un plano (r_x, r_y) como se muestra en la Figura 7

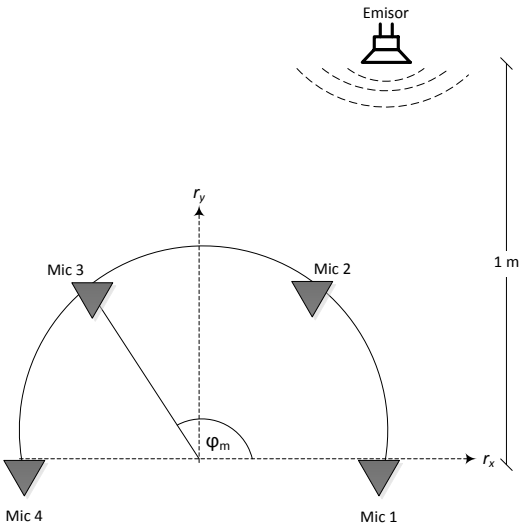


Fig. 7. Arreglo Semiesférico.

Las coordenadas esféricas de cada micrófono está dado por (r, θ, ϕ_m). Donde $m = 0, \cdots, 3$. Se utiliza cuatro micrófonos con $\theta = \pi/2$ y $\phi_m = m\pi/M$. Se supone un medio ideal, la velocidad de propagación de las ondas sonoras es constante y el tiempo de propagación depende solamente de la distancia recorrida. Para la presente investigación, el medio de interés es el aire cuyo valor es de $340m/s$ (no se considera las alteraciones debido a la altura en relación al nivel del mar). Según la Figura 7 la distancia d_m establece un camino directo entre la fuente emisora y el eje r_x del arreglo de los micrófonos. La distancia $d_m = 1m$.

$$t_m = \frac{d_m}{v} \tag{7}$$

El conocimiento del tiempo de propagación del emisor a los micrófonos constituye la base para el cálculo de los diferentes tiempos de llegada ya que la señal llega a los micrófonos en distintos tiempos, por tanto el tiempo de propagación desde la fuente emisora a los micrófonos Mic 1 y Mic 4 será el siguiente: $t_1 = t_4 = 1m/(340m/s) = 2,94ms$ El tiempo de propagación desde la fuente emisora a los micrófonos Mic 2 y Mic 3 es: $t_2 = t_3 = 0,87m/(340m/s) = 2,56ms$. Entonces, la señal llega al micrófono Mic 1 con un retardo de tiempo de 0,38 ms con relación al micrófono Mic 2 y de manera similar sucede con la señal que arriba al micrófono Mic 4 que lo hace con un retardo de 0,38 ms con relación al micrófono Mic 3. Las señales recibidas por los micrófonos son muestreadas a una tasa $f = 22050Hz$.

4 Análisis de los Resultados

Para realizar el analisis de los resultados partimos del costo computacional, en relación al numero de multiplicaciones como podemos observar Tabla 2, la estructura que menos multiplicaciones emplea es la estructura de Householder, como ya se menciono los reflectores householder mejorar el costo computacional.

Table 2. Costo computacional de los algoritmos LCMV

Alg.	Constrained	GSC	Householder
LMS	$N^2 + (2p+2)N + 1$	$N^2 + (3-p)N - (2p-1)$	$(2p+2)N - (p^2-1)$
NLMS	$N^2 + (3p+3)N + 1$	$N^2 + (4-p)N - (3p-1)$	$(2p+3)N - (p^2+p-1)$
RLS	$6N^2 + (19p+4)N + p$	$5N^2 + (6-9p)N + 4p^2 - 5p$	$4N^2 + (5-6p)N + 2p^2 - 3p + 2$
CG	$4N^2 + (4p+10)N + 1$	$4N^2 + (11-7p)N + 3p^2 - 10p + 1$	$3N^2 + (10-4p)N + p^2 - 8p + 2$

Otra manera de verificar los resultados es mediante la gráfica *beampattern* (patrón de directividad) de la Figura 8, de cada uno de los algoritmos Householder, según dicha figura los algoritmos presentan una buena resolución en sus *beampatterns* y el ángulo obtenido coincide con el de la señal de entrada de 90°.

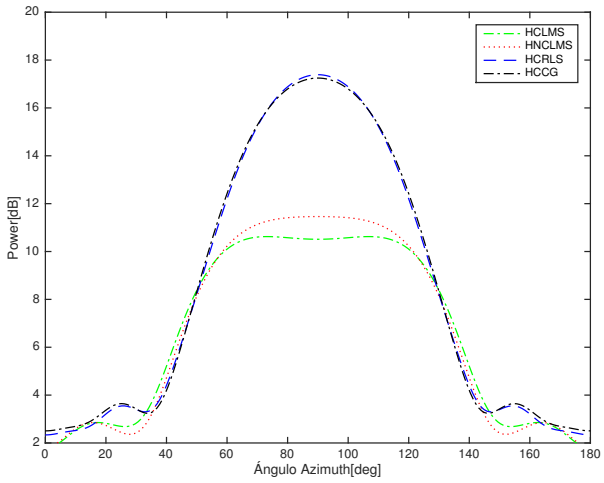


Fig. 8. Algoritmos Householder respecto al *beampattern*

Los algoritmos HCRLS y HCCG poseen un similar comportamiento, por tanto una mayor ganancia, de aproximadamente 6 dB más que los algoritmos HCLMS y HNCLMS.

5 Conclusiones

En el presente trabajo corrobora, que las estructuras LCMV, son las de menor costo computacional sobre las técnicas tradicionales de *beamforming* y *Direction of Arrival* (DOA) empleadas; a cambio provee una solución razonable con un costo computacional bajo y un MSE pequeño. En cuanto a la familia *constrained*, se necesita condicionar mejor la matriz $\mathbf{C}$, lo que conlleva un aumento en el costo computacional, por lo cual la estructura idónea mas es la estructura Householder, ya que sus reflectores son optimizados versus la estructura GSC y su costo computacional es reducido como se indica en Tabla 2. De esta estructura el algoritmo HCCG, es el que presenta mayores ventajas respecto a los otros, como se comprobo en el análisis de los resultados.

En trabajos a futuro, se pretende encontrar la forma de condicionar la matriz $\mathbf{C}$, de modo a obtener mejores resultados, con un MSE menor, sin comprometer el costo computacional, y con la idea de una implementación en un FPGA.

References

1. Rabiner L., & R. W., S. Theory and Application of Digital Speech Processing, 2009.
2. Van Veen, B., & Buckley, K. Beamforming Techniques for Spatial Filtering. CRC-netBASE, 2000 .
3. Monzingo R., Haupt R., & Miller, T. Introduction to Adaptive Arrays (Second ed.). Published by SciTech Publishing, Inc. ,2011.
4. W. Y., Cho, Y. S., & Choo, C. Y. (2012). MATLAB /Simulink for Digital Signal Processing. Seoul, Korea: Hongrung Publishing Company.
5. Paulo S. R. Diniz, Adaptative Filtering Algorithms and Practical Implementation, Third Edition, Springer, 2008
6. S. Chandran, Adaptive Antenna Arrays Trends and Applications. Springer-Verlag Berlin Heidelberg, 2004.
7. M. de Campos, S. Werner, & J. Apolinário. Constrained Adaptation Algorithms Employing Householder Transformation. IEEE Transactions on Signal Processing Vol. 50, No. 9, 9. September 2002
8. K. Gundersen, & J. Hakon Husoy, . Preconditioner structures for the CLMS adaptive filtering algorithm. IEEE. 2006
9. S. Werner, J. Apolinário, & Marcello L. R. , d. (s.d.). On the Equivalence of the Constrained RLS and the GSC-RLS Beamformers. Helsinki University of Technology, Instituto Militar de Engenharia, and Universidade Federal do Rio de Janeiro.
10. Z. Huang & C. Balanis, Adaptive Beamforming Using Spherical Array. IEEE, 2005.

Sistema de adquisición de datos para medir desorción ó absorción de hidrógeno activadas térmicamente

Jorge Runco, Marcos Meyer
IFLP - Depto.de Física – Facultad de Cs.Exactas– UNLP
CONICET
{ runco, meyer }@fisica.unlp.edu.ar

Resumen
Este trabajo se enmarca dentro de las actividades llevadas en conjunto entre el Laboratorio de Electrónica y el grupo de Física Experimental cuya línea de investigación tiene como objetivo el estudio de materiales nanoestructurados aptos para el almacenamiento de hidrógeno.
Además de las técnicas habituales de caracterización (difracción de rayos X, microscopía electrónica, calorimetría diferencial y volumetría [1], [7], en muestras que contengan Fe, es posible utilizar Espectroscopía Mössbauer [2], sensible al entorno atómico de los átomos de Fe. En este sentido, sería muy importante poder observar la evolución de la señal Mössbauer (en la modalidad de velocidad constante) en simultáneo con la detección del flujo de hidrógeno asociado a la desorción o absorción activadas térmicamente.
Con esta finalidad se ha construido un horno basado en un calefactor [3] anular, con cámara de acero refrigerada por agua, acoplado a un fluxímetro. La experiencia consiste en calentar la muestra de modo controlado, registrando el tiempo, la temperatura y el flujo de hidrógeno mientras de manera independiente y simultánea se mide el espectro Mössbauer.
En este trabajo se muestra el desarrollo e implementación de un sistema que controla y automatiza la adquisición de los parámetros antes mencionados.
El experimento es controlado a través de un sistema de cómputo (Notebook ó PC), midiendo, controlando, registrando y graficando en forma automática la evolución de las variables del experimento (temperatura y flujo de gas), en función del tiempo. El software se desarrolló en un lenguaje de programación de alto nivel (Matlab, Delphi) ofreciendo al usuario una interfaz gráfica típica de los lenguajes visuales.
.

Palabras claves: Espectroscopía Mössbauer, procesos, absorción y desorción, adquisición de datos.

1. Introducción

Uno de los desafíos más importantes para el desarrollo y la utilización del hidrógeno como vector energético es la posibilidad de almacenarlo de manera segura y eficiente.

Actualmente existen varias formas de hacerlo, cada una de ellas posee ventajas y desventajas.El almacenamiento de hidrógeno en estado gaseoso es el más simple, pero es muy voluminoso y necesita de altas presiones en los depósitos de confinamiento.

El almacenamiento en estado líquido necesita de depósitos a muy bajas temperaturas, debiendo emplearse grandes cantidades de energía para mantener temperaturas criogénicas.

Debido a que el hidrógeno es altamente reactivo hay una gran cantidad de elementos susceptibles de reaccionar con él para la formación de hidruros. Si las condiciones de presión y temperatura son adecuadas, podrían utilizarse hidruros como una alternativa más eficiente respecto a los modos anteriormente mencionados.

Entre otras ventajas, la absorción en metales [1], formando una fase hidruro respecto de los sistemas actuales (compresión y la licuefacción), no requiere trabajo para comprimir o licuar ni tampoco temperaturas criogénicas.

Una cuestión importante en la investigación experimental es el análisis de las propiedades de absorción-desorción de hidrógeno de nuevos materiales, como así también el estudio de las cinéticas de absorción y desorción.

2. Descripción del sistema de medida

Continuando con el desarrollo de equipamiento que permita realizar nuevas experiencias para estudiar las cinéticas de absorción-desorción de hidrógeno a distintas temperaturas, en este trabajo se muestra el horno construido, su control y el sistema que automatiza la experiencia.

Permite estudiar las cinéticas de absorción-desorción de hidrógeno, a distintas temperaturas, manteniendo constante la presión en la cámara de reacción, en un amplio rango de temperaturas (300K hasta 1000K) y presiones (1 mbar hasta 50 bar).

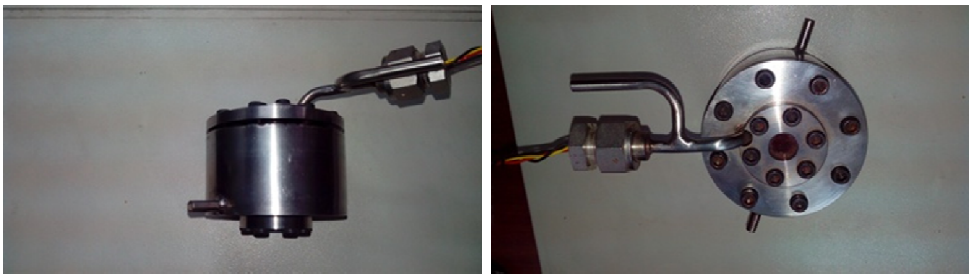


Figura 1. Horno utilizado

La imagen muestra el horno donde se encuentra la muestra a ensayar. El mismo está refrigerado con agua y se estudió su respuesta a fin de poder modelizarlo para realizar el control del mismo.

Se aplicó un escalón de potencia y se midió la respuesta del mismo (evolución de la temperatura) y en base a esto se determinó la estrategia de control más adecuada. Como resultado se utilizó un modelo con distintas técnicas de control (proporcional, proporcional + integral) de acuerdo a las especificaciones de la experiencia en cuanto a si se realizaba la misma llevando a una temperatura determinada a la muestra y luego debía mantenerse en ese valor con un determinado error, ó si la misma se realizaba siguiendo un crecimiento lineal de la temperatura (rampa). Todo esto fue tenido en cuenta a la hora de seleccionar el algoritmo de control.

La figura 2 muestra el diagrama esquemático de la experiencia donde se ve la interconexión de los distintos módulos y componentes.

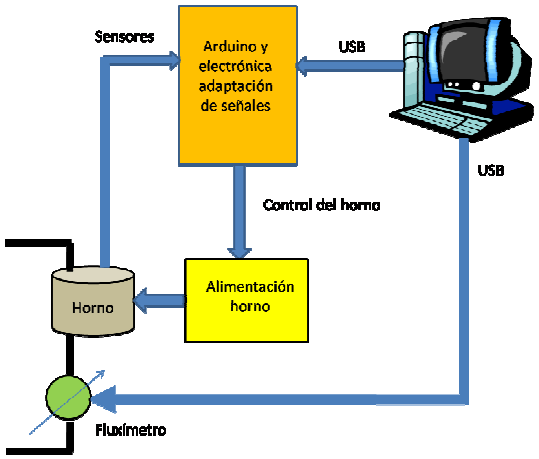


Figura 2

El instrumento de medida se desarrolló sobre sistema de cómputo, notebook ó tipo PC, que por medio de puertos usb controla a un microcontrolador arduino [4] y al medidor de flujo (fluxímetro).

El conversor A/D que posee la plaqueta arduino y electrónica adicionada para acondicionar las señales, son los encargados de la medida de temperatura y presión [5]. Salidas digitales del arduino se utilizan para controlar la etapa de potencia que alimenta al horno. Mientras que el medidor de flujo es controlado directamente desde un puerto USB, pues el instrumento entrega la información en forma digital (Protocolo MODBUS [6]).

Durante el desarrollo de la experiencia los parámetros antes mencionados, temperatura, presión y flujo de gas, son adquiridos y almacenados. El sistema cuenta con la posibilidad

de realizar la experiencia a temperatura constante, temperatura creciendo linealmente (rampa de temperatura).

Esta experiencia fue diseñada para poder medir simultáneamente, con equipo adicional, el efecto Mosbauer y la variación de esta señal con la desorción y absorción de hidrógeno dependiente de la temperatura. La figura 3 muestra la disposición del equipo adicional que no fue mostrado en la figura 2.

Con el equipamiento desarrollado, que permite controlar el flujo gaseoso y la temperatura, en las medidas realizadas por el detector se reflejarán los efectos que producirán las variaciones de dichos parámetros.

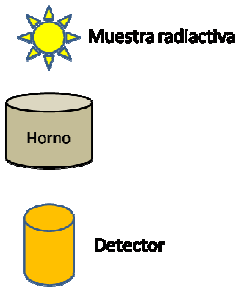


Figura 3

3. Conclusiones

Este equipo fue desarrollado a pedido de un grupo de Investigación del IFLP (Instituto de Física de La Plata) – Depto. de Física (Fac. Ciencias Exactas – UNLP) que investigan en el tema. Permite seguir la evolución de la señal Mosbauer (equipo adicional) en función de la desorción y absorción de hidrógeno activadas térmicamente (equipo mostrado en este trabajo).

Se desarrolló e implementó un equipo de "bajo costo" realizado con componentes comerciales que permite estudiar la absorción y desorción de hidrógeno en metales. El desarrollo fue realizado "a medida" para cumplir los requerimientos de las experiencias en cuanto a los distintos parámetros a medir. Se desarrolló e implementó el módulo de adaptación de señales y el software que controla y automatiza la experiencia como así también la cámara que contiene a la muestra y al calefactor del horno. También la etapa de potencia que alimenta al horno.

4. Referencias

[1] Optimización de un hidruro complejo para almacenamiento de hidrógeno. Tesis doctoral Junio 2009 – Cardozo, César Luis - Centro Atómico Bariloche.

[2] Blatt, Frank J, Modern Physics, McGraw Hill, 1992. Efecto Mossbauer, Ch 15.

[3] Watlow ULTRAMIC Advanced Ceramic Heaters. https://www.watlow.com

[4] http://dfists.ua.es/~jpomares/arduino/page_03.htm

[5] Data Acquisition and Control Handbook. Keithley

[6] MODBUS Protocol Specification - http://www.modbus.org

[7] Sievert-type measurement and acquisition system for the study of hydrogen storage in solids. Runco J. Meyer M. CACIC 2013 – Mar del Plata - Argentina

Evaluación de Parámetros Adecuados para una Red Inalámbrica de Sensores

Guillermo Reggiani, Guillermo Friedrich y Sergio Pellegrino

Grupo SITIC – Universidad Tecnológica Nacional – FRBB
{gfried, ghreggiani, spellegrino}@frbb.utn.edu.ar

Resumen. En este trabajo se presenta un análisis de funcionamiento de una red de sensores inalámbricos (WSN) basada en el estándar IEEE 802.15.4, a fin de evaluar el impacto de tener que atravesar nodos enrutadores en el camino hacia el nodo final o receptor. Esto continúa trabajos previos en los que se evaluó el desempeño de este tipo de redes en escenarios de aplicación típicos. Las WSN vienen teniendo un importante crecimiento en los últimos años, en aplicaciones de domótica, automatización industrial, control y monitoreo del medio ambiente, robótica, telemedicina, etc., motivando esto a realizar diversos estudios tendientes a obtener el mejor rendimiento de la red. Para los ensayos se utilizaron módulos CC2538EM de Texas Instruments. Los parámetros evaluados fueron el tiempo de transmisión, el tamaño del paquete y la tasa de envío, a fin de determinar las condiciones más adecuadas de operación.

Palabras Claves: LR -WPAN; 802.15.4; CC2538EM; tiempo de transmisión.

1 Introducción

Las redes de sensores inalámbricas (WSN: Wireless Sensors Networks) han tenido un rol preponderante en estos últimos años. Hoy en día tienen aplicación en áreas tales como domótica, automatización industrial, control y monitoreo del medio ambiente, robótica, telemedicina, entre otras, y por tal motivo han conducido a realizar diversos estudios tendientes a obtener su mejor rendimiento. El estándar IEEE 802.15.4 [1] define las especificaciones de capa física (PHY) y la subcapa de control de acceso al medio (MAC) para redes inalámbricas de área personal con baja tasa de transferencia (LR-WPAN). El comité IEEE 802.15.4 y la ZigBee Alliance [2] trabajaron en conjunto y desarrollaron la tecnología conocida comercialmente como ZigBee. ZigBee se apoya en las capas MAC y PHY de 802.15.4 para implementar las capas de enrutamiento y aplicación, correspondientes a las capas 3 y 7, respectivamente, del modelo OSI.

Las principales características de un LR-WPAN son la facilidad de instalación, transferencia de datos confiable, operación de corto alcance, bajo costo y una razonable duración de la batería, con un protocolo simple y flexible.

El presente trabajo es una continuación de [3] y tiene como objetivo ensayar una red de sensores en donde ciertos nodos actúan como enrutadores, retransmitiendo paquetes hacia el receptor final. Se evaluaron parámetros tales como el tiempo de

transmisión, el tamaño de paquetes y la tasa de envío. Se efectuaron los ensayos en escenarios similares a los que, típicamente, se pueden encontrar en las aplicaciones mencionadas al comienzo de esta sección.

El resto del trabajo está organizado de la siguiente manera: en la segunda sección se describen aspectos generales del estándar 802.15.4, en la tercera sección se mencionan trabajos relacionados, en la cuarta sección se muestran los ensayos realizados y en la quinta sección las conclusiones y lineamientos para futuros trabajos.

2 Estándar 802.15.4 (LR-WPAN)

El estándar IEEE 802.15.4 [1] define las especificaciones para las capas PHY y MAC para redes de tipo LR-WPAN. Una LR-WPAN es una red de comunicaciones simple y de bajo costo que brinda conectividad inalámbrica con potencia limitada, para aplicaciones que tienen bajos requerimientos de performance.

El estándar define distintas formas de operación de capa física en una variedad de bandas de frecuencia.

Se especifican diferentes tipos de dispositivos que pueden participar en una red 802.15.4: dispositivo de función completa (FFD) y dispositivo de función reducida (RFD). El FFD tiene tres modos de operación: Coordinador PAN, Coordinador y dispositivo. Un FFD puede dialogar con otros FFD y/o RFD. Un RFD solo puede hacerlo con un FFD.

Un RFD está orientado a aplicaciones extremadamente simples, como un interruptor de luz o un sensor infrarrojo pasivo, que requieren transmitir pequeñas cantidades de datos, y puede asociarse con un único FFD a la vez. En consecuencia, un RFD se puede implementar usando un mínimo de recursos y memoria.

Dependiendo de los requerimientos de la aplicación, el estándar tiene previstas dos topologías para su operación: punto a punto y estrella.

En la Fig. 1 se muestra la estructura de trama de 802.15.4.

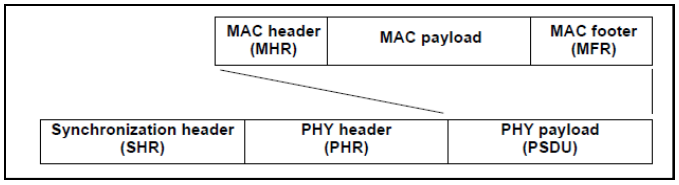


Fig. 1. Estructura de trama de 802.15.4

El estándar define cuatro tipos de estructuras de trama:

- Trama Beacon
- Trama de datos
- Trama de reconocimiento
- Trama de comandos MAC

IEEE 802.15.4 emplea dos tipos de mecanismos de control de acceso al medio: CSMA/CA y ALOHA. El primero consiste en esperar hasta que el canal esté libre para transmitir, según un procedimiento orientado a intentar evitar la ocurrencia de colisiones; mientras tanto, el segundo transmite sin verificar si el medio ya está ocupado por una transmisión iniciada previamente. Este segundo mecanismo, más simple, es factible de ser utilizado en redes compuestas por pocos nodos y/o con bajas tasas de envío, donde las probabilidades de colisionar sean muy reducidas.

3 Trabajos Relacionados

En [4] se presenta un sistema de monitoreo basado en una WSN ZigBee, destinado a la medición de la concentración de compuestos orgánicos volátiles (VOCs). Consiste en un gateway, una estación base, un sensor y un módulo ZigBee. El software se encarga de la adquisición de datos, procesamiento y transmisión, y las estrategias para la conservación de energía del sistema completo. La red consiste de dispositivos finales equipados con sensores específicos, routers que propagan la red a grandes distancias y un coordinador que se comunica con el server. El diseño está basado en un microcontrolador ATmega16 y un módulo ZigBee Atmel RF230, que procesan y comunican los datos con bajo consumo de energía. Los resultados mostraron que esta WSN es apta para monitorear concentraciones de VOCs con alto nivel de precisión y puede ser usada para implementar el monitoreo en forma automática.

En [5] se presenta la introducción a un sistema para monitoreo del entorno del agua basado en ZigBee. Se describe la arquitectura del hardware y el software de la WSN. La aplicación de una WSN para el dominio del agua tiene gran potencial para el monitoreo de variables en lagos y entornos marinos. Una WSN de este tipo permitiría monitorear variables tales como temperatura, presión, conductividad y PH entre otras.

En [6] se presenta un diseño mejorado de una WSN ZigBee, donde el coordinador sólo se encarga de cumplir con esa función dentro de la red, dejando el resto de las tareas a otro procesador. Con esto se obtiene una mejor respuesta para un sistema de tiempo real, menor pérdida de paquetes y mayor estabilidad. El módulo ZigBee utilizado fue el CC2430 de Texas Instruments. La conexión entre el coordinador y el procesador es a través de una interfaz RS232.

En [7] se analiza la topología de red, tipos de dispositivos y la arquitectura del protocolo ZigBee. Se utilizó para los ensayos el módulo CC2530 de Texas Instruments y el protocolo Z-Stack. Se evaluó la distancia y la cantidad de paquetes perdidos. Con el uso de este hardware se obtuvieron comunicaciones de buena calidad a mayores distancias.

En [8] se describe una WSN ZigBee realizada con módulos CC2530, equipada con un sensor de temperatura DS18B20 y un indicador de calidad de la señal, para verificar el funcionamiento del sistema. El nodo se compuso de procesador, módulo de RF, sensor, módulo de potencia, switches, leds e interfaces. Para el sistema de comunicaciones se utiliza la pila de protocolos Z-Stack, constituido por las capas PHY y MAC definidas en IEEE 802.15.4 y el framework provisto por ZigBee Alliance para las capas de Red y Aplicación. El escenario planteado fue de un nodo con sensor de temperatura y cinco nodos operando como routers. Además de

monitorear la temperatura también se evaluó la calidad del enlace con el indicador LQI. El LQI es proporcional a la energía de RF del dispositivo sensor. Se pudo observar que los obstáculos influyen significativamente en la comunicación.

En [9] se implementa un algoritmo que modifica ciertos parámetros de la capa MAC, mejorando la confiabilidad de la transmisión en una WSN ZigBee. El Algoritmo ADAPT (ADaptive Access Parameter Tuning) ajusta dinámicamente los parámetros MAC, basándose en el nivel deseado de confiabilidad y las condiciones actuales de operación experimentadas por el nodo. Los parámetros que se evaluaron fueron la tasa de paquetes entregados (rendimiento) y el consumo de energía. Se efectuaron diversos análisis. En primera instancia se comparó el desempeño de la red ZigBee con el uso de los protocolos de enrutamiento AODV y DSDV. Luego se comparó una red sin agrupar nodos y otra con nodos agrupados. A continuación se evaluó la red usando AODV pero con distintos tiempos de simulación. Por último se analizó el rendimiento de la red usando el algoritmo ADapt propuesto.

En [10] se presentan dos modificaciones sobre el modo MAC LLDN definido por el estándar 802.15.4e. La primera permite obtener un tiempo de configuración más reducido y predecible y el segundo se centra en el peor caso de latencia para tráfico de alta prioridad. Se comparó con el estándar LLDN a través de simulaciones por el método Monte Carlo y también analíticamente. Los resultados obtenidos mostraron que las modificaciones a la norma permitieron lograr un aumento del determinismo en el estado de configuración y una más baja latencia para el peor caso en el estado activo. Esto conduce a un uso más seguro de la WSN para aplicaciones del entorno industrial.

En [11] se presenta un algoritmo de encaminamiento jerárquico denominado Cluditem, implementado en la capa de red, integrado con los niveles inferiores del estándar IEEE 802.15.4, sobre un nodo CC2530. Cluditem está basado en clusters y realiza adquisición de datos en forma periódica, enviando información a la estación base en cada período de medición (T). El funcionamiento de Cluditem se divide en tres fases bien diferenciadas. La primera se ocupa del establecimiento del árbol de encaminamiento, la segunda se encarga del envío de datos al sink y durante la tercera los dispositivos permanecen en estado de bajo consumo (sleep).

En [12] se presenta un estudio de las tecnologías usadas en WSN, principales aplicaciones y estándares, características de diseño y evolución. En particular se analizaron aplicaciones de monitoreo del medio ambiente y se puso énfasis en el estándar IEEE 802.15.4 con el objetivo de ayudar a los diseñadores en la elección de la tecnología más apropiada.

Se espera que ZigBee tenga un gran crecimiento en sistemas de monitoreo médico, por las ventajas que representa su bajo costo, potencia de señal segura y fácil implementación. Sin embargo, en transmisión de datos intensivos el coordinador de una red ZigBee puede ser un cuello de botella y hacer una transmisión poco confiable para datos críticos. En [13] se propone un método CTD (Coordinator Traffic Diffusion) el cual redirecciona el tráfico excesivo desde el coordinador al sink. El diseño del CTD fue probado en una plataforma ZigBee CC2530 y simulado en ns2.

Los resultados experimentales demostraron que CTD puede asistir a los routers en la entrega de muestras de datos de un ECG (electrocardiógrafo) en tiempo real de manera satisfactoria con la mejor tasa de transmisión de 24 kb/s. Este desempeño no puede ser logrado por un diseño de ZigBee original.

En [14] se ve otro caso de uso intensivo de la WSN para enviar muestras de datos analógicos. En este caso se trata de un dispositivo que mide vibraciones mecánicas en una máquina y transmite las muestras correspondientes a la computadora encargada del correspondiente registro y procesamiento. El sensor utilizado tiene un ancho de banda de 264 Hz, y fue muestreado con una tasa de 1 kHz, con 16 bits. La carga impuesta a la red es baja, de aproximadamente 16 kbps. Sin embargo, los autores señalan la conveniencia de utilizar sensores de más ancho de banda, proponiendo uno de 10 kHz de ancho de banda. En tal caso, la tasa de transferencia requerida sería de 320 kbps, lo que obligaría a utilizar algún esquema de compresión que permita reducir la carga a la mitad

4 Ensayos Realizados

En [3] se realizaron ensayos con el fin de determinar la zona óptima de trabajo de una red 802.15.4. El análisis se llevó a cabo teniendo en cuenta las potenciales aplicaciones de una red de este tipo, diferenciando dos posibles escenarios:

- Uno para aplicaciones que utilizan la red para transportar comandos y respuestas y/o lecturas de sensores de variables relativamente lentas, que implican el envío de pequeños paquetes, con una cierta frecuencia de transmisión.
- Otro para aquellas aplicaciones que hacen un uso intensivo de la red para el transporte de volúmenes de datos, incluso con restricciones temporales.

El objetivo fue obtener la máxima tasa de transferencia de datos, medida en kbps, para una tasa de pérdida de paquetes que resulte aceptable para una determinada aplicación.

Los distintos ensayos han sido realizados en ambientes sin ruido ni interferencias en el canal utilizado. Los dispositivos empleados [15] fueron tres placas de evaluación SmartRF06, cada una de ellas equipada con un módulo CC2538 y un dongle USB CC2531, utilizado como sniffer. El método MAC usado fue CSMA/CA.

En los dos ensayos planteados se evaluó:

- Porcentaje de paquetes recibidos, para para un nodo transmisor y un receptor.
- Porcentaje de paquetes recibidos, para dos transmisores y un receptor.

A partir de los mismos también se obtuvieron tasa efectiva de bits recibida por el receptor, medida en kpbs para uno y dos transmisores como así también la tasa de pares comando-respuesta por segundo que es posible transferir por esta red, en función de la tasa de envío de paquetes por segundo y la probabilidad de pérdida de paquetes asociada.

4.1 Porcentaje de paquetes recibidos, para un nodo transmisor y un receptor.

El objetivo de este ensayo ha sido obtener un marco de referencia en cuanto a las prestaciones máximas que sería posible obtener de esta red. Se ha realizado transmitiendo paquetes con 20, 50 y 100 bytes de carga (más 11 bytes adicionales del encapsulado de 802.15.4), con tasas de envío de entre 10 y 1000 paquetes/s. Se utilizó

CSMA/CA para controlar el acceso al medio y hubo un sólo nodo transmitiendo hacia un único receptor. Los resultados de este ensayo se presentan en la Fig. 2.

Es posible observar que, independientemente del tamaño de las tramas, el límite práctico de trabajo para un único nodo transmisor está alrededor de los 200 paquetes por segundo. En el caso de enviar paquetes con 100 bytes de carga útil, la tasa de transferencia alcanzada es de aproximadamente 160 kbps.

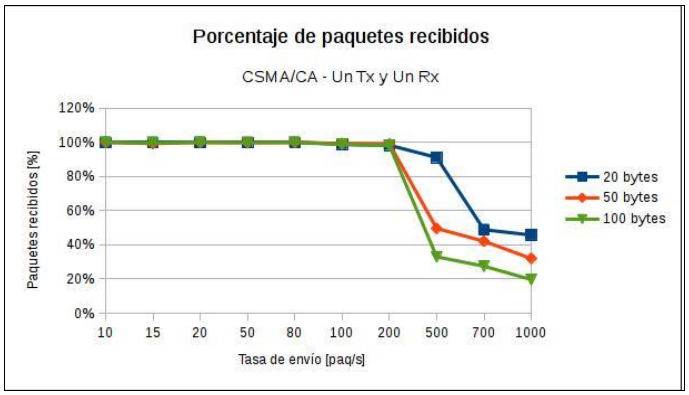


Fig. 2. Porcentaje de paquetes recibidos, para un único transmisor, usando CSMA/CA

4.2 Ensayos con dos transmisores y un receptor.

En este otro ensayo se plantea una situación más frecuente en las WSN, como es la competencia entre dos o más nodos por el uso del medio. Los ensayos fueron realizados utilizando dos nodos transmisores y un receptor.

En la Fig. 3 se presentan los resultados obtenidos, en términos del porcentaje de paquetes recibidos con respecto al total enviado por los dos transmisores. Se enviaron paquetes con 20, 50 y 100 bytes de carga útil, y tasas de envío de entre 10 y 1000 paq/s.

Comparando con los resultados del ensayo anterior, se puede observar que la performance ha sufrido una cierta reducción. Si en aquel caso se podía llegar a una tasa de 200 paq/s con una probabilidad de pérdida de paquetes del 1%, en este otro caso se ve que es esperable alcanzar una tasa de envío de 100 paq/s por cada nodo (que equivale a una tasa agregada de 200 paq/s), pero con una probabilidad de pérdida de paquetes de entre 4% y 6%.

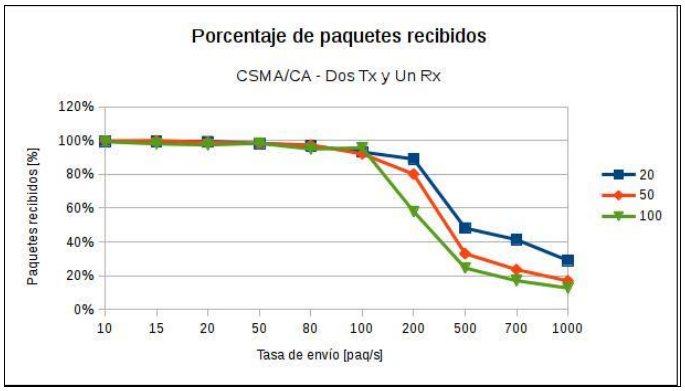


Fig. 3. Porcentaje de paquetes recibidos, para dos transmisores y un receptor

4.3 Tasa efectiva de transferencia para uno y dos transmisores.

Otros resultados, basados en valores obtenidos de los mismos ensayos ya mencionados, corresponden a la tasa efectiva de bits recibida por el receptor medida en kpbs, que se pueden observar en la Fig. 4.

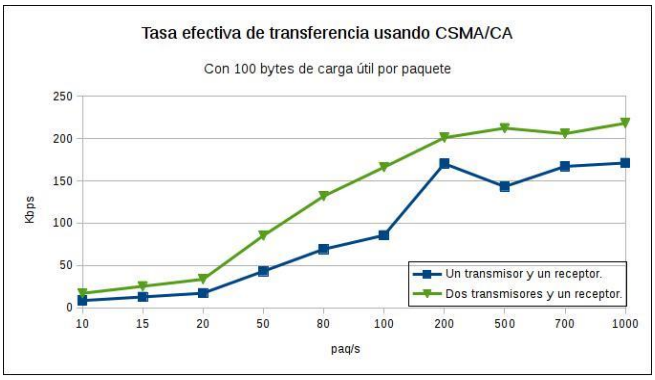


Fig. 4. Porcentaje de paquetes recibidos, para dos transmisores y un receptor.

4.4 Desempeño de la red para envío de comandos y respuestas, en función de la pérdida de paquetes.

En aquellas aplicaciones basadas en el envío de comandos y respuestas, la pérdida de un paquete, ya sea comando o respuesta, obliga a la retransmisión de otro par. Esta situación se puede observar en la Fig. 5.

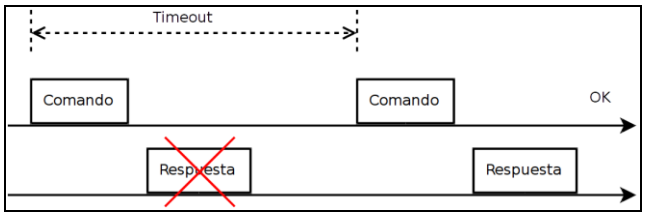


Fig. 5. Retransmisión debida a la pérdida de un paquete.

En la Fig. 6 se presenta la tasa de pares comando-respuesta por segundo que es posible transferir por esta red, en función de la tasa de envío de paquetes por unidad de tiempo y la probabilidad de pérdida de paquetes asociada (Fig. 2 y 3)

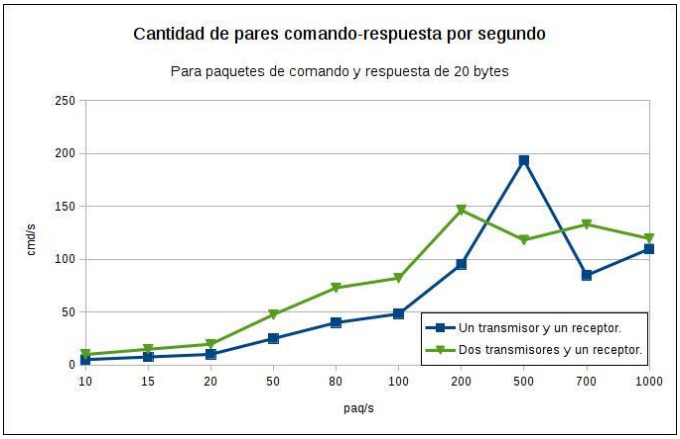


Fig. 6. Tasa máxima de pares comando-respuesta por segundo, en función de la tasa de envío de paquetes y la probabilidad de pérdida de paquetes asociada.

Cabe tener presente que la tasa de envío (paq/s) indicada en el eje x es por cada transmisor: Por ejemplo: para 100 paq/s y un solo transmisor, con 1% de pérdidas, se puede llegar a 49 pares comando-respuesta por segundo; para dos transmisores, con 9% de pérdidas, se podrán enviar hasta 82 pares comando-respuesta por segundo.

4.5 Ensayo de transmisión de datos entre dos nodos, pasando por un nodo enrutador.

Continuando con los ensayos, y en busca de incorporar otras condiciones típicas de operación, en este ensayo se evaluó la transmisión de datos entre un nodo emisor y otro receptor pasando por un nodo enrutador. El nodo receptor envía una respuesta al emisor por cada paquete recibido, que también debe pasar por el nodo enrutador. La Fig. 7 muestra la topología utilizada en el ensayo.

Se midió el tiempo de ida y vuelta, con paquetes de 50 y 100 bytes de carga útil. Los retardos obtenidos fueron de 5.95 y 9.3 ms, respectivamente. En función de estos tiempos es de esperar una tasa de envío máxima de alrededor de 100 paq/s, lo que se pudo verificar exitosamente, como así también que por encima de esa tasa el rendimiento cae abruptamente, debido a que entran en competencia por el acceso al medio el nodo transmisor y el nodo enrutador, e incluso el nodo destinatario (en función de las distancias entre nodos utilizadas en el ensayo).

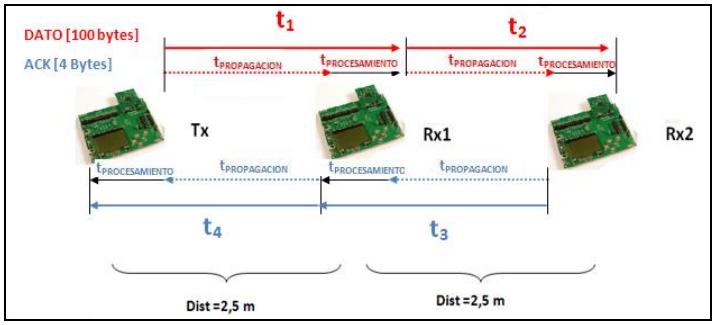


Fig. 7. Topología del ensayo con un nodo intermedio.

Debido a las cortas distancias entre los tres nodos, el protocolo CSMA/CA permite arbitrar el acceso al medio de los tres nodos. En próximos ensayos se adoptarán distancias mayores, a fin de que se pueda poner de manifiesto el efecto cercano-lejano entre los nodos más alejados con respecto al nodo enrutador. Otros ensayos previstos, cuando se adquieran otros dispositivos más, implican adoptar topologías más complejas, conformando una malla o parte de ella.

5 Conclusiones

Los ensayos realizados han permitido obtener algunos indicadores del estado de operación de una red 802.15.4, que posibilitan planificar su utilización de manera certera. Sin embargo, está prevista la realización de más ensayos, con topologías más complejas y distintos escenarios, en la medida en que se pueda ampliar la cantidad de dispositivos disponibles para los ensayos.

Para aquellas aplicaciones basadas en el envío de paquetes pequeños, para comando, control y/o mediciones de algunas variables, en las que es deseable una muy baja probabilidad de pérdida de paquetes, se pudo determinar que el punto más conveniente de operación es por debajo de los 200 paquetes por segundo, como total de lo enviado por los distintos nodos. Para un solo nodo transmisor la pérdida de paquetes se ubicó en un 1%, mientras que para dos transmisores la pérdida de paquetes alcanzó el 9%. Cabe efectuar otros ensayos con mayor cantidad de nodos compitiendo para transmitir, a fin de evaluar la degradación del rendimiento de la red.

Otro caso son las aplicaciones con requerimientos de transmisión intensiva de datos, como por ejemplo las que transmiten muestras de una señal analógica, que deberían hacer uso de paquetes grandes (unos 100 bytes de carga útil). Se ha podido

observar que la zona de operación más conveniente está por debajo de los 200 paquetes por segundo, pudiendo alcanzarse una tasa de transferencia efectiva de entre 160 y 170 kbps. También, para estos casos se propone realizar otros ensayos con mayor cantidad de nodos compitiendo por el acceso al medio, a fin de evaluar la degradación del rendimiento.

Otra línea de trabajo está en el análisis del desempeño de la red desde el punto de vista de los requerimientos de tiempo real, en función de los retardos que se introducen al tener que retransmitir a causa de un paquete perdido, como así también cuando la topología es más compleja y se debe saltar por distintos nodos enrutadores.

Referencias

1. "IEEE Std 802.15.4; (2011): Low-Rate Wireless Personal Area Networks (LR-WPANs)
2. http://www.zigbee.org/home.aspx
3. Friedrich G, Reggiani G, Cofre L,Velazquez G y Moreno W, "Determinación de la zona óptima de trabajo de una red ZigBee", VI Congreso de Microelectronica Aplicada (UEA) 2015 – La Matanza, Buenos Aires. ISBN: 978-987-3806-24-7
4. Changhai Peng, Kun Qian and Chenyang Wang, "Design and Application of a VOC-Monitoring System Based on a ZigBee Wireless Sensor Network", IEEE Sensors Journal, Vol. 15, No. 4, April 2015
5. LieGuo Wu, Lingcheng Kong and Zhihua Zhang, Water Environment Monitoring System Based On ZigBee Wireless Sensor Network", Software Engineering and Service Science (ICSESS), 2013 4th IEEE International Conference on
6. Liu Yanfei , Wang Cheng, Qiao Xiaojun, Zhang Yunhe and Yu Chengbo, "An Improved Design of ZigBee Wireless Sensor Network", Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on
7. Jian-Ming Liao, Xue-Qin He, Guo-Ming Lu and Si-Yu Zhan, "The Research and Design of ZigBee Wireless Networking based on CC2530", Wavelet Active Media Technology and Information Processing (ICCWAMTIP), 2013 10th International Computer Conference on.
8. Wangqun Jie, Shiyan Ying, Biaobiao Zhang, Jintian Ma and Hui Wang, "Development of a ZigBee-based wireless sensor network system", 2012 International Conference on Control Engineering and Communication Technology.
9. Sangeetha C P and C. D. Suriyakala, "Performance Analysis of IEEE 802.15.4/ZigBee Sensor Networks using ADAPT Algorithm" 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)
10. Dariz, L., Malaguti, G. and Ruggeri, M., "Performance Analysis of IEEE 802.15.4 real-time Enhancement", Industrial Electronics (ISIE), 2014 IEEE 23rd Intl. Symposium on.
11. Corti R., Belmonte J., Giandoménico E. y Martínez R, "Plataforma software de un nodo sensor basado en el System on Chip CC2530", V Congreso de Microelectrónica Aplicada (UEA) 2014 – Cordoba.
12. C.Buratti et al. (2009, August). An Overview on Wireless Sensor Networks Technology and Evolution, Sensors [Online]. 9 (9) Disponible: http://www.mdpi.com
13. Chinyang Henry Tseng, "Coordinator Traffic Diffusion for Data-Intensive Zigbee Transmission in Real-time Electrocardiography Monitoring", IEEE Transactions on Biomedical Engineering, VOL. 60, NO. 12, DECEMBER 2013
14. Banfi D., Friedrich G., Banchieri M. y Baldini P, "Medición de vibraciones mecánicas mediante el uso de acelerómetros MEMS", VI Congreso de Microelectrónica Aplicada, (uEA 2015), La Matanza, Bs. As., 2015.
15. http://www.ti.com/tool/cc2538dk

IV Workshop de Seguridad Informática (WSI)

7416 | Plan Estratégico para la implementación de un Centro de Servicios de Informática Forense
Sergio Daniel Appendino (UCASAL), Fredi Aprile (UCASAL) y Herminia Beatriz P. De Gallo (UCASAL)

7469 | Automated Analysis of Source Code Patches using Machine Learning Algorithms
Antonio Castro Lechtaler (IESE), Julio Cesar Liporace (UCA), Marcelo Cipriano (EST-IESE), Edith Garcia (IESE), Ariel Maiorano (IESE), Eduardo Malvacio (EST-IESE) y Nestor Tapia (EST-IESE)

7549 | El tratamiento de la evidencia digital y las normas ISO/IEC 27037:2012
Santiago Roatta (UAI), María Eugenia Casco y Martin Fogliato (UNR)

7564 | Probabilidades de referencia para aplicar en la detección de Infraestructuras de Clave Pública anómalas
Antonio Castro Lechtaler (IESE), Marcelo Cipriano (EST-IESE) y Eduardo Malvacio (EST-IESE)

7570 | Controles de Seguridad propuesta inicial de un Framework en el Contexto de la Ciberdefensa
Pablo Gastón Sack (UM) y Jorge Ierache (UM)

7645 | Automatizando el descubrimiento de portales de autenticación y la evaluación de la seguridad mediante ataques de fuerza bruta en el marco de una auditoría de seguridad
Gastón Traberg (UNLP), Lía Molinari (UNLP), Paula Venosa (UNLP), Nicolás Macia (UNLP) y Einar Felipe Lanfranco (UNLP)

Plan Estratégico para la implementación de un Centro de Servicios de Informática Forense

Sergio Daniel Appendino[1], Fredi Aprile[1], Herminia Beatriz P. De Gallo[1]

[1] Grupo de Investigación sobre Informática Forense
Facultad de Ingeniería de la Universidad Católica de Salta,
Salta, Argentina
{sappendino, faprile,bgallo}@ucasal.edu.ar

Abstract. El proceso de implementación de un centro de servicio profesional y científico supone la formulación de un plan estratégico que oriente la propuesta. La definición de la misión esencial, la visión que se persigue y los objetivos marcan las líneas a seguir en la transformación paulatina de la idea inicial en un hecho concreto. Elaborados estos componentes, se toman de base para definir un Plan de Acción que vuelque a la práctica la idea inicial, dando forma a una institución que cada día tiene mayor demanda, tanto desde los contextos judiciales como empresariales: un Centro de Servicios de Informática Forense.

Keywords: Informática Forense

1 Introdución

El presente documento resulta de las conclusiones arribadas en el Proyecto de Investigación sobre "APLICACIÓN DE METODOLOGIAS, PROCESOS Y TECNICAS PARA LA REALIZACION DE PERICIAS INFORMATICAS" aprobado por Resolución Rectoral N° 332/11 y desarrollado en la Facultad de Ingeniería de la UCASAL. Con el objetivo de estudiar y definir una metodología científica para el desarrollo del proceso de pericias informáticas acorde al proceso judicial propio del Poder Judicial de la Provincia de Salta, el proyecto fue desarrollado durante los años 2012 y 2013, y el presente informe muestra parte de los resultados logrados en el mismo.

2 Conceptos Metodológicos

Los autores referentes del tema identifican tres etapas en la planificación estratégica (formulación, implementación y evaluación) partiendo de la definición de la misión, visión, el análisis interno y del entorno para concluir en los objetivos que se persiguen.
Se propone el enfoque estratégico como un mecanismo adecuado para este trabajo ya que la formulación de una estrategia debe hacerse respetando el ser de la

organización, el deber ser, los escenarios probables, los recursos internos y los desafíos que se le presentan.
En resumen, los pasos a seguir son: a) Definición de la Misión y la Visión del Centro de Servicios de Informática Forense; b) Análisis del Contexto Externo e Interno; c) Formulación de Estrategias; y d) Plan de acción.

2.1 Definición de la Misión y la Visión

Declarar la ***Misión Institucional*** significa responder a la pregunta ¿Qué nos hace "ser"? ¿Cuáles son las características esenciales que nos distinguen de otros?
Un Centro de Servicios de Informática Forense es una institución que surge de la necesidad de dar respuesta a la demanda de la justicia respecto de la recolección de evidencias significativas o determinativas para ser presentadas como pruebas en un proceso judicial y que permita la determinación del/los responsable/s de delito/s tanto informáticos como otros contemplados en el Código Procesal Penal y cuya prueba se encuentra conservada en dispositivos de Tecnología de Información (TI). Se declara entonces como Misión del Centro de Servicios de Informática Forense: "Ser un colaborador de la justicia en la identificación de evidencias relevantes para ser presentadas en juicio".
La ***Visión Estratégica*** permite identificar ¿Cómo queremos cumplir nuestra misión?, i.e., define el camino a seguir para no desviarnos de la misión esencial. Tratándose de acciones que impactan directamente en la seguridad, la vida y la libertad de las personas, es imprescindible que el servicio a brindar se sustente en la ciencia y la técnica, dando rigor y formalidad profesional a las actividades que se desarrollan.
Es así que definimos como Visión del Centro de Servicios de Informática Forense: "Ser una institución referente para la Justicia", para lo cual formulamos los siguientes propósitos: a) Sustentar nuestro trabajo en la ciencia, el rigor metodológico y la formalidad professional; b) Contribuir al desarrollo de la Informática Forense; y c) Actualizar nuestra base de conocimientos de manera acorde al desarrollo tecnológico.

2.2 Análisis del Contexto Externo e Interno

Además del ***contexto básico*** conformado por las instituciones jurídicas, se debe considerar el ***contexto tecnológico*** que impacta también en la comisión de delitos.
Por su parte las instituciones jurídicas (Poder Judicial, Ministerio Público, comunidad forense) se caracterizan por el desconocimiento de las tecnologías informáticas y de las comunicaciones, y se ven sorprendidas y avasalladas por la diversidad y complejidad de la "prueba digital". Esta situación dificulta la tarea pericial debido principalmente a que el promotor de dicha prueba generalmente desconoce lo que está solicitando. Se observa que el ambiente jurídico, principal demandante de la Informática Forense, presenta características que dificultan el desarrollo del Centro de Servicios de Informática Forense, a saber: a) Barreras de comunicación muy altas con los actores judiciales (abogados, jueces, Fiscales, personal administrativo); b) Dificultades para comprender y absorber la prueba intangible; c) Dificultad para especificar los puntos de pericia desde el vocabulario técnico; d) Legislación muy

genérica y poco actualizada; y e) Falta de tipificación de algunos de los delitos más comunes relacionados con las Tecnologías de la Información y las Comunicaciones.
Por su parte el contexto tecnológico actual no detiene su crecimiento y diversificación, generando un ámbito de dependencia tecnológica sin posibilidades de obviar. Esto impacta cada vez en el acto delictivo, haciendo que la "prueba digital" esté presente cada vez con más fuerza en las evidencias de la comisión del delito, abriendo posibilidades de intentar comprender lo que pasó antes, durante y después del hecho.
Un apartado especial merece el aporte de la investigación en las ciencias de la criminalística y su relación con la Informática Forense, dando el ámbito adecuado para que desde el conocimiento Científico Tecnológico se logre la identificación, preservación, adquisición, análisis, documentación y presentación de la prueba, en especial a partir de la consideración de la prueba digital como "prueba indiciaria" sumando el componente "virtual" a los hechos investigados.
La conformación del Centro de Servicios de Informática Forense supone varios desafíos internos. Por una parte, resulta necesario considerar el ambiente interdisciplinario que se requiere para el desarrollo adecuado de la Informática Forense. No solo desde su vinculación con las ciencias jurídicas y criminalísticas, sino con otras disciplinas tecnológicas como la electrónica, las telecomunicaciones, la ingeniería de sistemas y el tratamiento de imágenes, videos y voz. Considerar este ámbito plural ayuda a definir el alcance de la actividad pericial. Por otra parte, al ser tan amplia la brecha de comunicación con el contexto externo (ámbito judicial), requiere un esfuerzo extra al profesional tecnológico que debe introducirse en la ciencia jurídica mediante la capacitación y la actualización continua para comprender cabalmente el ámbito en el que debe desarrollar su trabajo.

2.3 Formulación de Estrategias

La conformación del Centro de Servicios de Informática Forense requiere de un trabajo orientado en cinco líneas:
Creación del Centro de Servicios de Informática Forense: Resulta fundamental definir las características institucionales del Centro, mediante la descripción de las funcionalidades y la estructuración organizacional del mismo.
Generación de acciones de capacitación: desde dos espacios de trabajo. Por un lado se requiere la instrucción necesaria a impartir en las áreas internas al Centro de Servicios de Informática Forense, que tienen que ver con la formación específica en Informática Forense y en particular sobre la identificación de pruebas. Por otro lado, es necesario capacitar al cuerpo de profesionales judiciales (jueces, fiscales, abogados en general, administrativos), mediante un plan de alfabetización tecnológica, delitos informáticos, tecnologías informáticas y de las comunicaciones y su impacto en la comisión de delitos. Estas acciones de capacitación se proponen como línea de trabajo continua, de manera que –mantenida en el tiempo- promueva la concientización y la culturalización de los recursos humanos en el tema que nos ocupa. El plan de capacitación deberá tener presente la conformación de grupos de capacitandos en función de los criterios habituales para estos casos (grado de conocimiento

tecnológico de la audiencia, niveles de la estructura funcional de la organización, interés o motivación en el tema, etc.)

Conformación de un Laboratorio de Informática Forense: mediante la incorporación de personal especializado en las distintas disciplinas de la TI Forense, como el tratamiento de las imágenes, videos y la voz, las Telecomunicaciones y la Informática y también mediante la adquisición del equipamiento necesario de herramientas Forenses, tanto Software como Hardware e Infraestructura, con el fin de contar con una batería de herramientas para atender las distintas alternativas que pueden surgir a partir de los distintos soportes de prueba, sus sofisticaciones, ámbitos geográficos y plataformas tecnológicas. Esta línea de trabajo presupone un plan de adquisición de productos de hardware y software que permitan –a mediano y largo plazo- la constitución de un ámbito tecnológico adecuado para garantizar la realización de pruebas y exámenes forenses con el marco científico y metodológico necesarios. A corto plazo debe implementarse una red de puestos de trabajo en cantidad suficiente, con equipamiento informático básico de última generación, más la instalación complementaria correspondiente (cableado estructurado, servidores, impresoras, etc.). A mediano plazo y largo plazo se debe incorporar componentes específicos para la actividad: Herramientas de hardware y software para la Satinado, Recolección y Preservación de evidencias; para el Análisis de Evidencia y para la Presentación de Informes.

Desarrollo de procedimientos técnico-legales: para garantizar la revisión científica, tecnológica y técnica de la prueba indiciaria es necesario establecer procedimientos formales, basados en normas de calidad, acerca de todos los procesos involucrados en la actividad pericial: recolección de la prueba y cadena de custodia, análisis y diagnóstico tecnológico, elaboración del informe técnico de pericia.

Plan de Crecimiento: con una mirada táctica/estratégica del sector demandante, es necesario tener en cuenta que en un futuro no muy lejano se requerirá una estructura de recursos humanos, tecnológicos y de infraestructura necesaria para alinear el servicio a las necesidades venideras. Las acciones formuladas deben consolidarse en el tiempo, mediante la realización de actividades que fortalezcan la propuesta. A saber: a) Tener presente los probables escenarios futuros en el desarrollo de la disciplina, contemplando las variables tecnológicas, sociales y su impacto en la comisión de delitos; b) Buscar el asesoramiento técnico de profesionales referentes en el ámbito de la Informática Forense, a fin de revisar y orientar las acciones hacia la búsqueda de la calidad; c) Monitorear el avance de la tecnología y de la ciencia con el objetivo de constituir la mejor arquitectura tecnológica para el desarrollo de la práctica de la Informática Forense; y d) Validar y ajustar los procedimientos técnico-legales mediante la experiencia continua.

3 Plan de Acción

Cada una de las estrategias definidas se concreta en un Plan de Acción. Si bien en el proyecto de investigación se ha destinado un espacio para desarrollar y formular en detalle cada una de estas estrategias, por razones de espacio en el presente documento se describen los más destacados: el Proyecto de Creación del Centro de Informática

Forense y el Procedimiento de Incautación de Pruebas Digitales, como un ejemplo del apartado correspondiente al desarrollo de los procedimientos técnico-legales.

3.1 Proyecto de Creación del Centro de Informática Forense

Con el objeto de formular los objetivos, características y descripción organizativa de un Centro de Servicios de Informática Forense que cubra el marco de lo estudiado en el presente proyecto de investigación, resulta necesario considerar un contexto real – el Gabinete de Informática del Ministerio Público de Salta- del cual partir, para analizar las fortalezas y debilidades actuales y formular los elementos que deberán ser considerados, a fin de formular una propuesta estratégica para la creación del Centro de Servicios.

Ante un ambiente en constante crecimiento y cambio, como lo es el de las Tecnologías con soporte Digital, la incorporación, la capacitación y el desarrollo de las distintas especialidades técnicas que den solución a los requerimientos recibidos de una manera adecuada y con soporte de conocimientos científicos forenses, hacen necesario repensar y plantear una nueva estructura funcional organizativa que contenga e integre el marco de conocimientos. Asimismo, el significativo e incesante crecimiento del número de solicitudes de Informes Técnicos por parte de las Fiscalías y la amplitud de conocimientos técnicos necesarios para resolver los requerimientos en el Área Digital hizo necesario desarrollar y ampliar la estructura de infraestructura y tecnologías de análisis de evidencias necesarias a fin de brindar adecuadamente los resultados del Área Forense Digital.

Tomando como caso ejemplo el Cuerpo de Investigaciones Fiscales (CIF) del Ministerio Público de la Provincia de Salta, es dable destacar que ante la importancia que aporta la Evidencia Digital puesta de manifiesto por los medios de comunicación tanto locales como Nacionales e Internacionales es necesario definir una estructura de Recursos Humanos, roles y responsabilidades dividida por funciones y que presenten la autonomía adecuada de funcionamiento del resto de las disciplinas definidas en el Artículo 7°- Estructura de la Ley 7665 de creación del CIF y compongan el área del campo del conocimiento Técnico Científico específico.

Las áreas de incumbencia propuestas dentro de la especialidad Forense Digital y que darían soporta a los procesos de evaluación y emisión de Informes Técnicos del CIF son:

Seguridad y Soporte Técnico Servidor CIF (Área Staf): Debido a la sensibilidad de información que se administra el Cuerpo de Investigadores Fiscales, a la Ley 7775 promulgada por el Senado y la Cámara de Diputados de la Provincia de Salta, "Registro Provincial de Condenados vinculados a Delitos contra las Personas y contra la Integridad Sexual" y a la incorporación de nuevas tecnologías que se están realizando en todas sus áreas, es necesario contar con un profesional técnico que se especialice en brindar soluciones tanto para la seguridad de la información interna como para el mantenimiento de la infraestructura de procesamientos de datos del todas las áreas del CIF. Asimismo este profesional, al estar fuera de la estructura operativa del trabajo diario, será el que lleve a cabo las evaluaciones de nuevas tecnologías que se implementen, tanto las internas para dar soluciones a los trabajos de las áreas técnicas del CIF como las que podrían ser utilizadas por el crimen

organizado. Eventualmente puede interactuar con estructuras de investigación a nivel Nacional como la Comisión Nacional del Cibercrimen, creada por la Jefatura de Gabinete de Ministros y el Ministerio de Justicia y Derechos Humanos, por Resolución Conjunta 866/2011 y 1500/2011.

Video, Imagen y Voz Forense: Área de análisis, desarrollo y soporte que tiene como funciones: a) Verificación e identificación de voz basadas en conversaciones por medio de herramientas biométricas Forense, como el software Batvox; b) Diseño de reconstrucciones virtuales de hechos investigados por medio de las herramientas de informática recientemente adquiridas como el Google Sketch Up Pro 8, Poser Pro 2012, Scene PD 5, Easy Street Drow 5, Adobe Photoshop CS6 y Corell Draw Graphics Suite X5; c) El mejoramiento de imágenes contenidas en fotos y/o videos por medio del software de análisis forense como por ejemplo el AmpedFive. Teniendo en cuenta que se encuentra actualmente en proceso de instalación 1.000 cámaras de seguridad de alta resolución en le provincia de Salta, 600 dentro del ejido Urbano de la Capital de Salta en donde se preveen para el área una significativa cantidad de trabajo; y d) Soporte técnico documental de video e imágenes relativos a las funciones del CIF (allanamientos, crímenes, accidentes, etc.) como también institucionales (videocámaras, conferencias, visitas, etc.).

Tecnología de Información Forense: Área de análisis y desarrollo que tiene como funciones: a) Verificación e identificación de evidencia digital contenida en soportes de medios magnéticos como son los discos rígidos, Pen-drives, tarjetas de memoria, etc. por medio del software Encase Forensics; b) Identificación y consolidación de datos para el desarrollo de estructuras de Bases y Bancos de Datos a fin de obtener, administrar y preservar la información relacionada a las evidencias digitales

Telecomunicaciones Forenses: Área de análisis y desarrollo que tiene como funciones:a) Verificación e identificación de evidencia digital contenida en soportes de aparatos de telefonía y geo-referenciados como son los aparatos celulares, tablet PC, GPS, cámaras de fotografía con GPS, etc., por medio de la herramienta de Hardware UFED y también de software MobilEdit; y b) Canalizar las solicitudes de información de las Fiscalías y servir de vínculo con las empresas prestatarias del servicio de Telefonía, sistematizando la información solicitada a fin de que sea posible la eficiente utilización del Software de entrecruzamiento de llamadas I2 Analyst´s Notebook.

3.2 Desarrollo de procedimientos técnico-legales

Resulta necesario formalizar los procedimientos para la recolección de la prueba y cadena de custodia, análisis y diagnóstico tecnológico, y la elaboración del informe técnico de pericia. Esta formalización permite ajustar el procedimiento a una norma que garantice la utilización de criterios metodológicos y buenas prácticas en la actividad, a saber:

Procedimiento para la Incautación de la Prueba Digital

Es de suma importancia tratar de individualizar acciones adecuadas que puedan llevarse a cabo cuando se trata de recuperar potenciales fuentes de pruebas digitales

de los lugares controlados por un sospechoso. Existe una diferencia radical entre recuperar físicamente los dispositivos y después examinarlos en un laboratorio forense y, capturar los datos de dispositivos antes de que sean apagados o desconectados de las redes o fuentes de alimentación. Este último requiere un mayor nivel de especialización y no debería ser llevado a cabo por cualquier persona que no tenga la formación adecuada y que está calificado para realizar el trabajo.

Preparar arribo a la escena

El proceso de planificación y preparación debe permitir una fácil identificación del nivel de apoyo que se requerirá en la escena, en situaciones en las que se pueden encontrar pruebas digitales.
En la legislación de la Provincia de Salta el único Organismo autorizado a realizar allanamientos y secuestros de incautación ordenadas por el Juez (o Fiscal) es la Policía de Salta, en el caso de secuestro de evidencia digital la persona responsable del operativo debería informar la necesidad de contar con el apoyo de una unidad de identificación compuesta por especialistas, en el caso de Salta, el Cuerpo de Investigadores Fiscales (CIF)
En caso de un escenario que no es ordenado por la Justicia es recomendable contar con la asistencia de un Escribano que certifique, tanto el escenario en general como todas las operaciones realizadas sobre la evidencia, y sea éste el que dirige el proceso de secuestro o incautación de elementos. La primera decisión a tomar es qué tipo de la incautación se debe realizar, si deberán secuestrarse equipos o se realizará la captura de datos en directo, inclusive puede ser necesaria una combinación de ambos. Si bien un procedimiento de allanamiento se inicia con la orden de un Juez con detalle del trabajo que se debe realizar, hay rango de decisiones que serán tomados en el lugar, cuando las circunstancias son más claras. Sin embargo sería deseable contar con la mayor cantidad información posible y que con antelación se conozca qué sistema de TI y las posibles fuentes de pruebas que puedan existir en el lugar de la escena. Los ejemplos de los tipos de información que ayuden a la planificación incluyen: Hardware/sistema operativo/software/aplicaciones y medios de almacenamiento relacionados con la información, la comunicación y la información relacionada con la red (ISP, teléfono, fax, módem , equipos de red LAN , etc.); Responsable del sistema informático y/o de la red (por ejemplo , si tiene un administrador o es administrada por una empresa externa); Cuantos equipos se puede encontrar; Qué cantidad de datos se pueden copiar y ¿existe una copia de seguridad del sistema disponible en medios de almacenamiento
La fase de preparación incluye los siguientes pasos: Asegurarse de que la toma de pruebas digitales esté autorizada correctamente e interpretar correctamente la orden judicial; en caso de que se deban recolectar datos de otra jurisdicción Judicial, asegurarse que se encuentra autorizado por la autoridad Judicial correspondiente; obtener la mayor cantidad de información posible sobre el sistema de TI que hay que peritar; elegir los miembros del equipo (incluir especialistas externos si es necesario); asignar tareas individuales para los miembros del equipo; informar a los miembros del equipo acerca de cómo realizar sus tareas, y proporcionar las herramientas y equipos adecuados y necesarios (logística).

Si se sabe o se cree que las pruebas digitales pueden ser encontradas en la escena, el equipo responsable del secuestro de elementos de debe incluir a los miembros especialmente entrenados para las tareas de búsqueda e incautación de los equipos informáticos y de evidencias digitales. En algunos casos puede incluso ser necesario consultar a un especialista independiente. Por ejemplo, si el sistema es administrado por una empresa externa o administrador, puede considerarse la participación de esta persona como testigo experto, siempre y cuando no sea el sospechado. Los avances en la tecnología o el ambiente de tecnología relacionado a las causas judiciales locales pueden obligar a cambiar las herramientas y equipos necesarios para la recolección de evidencias. Un equipo o conjunto básico es de sumo valor durante la búsqueda y el secuestro y debe estar disponible en el lugar de trabajo. Así se debe contar con: a) Herramientas para el armado/desarmado de componentes informáticos; b) Elementos para la documentación e identificación de la evidencia; c) Elementos para el empaque y transporte de la evidencia; y d) Elementos para la comunicación con el equipo técnico-jurídico que participa de la incautación

En la escena

La persona encargada del operativo o búsqueda debe garantizar la seguridad de todas las personas en el lugar y la integridad de todas las pruebas, tanto tradicionales como electrónicos/digitales. Debe tenerse en cuenta que las posibles pruebas en computadoras y otros dispositivos electrónicos pueden ser fácilmente alterados, borrados o destruidos.
Se proponen para esta etapa los siguientes pasos: a)seguir la política de competencia para fijar la escena, en general las trazadas por la especialidad de Criminalística; b) sacar todas las personas fuera de la zona inmediata donde se encuentra la evidencia se debe recogida; c) asegurar todos los dispositivos electrónicos, incluyendo los dispositivos personales y portátiles; d) rechazar ofertas de ayuda o asistencia técnica de cualquier persona no autorizada; e)se debe dejar un computador o dispositivo electrónico apagado si ya estuviera apagado; f)si un equipo está encendido o el estado no se puede determinar, evaluar la situación si es conveniente recuperar "datos en vivo" del mismo, sacar imágenes de la pantalla actual, u otra técnica que el perito decida; g) se debe proteger los datos volátiles física y electrónicamente por todos los medios posibles; h) identificar y documentar los componentes electrónicos relacionados que no serán recogidos; y otras cuestiones relativas a la búsqueda de pistas no necesariamente digitales y al resguardo de la prueba y de la escena del crimen.

Documentar la escena

Es un proceso continuo a lo largo de todo el procedimiento de secuestro. Es muy importante documentar con precisión la ubicación y el estado de los equipos, medios de almacenamiento, otros dispositivos electrónicos y convencionales.
En general, se debe documentar lo siguiente durante la recopilación de evidencias: a) Fotografiar o registrar con video, documentando la escena completa en 360 grados de cobertura, si es posible, y detallar los sistemas informáticos y los componentes, dispositivos y equipos electrónicos; b) Documentar con detalles todos los equipos de

almacenamiento digital que se encuentran, por ejemplo, marca, modelo, número de serie, etc.; c) Estado y la ubicación de cada sistema informático que contiene o puede contener pruebas digitales, incluyendo el estado de energía del computador o dispositivo, si se encuentra encendido, apagado o en modo de hibernación o suspensión; entre otras.

Recolección de la evidencia

Un sistema informático no deberá utilizarse como evidencia sólo porque se encuentran en el lugar. La recolección de los elementos debe ser justificada por autoridad competente, por lo que la autoridad que ordenó la búsqueda debe tomar una decisión consciente de que si un artículo está siendo recolectado por el perito, debe tener una sospecha razonable o evidencia suficiente que justifique el hecho. La evidencia electrónica, al igual que con cualquier otra evidencia, debe manejarse con cuidado y de manera que conserve su valor probatorio. Esto se refiere no sólo a la integridad física de un elemento o dispositivo, pero también a la electrónica de datos que contiene. Ciertos tipos de pruebas electrónicas por lo tanto requieren cuidados especiales para su recolección, embalaje y transporte. La evidencia digital es susceptible a daños o alteración de los campos electromagnéticos como los generados por la electricidad estática, los imanes, transmisores de radio y otros dispositivos por lo que deben ser protegidos adecuadamente. La recuperación de la evidencia no electrónica o pruebas convencionales, también puede ser crucial en el investigación de los delitos digitales/informáticos. Toda la evidencia debe ser identificado, asegurada y conservada dando cumplimiento de normas y procedimientos de documentación como registro de identificación, cadena de custodia, registro de actuario de Escribano y otras basadas en las leyes aplicables de cada jurisdicción.

Embalaje, transporte y almacenaje:

Las computadoras y los dispositivos de almacenamiento digital son instrumentos electrónicos frágiles, que son sensibles a la temperatura, humedad, golpes, electricidad estática, fuentes magnéticas, e incluso a algunas acciones como el encendido y apagado. Por lo tanto, se deben tomar precauciones especiales cuando se envasan, transportan y almacenan las pruebas. En general, todos los componentes del equipo digital y medios de almacenamiento deben ser manejados con sumo cuidado ya que un manejo inadecuado puede causar daños o la destrucción de pruebas electrónicas.

4. CONCLUSIONES

La propuesta de desarrollar un Centro de Servicios de Informática Forense no se agota en estos escritos, todo lo contrario, se ha formulado una guía de trabajo que deberá ser validada pragmáticamente en un proceso de implementación gradual y dirigido. La planificación estratégica incluye una etapa de evaluación, que tiene como fin último saber si las estrategias propuestas funcionaron adecuadamente. Todas las estrategias están sujetas a modificaciones futuras porque los factores externos e internos cambian

constantemente, de manera que sería necesario realizar la revisión de los factores externos e internos en que se basan las estrategias actuales, hacer mediciones del rendimiento, y tomar medidas correctivas. Es importante también incorporar una visión multidisciplinar, con la asistencia de la Justicia y la Criminalística para que sus aportes enriquezcan y permitan ajustar el modelo a la realidad técnico-jurídica necesaria. La Informática Forense no actúa solamente en el ámbito de la Justicia, resulta necesario considerar la perspectiva de difundir y consolidar la disciplina en el área empresarial y social de nuestro medio.

References

1. David, Fred R, Conceptos de Administración Estratégica, PEARSON EDUCACIÓN, México, 2003
2. Instituto Nacional de Estadística e Informática (INEI), Guía Teórica-Práctica para la formulación de Proyectos Tecnológicos, Perú, 2002, vigente al [www.ongei.gob.pe/publica/metodologias/5162.pdf] el 14/05/2013.
3. Orión Aramayo et alt., MANUAL DE PLANIFICACIÓN ESTRATÉGICA, Instituto de la Comunicación e Imagen de la Universidad de Chile, Chile, vigente al [guiametodologica.dbe.uchile.cl/.../planificacion_estrategica.pd] el 31/03/2013.
4. Iván Silva Lira, Metodología para la elaboración de estrategias de desarrollo local, Instituto Latinoamericano y del Caribe de Planificación Económica y Social (ILPES), Chile, 2003, consultado en [www.eclac.org/publicaciones/xml/7/13867/sgp42.PD], vigente al 14/05/2013.
5. Castillo, Rafael et alt., Concientización en la Seguridad Informática, Universidad de Los Andes, Colombia, consultado en [www.criptored.upm.es/guiateoria/gt_m142r.htm] vigente el 10/05/2013.
6. Richard Brian Adams, 2012 The Advanced Data Acquisition Model (ADAM): A Process Model for Digital Forensic Practice, This thesis is presented for the degree of Doctor of Philosophy of Murdoch University
7. Data Protection and Cybercrime Division, 2013, Electronic evidence guide, A basic guide for police officers, prosecutors and judges, Version 1.0, Cyber Crime @IPA, www.coe.int/cybercrime, Council of Europe Strasbourg, France, 18 March 2013.
8. U.S. Department of Justice, Office of Justice Programs, 2008, Electronic Crime Scene Investigation: A Guide for First Responders, Second Edition,www.ojp.usdoj.gov/nij
9. Peter Sommer, 2013, Digital Evidence,Digital Investigations and E-Disclosure: A Guide to Forensic Readiness for Organisations, Security Advisers and Lawyers Information Assurance Advisory Council (IAAC), United Kingdom.
10. IACP Technology Policy Framework,2014,IACP Divisions, Committees, Sections, the IA CP National Law Enforcement Policy Center, and others, [http://www.theiacp.org/Portals] vigente al 10/03/2014.
11. SWGDE Model Standard Operation Procedures for Computer Forensics, 2012, Scientific Working Group on Digital Evidence, Version: 3.0
12. Good Practice Guide for Computer-Based Electronic Evidence, 2007,published by 7Safe, www.7safe.com/electronic_evidence , Association of Chief Police Officers (ACPO).
13. Best Practices For Seizing Electronic Evidence, 2013, v.3, A Pocket Guide for First Responders, U.S. Department of Homeland Security United States Secret Service.

Automated Analysis of Source Code Patches using Machine Learning Algorithms

Antonio Castro Lechtaler[1,2], Julio César Liporace[1], Marcelo Cipriano[1], Edith García[1], Ariel Maiorano[1], Eduardo Malvacio[1], Néstor Tapia[1]

[1] Grupo de Investigación en Criptografía y Seguridad Informática (GICSI), Instituto Universitario del Ejército (IUE), [2] Universidad Nacional de Chilecito (UNdeC), Argentina

{antonio.castrolechtaler, edithxgarcia, jcliporace,maiorano, cipriano1.618, edumalvacio, tapianestor87}@gmail.com

Abstract. An updated version of a tool for automated analysis of source code patches and branch differences is presented. The upgrade involves the use of machine learning techniques on source code, comments, and messages. It aims to help analysts, code reviewers, or auditors perform repetitive tasks continuously. The environment designed encourages collaborative work. It systematizes certain tasks pertaining to reviewing or auditing processes. Currently, the scope of the automated test is limited. Current work aims to increase the volume of source code analyzed per time unit, letting users focus on alerts automatically generated. The tool is distributed as open source software. This work also aims to provide arguments in support of the use of this type of tool. A brief overview of security problems in open source software is presented. It is argued that these problems were or may have been discovered reviewing patches and branch differences, released before the vulnerability was disclosed.

Keywords: automated, source code review, source code analysis, patch analysis, machine learning, text mining, software quality.

1 Introduction

This work presents a software tool for automated source code patches and branch differences analysis. It aims to systematize updates and source code reviews to alert on potential bugs, implying some degree of system security compromise, or vulnerabilities. The tool is distributed as an open source project and available at http://github.com/gicsi/aap. Currently, we have not found other open source tools available for the proposed functionality presented here. Other projects have been published but they largely deal with software engineering. For instance: trackable recoveries between source code and corrected bugs through patch analysis [5], or the use of patches for bug reporting. In the first case, a data processing tool was presented for Bugzilla systems, identifying the information with CVS tags.

In the case of open source analysis with cryptographic functionalities, Android case studies show that, for instance, only 17% of 269 vulnerabilities - reported in the period between January 2011 and May 2014 - were attributed to glitches in the

cryptographic libraries. The remaining 83% were attributed to flaws in the application´s use of these libraries [14].

Furthermore, Android applications were analyzed in [8]: 10.327 out of 11.748 applications (88%) reported at least one implementation error. This last work presented an automatic detection tool, but was not distributed freely [15].

Other tools are available for general use to assist in source code management. However, they do not focus on security problems or patch reviews . They operate over the entire development cycle. Among the most common available open source alternatives, [12] and [23] are worth mentioning.

In the next section, references are surveyed on open and commercial software alternatives, dealing with security issues [4, 19, and 22].

It should be noted that control systems aiming at specific revisions have their own difficulties and limitations.

A study analyzing 210 samples showed that over 40% were not reported in the evaluation of C/C++ source code with five analytical tools; while only 7% of those samples were appropriately reported by the five tools [30]. Analogous results were found in the same study, using six tools based on Java.

The Open project OWASP [22] has identified the weaknesses of automated tests. It has also pointed out their strengths: ability to perform fast and repetitive analysis over a high volume of source code, the capacity to detect high probability errors, and the specificity provided in the reports.

2 Source Code Analysis

2.1 The Cost of Software Quality

Source code analysis and inspections are a continuous best-practice, aiming to improve software quality. It should be noted that although the quantity of bugs in a software project is not the only quality indicator, it constitutes a valuable measure for control and potential enhancements to development processes. Strictly, the goal of these reviews is to reduce the cost of software quality by identifying and correcting bugs at early stages of development [25].

It has been shown empirically [9] that software quality depends on control mechanisms used as an integral part of processes. Furthermore, the rate of personal inspections (measured in quantity of source code lines per time unit) has been shown to affect the effectiveness in detection (and correction) of bugs. Data shows, for instance, that review quality declines as this rate exceeds its recommended maximum of 200 code lines per hour. The authors cite previous articles in which a rate of 125 lines per hour is considered the optimum [3]. They also note: *"it is almost one of the laws of nature about inspections, i.e., the faster an inspection, the fewer defects removed"* [25].

2.2 Analysis Systematization

When dealing with software quality and the effectiveness of source code review, analysis systematization may yield efficient quality and development control

processes when accounting for the time demanded from an analyst. Quality assurance tools are an essential resource in the improvement of software applications.

In a NIST publication, the minimum required functionality specifications in a source code analyzer for searching vulnerabilities are laid out [2]. The article suggests that it should identify security weaknesses, and report them tagging their type and location.

According to SEI/CERT, relying exclusively on policies, standards, and good practices for software development have proven inadequate [30]. It points out that these factors are not being applied consistently and that manual source code audits can be complemented and enhanced with systematized and automated tests.

2.2.1 Current Available Tests

Different organizations maintain lists from which a great range of tools for testing source code are referenced. Among them: the NIST webpage, Source Code Security Analyzers [19], and the Secure Coding Tools from CERT [4].

Finally, the OWASP project is an obliged reference, maintaining a thorough list of source code analysis tools [22].

3 Post Patch Vulnerabilities

3.1 Errors in the Categorization of Vulnerabilities

In his recent thesis [34] and revising previous work [35, 36, 33], Jason Wright - best known for his work in the cryptographic framework of the OpenBSD operating system [13] – emphasizes the importance of patch reviews. In the third section of his last publication, he introduces the concept of *hidden impact bugs*. The concept is also found in [35] and [33]. In the latter the term used is *hidden impact vulnerabilities*, referring to previous work [1]. However, they all deal with the same concept; i.e., vulnerabilities which were reported as bugs and whose real impact – their vulnerability category – was recognized after issuing the initial report which included a patch to solve what was believed to be a software flaw or defect with no security implications.

Impact delay is defined as the time elapsed since the report of the flaw – as a patch – until the time when the bug was tagged with a CVE. In 2012, the author, contributing with others at a second stage of analysis, published a review of hidden impact bugs in the source code (patches) of the Linux Kernel and MySQL database [33]. Later, in 2013, they published an extended analysis, focusing on MySQL exclusively [35].

The tests consisted of detailed reviews of a subset of the reported bugs. The source code associated to the bugs is analyzed to determine the percentage of unclassified vulnerabilities. The authors extrapolate the results to yield an estimate of the total percentage on reported flaws. From the available data, the results are summed up as follows:

- In the second stage analysis of the Linux kernel - from January 2009 to April 2011 - results show that from a total of 185 hidden impact bugs, 73 (39%) had

an impact delay of less than 2 weeks; 55 bugs (30%) had a delay of less than 4 weeks; and 29 (16%) had a delay of less than 8 weeks.

- In the MySQL study, total hidden impact bugs sum up to 29. From them, 19 (65%) had an impact delay of less than 2 weeks, also 19 had a delay of less than 4 weeks, and 16 bugs (55%) had a delay of less than 8 weeks.

3.2 Fixed Issues in NTP-devel

A reference implementation of the protocol used to synchronize clocks over the network, known as *Network Time Protcol* (NTP), is distributed as part of the Linux operating systems, among others. In December 2014, a series of security issues – some of them categorized as critical – were identified. The information went public along with patches to fix the problems. However, the bug database used by the development team showed that some of these bugs had already been fixed in the *devel* version, years before.

Additional information is available in their security webpage [16]. A brief outline of the particular details is given below to illustrate the way in which a comparison between two versions can point out critical fixes not implemented in the stable version.

Although Bugzilla´s historical data show that the information was made accessible to the public on 12/12/2014, messages on bug 2665 [17] - regarding a weak cryptographic default key - point out that the vulnerability had already been "corrected" four years before, in the NTP-devel (4.2.7), particularly "(4.2.7p11) 2010/01/28."

An additional example involves bug 2666 [18] dealing with a random number generator – cryptographically insecure with a weak initialization seed. Although it was later fixed once again before the release of version 4.2.8, messages point out that the development version n 4.2.7 p30 (of November 1st, 2011) had already corrected the problem, referring an enhancement of the seed management.

3.3 Multiple Patches for Shellshok Vulnerabilities

Recently discovered in the UNIX bash shell system, the Shellshok vulnerabilities represent an example of critical security problems which are not corrected completely or adequately in the first published patches. The importance and criticality of the vulnerability was highlighted in The Register [28], in which they recommended immediate installation of the patch to avoid wide open access to Linux and OS X operating systems. The news, along with a public notice from Red Hat [26], and the updated packages are dated September 24th, 2014. Nonetheless, later, related problems were identified and for which Red Hat did not offer updates until September 26th, 2014 [27].

3.4 Simultaneous Discovery of the Heartbleed Vulnerability

During 2014, a critical vulnerability, known as Heartbleed, was disclosed. It consisted of an exploitable bug in the OpenSSL library. According to Bruce Schneier,

half a million sites were infected, turning the problem catastrophic [29]. A Red Hat Security representative [6] considered that the coincidence of two simultaneous findings of a single problem would increase the risk of maintaining the vulnerability secret. Consequently, it can be inferred that this rationale could have encouraged OpenSSL to release updated packets hastily. The rush in the releases may be responsible for the lack of coordination in issuing the patches. Thus, this illustrates how, under certain circumstances, several organizations can publish patches in an non-coordinated manner.

3.5 Vulnerability in OpenBSD after a Misclassified Bug

In 2007, an IPv6 vulnerability was found in the OpenBSD kernel [21]. A specially crafted ICMP packet could generate a remote buffer overflow, when handled by an internal function which miscalculated the buffer´s length. According to the author, the vulnerability was identified while analyzing – and trying to reproduce – a problem already fixed by a patch rated as reliability fix instead of vulnerability or security fix [20].

4 Patch Analysis Tool

4.1 Automated Analysis of Patches, or AAP, version 0.2b

4.1.1 Idea and Rationale

Beside obvious differences in source code and binary analysis, the project is based on the criteria used by automated malware analysis tools, such as the Cuckoo Sandbox.

In [11], two problems are identified when performing this type of analysis manually: the increasing volume of malware in the wild and the time required for a thorough inspection.

The goal in this work also involves an efficacy assessment in the use of IA techniques, especially text mining and machine learning, applied to source code patches and “texts” in general comments.

The rationale relies on published work regarding the discovery of vulnerabilities in the patches distribution of open source code projects. The references in the third section of this article constitute examples of this problem.

4.1.2 Machine Learning

Although beyond the scope of this article, a brief overview is presented here on the techniques and applications of machine learning implemented in the tool presented in this article:

Natural Language Processing (NLP). Natural language is understood as the language used by human beings to communicate: English, Spanish, ... Generally, NLP involves the manipulation of a natural language through the use of computer systems. For instance, text analysis enables the detection of positive or negative sentiments in tweets – in favor or against.

Supervised Classification. Classifying comprises assigning tags of the appropriate category to particular entries or inputs. The set of tags is defined beforehand. Some classification examples are SPAM identification or news categorization (sports, finance, and others). A classifier is supervised when it is trained based on data (corpora) containing the correct tag for each entry or input.

AAP Implementation. For the generation of tagged data, bug reports, with attached patches, were downloaded from Red Hat and OpenBSD public bug database system and errata pages, respectively. The labels "vulnerability" and "bug" were defined according to the criticality established in the original reports. With this set of tagged entries or inputs, classifiers - of the naïve Bayes and decision tree types - were trained. These classifiers can then be used from a plugin to estimate the appropriate tag for each automatically analyzed patch.

4.1.3 Functionality Currently Implemented

Through its web interface, the tool maintains and organize information of software projects, reference to its repositories, branches, updates, and analysts users assigned to them.

In addition, the web interface handles the general configuration for the automatic reviews of patches and different versions. These rules are implemented in the form of plugins, programmed in Python and editable on the interface itself.

The verifications of these rules, automatically executed over patches or over branch differences of a software project update, activates alerts whenever necessary. Currently, these rules include, among others, searching for configured patterns in source code and comments, detecting source code implementing cryptographic functionality, and checking for changes in specific files or done by particular authors or committers.

Since version 0.2b, the tool contemplates the application of machine learning techniques through trained classifiers with patches and comments from public bug information repositories. Classification analysis applies a per "vulnerability" or "bug" criteria.

Configured rule inspections are performed over patches recovered from GIT repositories [10].

The periodic batch process updates the repositories and automatically performs the analysis configured for each of its branches. The outcome includes the generation of system alerts -and optionally sending notification emails with analysis results- to the assigned analysts.

4.1.4 Technologies

The tool consists of a web-based application developed in the Python language [24] and using Web Django framework [7]. It makes use of GIT versatility for updating and handling source codes [10]. Other external tools may be used optionally.

Currently, plugins to classify patches and comments are implemented using an open source library or toolkit, also developed in Python: NLTK (Natural Language

Toolkit). It includes software, data and documentation, and can be freely downloaded from http://nltk.org/.

NLTK Trainer constitutes another resource – a set of Python scripts for NLTK model training.

Particular plugins for static analysis were implemented for different languages (C/C++, Python, Perl, PHP) based on the use of open source tools like FlawFinder, RATS, Splint and CppCheck.

4.1.5 Open Source Software

The tool's source code is publicly published using a free service for open source projects management: http://github.com/gicsi/aap. It is licensed under the terms of the GPL, or GNU General Public License, from the Free Software Foundation.

5 Further Work

Functionality enhancements are under design. The expectation is to capture the interest of other developers and encourage them to get involved with the project. Among other improvements, the following features are under consideration: to achieve a higher degree of sophistication in rule and algorithm design, to extend the use of machine learning in the tool, to incorporate information retrieved from developers´ discussion forums, and to integrate the tool with other databases dealing with vulnerability reports and security advisories.

References

1. Arnold, J., Abbott, T., Daher, W., Price, G., Elhage, N., Thomas, G.A. Kaseorg Security Impact Ratings Considered Harmful. Proceedings 12th Conference on Hot Topics in Operating Systems. USENIX. May 2009. [Online – accessed 29/12/2014: http://www.inl.gov/technicalpublications/Documents/5588153.pdf].
2. Black, P., Kass, M., Koo, M., Fong, M. Source Code Security Analysis Tool Functional Specification Version 1.1. NIST Special Publication 500-268 v1.1. February 2011. [Online – accessed 29/12/2014: http://samate.nist.gov/docs/source_code_security_analysis_spec_SP500-268_v1.1.pdf].
3. Buck, F. Indicators of Quality Inspections. IBM Technical Report TR21.802, Systems Comm. December 1981.
4. CERT Division - Secure Coding Secure Coding Tools. CERT, Software Engineering Institute (SEI), Carnegie Mellon University. [Online - accessed 29/12/2014: http://www.cert.org/secure-coding/tools/index.cfm].
5. Corley, C., Etzkorn, L., Kraft, N., Lukins, S. Recovering Traceability Links between Source Code and Fixed Bugs via Patch Analysis. University of Alabama. 2008. [online - accessed 29/12/2014: http://www.cs.wm.edu/semeru/tefse2011/papers/p31-corley.pdf].
6. Cox, M. Heartbleed. Mark Cox, J. Google+. [Online - accessed 29/12/2014: https://plus.google.com/+MarkJCox/posts/TmCbp3BhJma].
7. Django Framework. Django overview. Django Software Foundation. [Online - accessed 29/12/2014: https://www.djangoproject.com/start/overview/].
8. Egele, M., Brumley, D., Fratantonio, Y., Kruegel, C. An empirical study of cryptographic misuse in android applications. CCS '13 Proceedings of the 2013 ACM SIGSAC

conference on computer & communications security. Pages 73-84. 2013. U.S.A. [Online – accessed 29/12/2014: http://www.cs.ucsb.edu/~chris/research/doc/ccs13_cryptolint.pdf].
9. Kemerer, C., Paulk, M. The Impact of Design and Code Reviews on Soft-ware Quality: An Empirical Study Based on PSP Data. IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 35, NO. XX April 2009. [Online - accessed el 29/12/2014: http://www.pitt.edu/~ckemerer/PSP_Data.pdf].
10. Git SCM. About Git. Git - Software Freedom Conservancy. [Online: http://git-scm.com/about - accessed 2912/2014].
11. Guarnieri, C. One Flew Over the Cuckoo's Nest. Hack in the Box 2012. May 2012. Netherlands. [Online – accessed 29/12/2014: http://sebug.net/paper/Meeting-Documents/hitbsecconf2012ams/D1T1%20-%20Claudio%20Guarnieri%20-%20One%20Flew%20Over%20the%20Cuckoos%20Nest.pdf].
12. Gerrit Website Gerrit Code Review. Google Inc. [online: https://code.google.com/p/gerrit/ - accessed 29/12/2014].
13. Keromytis, A., Wright, J., de, T. Raadt. The Design of the Open BSD Cryptographic Framework. International Conference on Human System Interactions (HSI). June 2012. Australia. [Online - accessed 29/12/2014: http://www.thought.net/papers/ocf.pdf].
14. Lazar, D., Chen, H., Wang, X., Zeldovich, N. Why does cryptographic software fail? a case study and open problems. Proceedings of 5th Asia-Pacific Workshop on Systems Article No. 7. 2014. U.S.A. [online - accessed 29/12/2014: http://pdos.csail.mit.edu/papers/cryptobugs:apsys14.pdf].
15. Mujic, A. Reimplementation of CryptoLint tool. Blog for and by my students. December 2013. [Online - accessed 29/12/2014: http://sgros-students.blogspot.com.ar/2013/12/reimplementation-of-cryptolint-tool.html].
16. Network Time Protocol project. NTP Security Notice. NTP support website. Network Time Foundation. [Online - accessed 29/12/2014: http://support.ntp.org/bin/view/Main/WebHome].
17. Network Time Protocol project. Bug 2665 - Weak default key. NTP Bugzilla. Network Time Foundation. [Online - accessed 29/12/2014: http://bugs.ntp.org/show_bug.cgi?id=2665].
18. Network Time Protocol project. Bug 2666 - non-cryptographic random number generator with weak seed. NTP Bugzilla. Network Time Foundation. [Online - accessed 29/12/2014: http://bugs.ntp.org/show_bug.cgi?id=2666].
19. NIST Source Code Security Analyzers. SAMATE - NIST. [Online - accessed 29/12/2014: http://samate.nist.gov/index.php/Source_Code_Security_Analyzers.html].
20. Ortega, A. OpenBSD Remote Exploit. Core Security. Julio de 2007. [Online – accessed 29/12/2014: https://www.blackhat.com/presentations/bh-usa-07/Ortega/Whitepaper/bh-usa-07-ortega-WP.pdf].
21. Ortega, A., Richarte, G. OpenBSD Remote Exploit. Core Security. April 2007. [Online - accessed 29/12/2014: https://www.blackhat.com/presentations/bh-usa-07/Ortega/Whitepaper/bh-usa-07-ortega-WP.pdf].
22. OWASP Wiki. Source Code Analysis Tools. The Open Web Application Security Project (OWASP). Ult. mod. 29/10/2014. [Online: https://www.owasp.org/index.php/Source_Code_Analysis_Tools - accessed 29/12/2014].
23. Phabricator Website. Phabricator, an open source, software engineering platform. Phacility, Inc. [online: http://phabricator.org/ - accessed 29/12/2014].
24. Python Website. About Python. Python Software Foundation. [Online: https://www.python.org/about/ - accessed 29/12/2014].
25. Radice, R. High Quality Low Cost Software Inspections. Paradoxicon Publishing. 2002.
26. Red Hat. Security. CVE Databases. CVE-2014-6271. Red Hat Customer portal. September 24th, 2014. [Online: https://access.redhat.com/security/cve/CVE-2014-6271 - accessed 29/12/2014].

27. Red Hat. Security. CVE Databases. CVE-2014-7169. Red Hat Customer portal. September 24th, 2014. [Online: https://access.redhat.com/security/cve/CVE-2014-7169 - accessed 29/12/2014].
28. The Register. Leyden, J. Patch Bash NOW: 'Shellshock' bug blasts OSX, Linux systems wide open. The Register online tech publication. September 24th, 2014. [Online: http://www.theregister.co.uk/2014/09/24/bash_shell_vuln/ - accessed 29/12/2014].
29. Schneier, B. Heartbleed. Schneier on Security, Blog. April 2014. [Online: https://www.schneier.com/blog/archives/2014/04/heartbleed.html – accessed 29/12/2014].
30. Seacord, R., Dormann, W., McCurley, J., Miller, P., Stoddard, R., D.Svoboda, Welch, J. Source Code Analysis Laboratory (SCALe). CERT, Software Engineering Institute (SEI), Carnegie Mellon University. April 2012. [Online: https://resources.sei.cmu.edu/asset_files/TechnicalNote/2012_004_001_15440.pdf - accessed 29/12/2014].
31. W. Weimer. Patches as Better Bug Reports. University of Virginia. 2006. [Online: https://www.cs.virginia.edu/~weimer/p/p181-weimer.pdf - accessed 29/12/2014].
32. Wheeler, D. Flawfinder. David Wheeler, A.'s Personal Home Page- Flawfinder Home Page. [Online: http://www.dwheeler.com/flawfinder/ - accessed 29/12/2014].
33. Wijayasekara, D., Manic, M., Wright, J., McQueen, M. Mining Bug Databases for Unidentified Software Vulnerabilities. Proceedings International Conference on Human System Interactions (HSI). June 2012, Perth, Australia. [Online: http://www.inl.gov/technicalpublications/Documents/5588153.pdf – accessed 29/12/2014].
34. Wright, J. Software Vulnerabilities: Lifespans, Metrics, and Case Study. Master of Science Thesis. University of Idaho. May 2014. [Online: http://www.thought.net/papers/thesis.pdf – accessed 29/12/2014].
35. Wright, J., Larsen, J., McQueen, M. Estimating Software Vulnerabilities: A Case Study Based on the Misclassification of Bugs in MySQL Server. Proceedings International Conference of Availability, Reliability, and Security (ARES). September 2013. pp. 72-81. Regensburg, Germany. [Online – accessed 29/12/2014: http://www.inl.gov/technicalpublications/Documents/5842499.pdf].
36. Wright, J., McQueen, M., Wellman, L. Analyses of Two End-User Software Vulnerability Exposure Metrics (Extended Version). Information Security Technical Report, 17(4), Elsevier. April 2013. pp. 44-55. [Online: http://www.thought.net/papers/INL-JOU-12-27465-preprint.pdf – accessed 29/12/2014].

El tratamiento de la evidencia digital y las normas ISO/IEC 27037:2012

Santiago Roatta[1], María Eugenia Casco[2], Martín Fogliato[3],

[1] Facultad Tecnología Informática / Universidad Abierta Interamericana
Ovidio Lagos 934, Rosario, 0341-4356510
santiago.roatta@gmail.com

[2] Dirección de Inteligencia Criminal Estratégica de la Dirección General de Policía de Investigaciones / Ministerio de Seguridad de la Prov. De Santa Fe
Primera Junta 2823, 0342-4505100
mecasco@gmail.com

[3] Facultad de Derecho / Universidad Nacional de Rosario
Córdoba 2020, Rosario, 0341-4802634
gfogliato@netcoop.com.ar

Resumen. La evidencia digital bien procesada puede aprovecharse al máximo en distintos escenarios. En cada uno de ellos existe una orientación diferente respecto de lo que se pretende obtener: calidad probatoria, precisión en el análisis, restauración del servicio y/o el costo de la recolección de la evidencia. Los componentes clave que proporcionan credibilidad en la investigación son la metodología aplicada durante el proceso y la calificación de los individuos que intervienen en el desarrollo de las tareas especificadas en la metodología. Este trabajo presenta pautas para el manejo de la evidencia digital; sistematizando la identificación, adquisición, análisis y preservación de la misma. Estos procesos están diseñados para mantener la integridad de la evidencia, con una metodología aceptable para contribuir a su admisibilidad en procesos legales y en sintonía con las normas ISO/IEC 27037:2012 [1]. En concordancia con la ley provincial 13139 [2] (que tiene por objeto migrar el software propietario que utiliza la provincia de Santa Fe hacia software libre), un objetivo adicional es no solo generar nuevas herramientas y protocolos de análisis digital forense sino también evaluar aquellos ya existentes. Finalmente, se muestran los resultados del análisis y la valoración de algunas herramientas GNU disponibles.

Palabras Clave: Informática forense, Normas ISO/IEC 27037:2012.

1 Los orígenes de la informática forense

El análisis digital forense es la aplicación de técnicas científicas y analíticas especializadas que permiten identificar, preservar, analizar y presentar datos que sean válidos dentro de un proceso legal. Es una disciplina que comienza con los orígenes mismos de la electrónica digital pero se ha desarrollado de manera diferente. Mientras el hardware digital y la informática han tenido un desarrollo y difusión tan extendido, que son un acabado ejemplo de la llamado globalización; la informática forense tiene un modelo actuación propio según las leyes de cada país.

El campo de la informática forense se inició a fines de los años 70, poco después de que las computadoras personales se convirtieran en una opción viable para los consumidores.

En 1978, el estado de Florida reconoce los crímenes de sistemas informáticos en el Computer Crimes Act, en casos de sabotaje, copyright, modificación de datos y ataques similares. Nace Copy II PC de Central Point Software en 1981. También es conocida como copy2pc, que se usaba para la copia exacta de disquetes, que generalmente estaban protegidos para evitar copias piratas. El producto es posteriormente integrado en las Pc Tools. La compañía es un éxito y es comprada por Symantec en 1994. En 1982 Peter Norton publica UnErase: Norton Utilities 1.0, la primera versión del conjunto de herramientas Norton Utilities, entre las que destacan UnErase, una aplicación que permite recuperar archivos borrados accidentalmente. Otras aplicaciones también serán útiles desde la perspectiva forense, como FileFix o TimeMark. Con el éxito de la suite de aplicaciones, Norton publica varios libros técnicos, como Inside the I.B.M. Personal Computer: Access to Advanced Features and Programming, del que su octava edición se publicó en 1999, 11 años después de la primera edición. En 1984 el FBI forma el Magnetic Media Program, que más tarde, en 1991, será el Computer Analysis and Response Team (CART). En 1986 Clifford Stoll colabora en la detección del hacker Markus Hess. En 1988 publica el documento Stalking the Wily Hacker contando lo ocurrido. Este documento es transformado en 1989 en un libro, anticipando una metodología forense. En 1987 nace la compañía AccessData, pionera en el desarrollo de productos orientados a la recuperación de contraseñas y el análisis forense con herramientas como la actual Forensic Toolkit (FTK). En 1988 se crea la International Association of Computer Investigative Specialists (IACIS), que certificará a profesionales de agencias gubernamentales en el Certified Forensic Computer Examiner (CFCE), una de las certificaciones más prestigiosas en el ámbito forense. En este mismo año se desarrolla el programa Seized Computer Evidence Recovery Specialists o SCERS, con el objetivo de formar a profesionales en computer forensics. El libro A forensic methodology for countering computer crime, de P. A. Collier y B. J. Spaul acuña en 1992 el término computer forensics. Otros libros posteriores continuarán desarrollando el termino y la metodología, como: High-Technology Crime: Investigating Cases Involving Computers de Kenneth S. Rosenblatt. En 1995 se funda el International Organization on Computer Evidence (IOCE), con objetivo de ser punto de encuentro entre especialistas en la evidencia electrónica y el intercambio de información. A partir de 1996 la Interpol organiza los International Forensic Science Symposium, como foro para debatir los avances forenses, uniendo fuerzas y conocimientos. En agosto de

2001 nace la Digital Forensic Research Workshop (DFRWS), un nuevo grupo de debate y discusión internacional para compartir información.

2 Panorama actual

El equipo participante en el proyecto tiene una gran experiencia sobre las necesidades, limitaciones y dificultades materiales que surgen a la hora de validar los resultados periciales de un sistema electrónico para formar parte en un proceso legal. A partir de este conocimiento, se está investigando acerca de los desarrollos preexistentes, el estado general del arte y experiencias anteriores. Mediante benchmarking y análisis teóricos se han validado algunas herramientas y se han desechado otras. Se ha generando un protocolo de actuación para profesionales de los equipos policiales que intervengan en la recolección de evidencia digital, para lo cual se ha tomado como punto de partida la experiencia de la provincia de Neuquén [3] y [4].
Hasta el año 2012, dos instituciones eran consideradas referencias ineludibles para el análisis forense, el National Institute of Standart and Technology [5] (NIST), y el Federal Bureau of Investigation [6] (FBI). Actualmente, la referencia es la norma de alcance global ISO/IEC 27037:2012. Bajo esta norma está realizado todo nuestro trabajo.
La citada norma proporciona pautas para el manejo de la evidencia digital; sistematizando la identificación, recolección, adquisición y preservación de la misma. Estos procesos deben diseñarse para mantener la integridad de la evidencia y con una metodología aceptable para contribuir a su admisibilidad en procesos legales. De acuerdo con la ISO/IEC 27037:2012 la evidencia digital es gobernada por tres principios fundamentales: la relevancia, la confiabilidad y la suficiencia. Estos tres elementos definen la formalidad de cualquier investigación basada en evidencia digital. Esta norma internacional también proporciona directrices generales para la obtención de pruebas no digitales que pueden ser útiles en la etapa de análisis de la evidencia digital. La norma pretende orientar a aquellos responsables de la identificación, recolección, adquisición y preservación de potencial evidencia digital. Estos individuos incluyen: Digital Evidence First Responder (DEFRs) o especialista en evidencia digital de primera intervención, Digital Evidence Specialist (DESs) o especialista en evidencia digital, especialistas en respuestas a incidentes y directores de laboratorios forenses. De esta manera se asegura que las personas responsables de gestionar potencial evidencia digital lo hagan con prácticas aceptadas en todo el mundo, con el objetivo de realizar la investigación de una manera sistemática e imparcial, preservando su integridad y autenticidad. La norma también está destinada a aquellas personas que necesitan determinar la confiabilidad de la evidencia digital que se les presenta. Es aplicable a las organizaciones que necesitan formalmente establecer un marco de aceptabilidad. La evidencia digital a la que hace referencia puede obtenerse de diferentes tipos de dispositivos digitales, redes, bases de datos, etc. Se refiere a datos que ya están en formato digital y no abarca la conversión de datos analógicos a formato digital. La rigurosidad de la aplicación de una metodología adecuada se debe a la fragilidad de la evidencia digital. Como la norma no impone el

uso de herramientas o métodos particulares nosotros hemos utilizado exclusivamente software libre. Es importante destacar la pertinencia, importancia y ventajas que implicaría el hecho de trabajar con software libre en el ámbito de la informática forense de la justicia de la provincia de Santa Fe. La más relevante de ellas y tal vez la menos conocida es el principio de Kerckhoffs [7]; la efectividad del sistema no debe depender de que su diseño permanezca en secreto. El código fuente abierto permite auditar los programas impidiendo puertas traseras y disminuye el tiempo de reacción ante bugs. El diseñador se siente naturalmente obligado a programar de manera "académica y elegante" y una ventaja no menor es que gran parte de las aplicaciones GNU disponibles son gratis.

La norma ISO/IEC 27037:2012 no aborda los procedimientos legales, procedimientos disciplinarios y otras acciones relacionadas con el inadecuado manejo de la evidencia digital. La aplicación de esta norma internacional exige el cumplimiento de las leyes, normas y reglamentos nacionales. No sustituye los requisitos legales específicos de cualquier jurisdicción. En cambio, puede servir como una guía práctica para cualquier DEFR o DES en la investigación. La norma no hace ninguna referencia a requisitos específicos de cada jurisdicción que se refieren a cuestiones como la admisibilidad, la ponderación probatoria, la pertinencia y otras limitaciones que controlan el uso de la evidencia digital en los tribunales de justicia. Sin embargo puede ayudar al intercambio de evidencia digital entre jurisdicciones. Los usuarios de esta norma deben adaptar y modificar los procedimientos descriptos en esta norma internacional de conformidad con los requisitos legales de cada jurisdicción.

Es de suma importancia el concepto de Cadena de Custodia (CoC) estableciendo los recaudos mínimos a tener en cuenta:

- Un identificador univoco de la evidencia.
- Quién, cuándo y dónde se accede a la evidencia.
- El pasaje de la evidencia de un sitio a otro y tareas realizadas.
- Todo cambio potencial en la evidencia digital debe registrarse con el nombre del responsable y la justificación de las acciones realizadas.

2 Alcance del trabajo

Nuestro trabajo se enmarca en dos proyectos radicados en diferentes unidades académicas: Análisis Digital Forense, Conceptos y aplicaciones (proyecto acreditado en la Facultad de Ingeniería de la Universidad Nacional de Rosario) e Informática Forense con herramientas de software libre (proyecto acreditado en la Facultad de Tecnología Informática de la Universidad Abierta Interamericana). La diversidad del equipo de trabajo compuesto por especialistas en derecho, software y hardware genera un enfoque colaborativo, produce una visión multidisciplinaria de la problemática y no solo comprende actividades de investigación, sino que ha tenido impacto en docencia de grado y extensión. En la asignatura electiva Arquitectura y Diseño de Computadoras de ingeniería electrónica de la Universidad Nacional de Rosario ya se han incorporado retos forenses como problemas de ingeniería para comenzar proyectos de fin de carrera en el año 2012. En la asignatura Sistemas de Hardware de ingeniería en sistemas informáticos de la Universidad Abierta Interamericana se ha

presentado la problemática del diseño de un laboratorio de informática forense. En la asignatura Seguridad Informática de la Universidad Abierta Interamericana es donde mayor impacto se espera. En 2014 y 2015 hemos compartido nuestro trabajo de investigación con dependencias públicas de la provincia de Santa Fe. Colaborando y poniendo a disposición el know how del grupo académico de investigación en el diseño y desarrollo de las políticas de seguridad informática para la Dirección de Inteligencia Criminal Estratégica de la Dirección General de Policía de Investigaciones de Santa Fe. Está prevista la realización de convenios entre las universidades y la provincia de Santa Fe a fin de capacitar personal policial y del poder judicial.

3 Un Caso Real

Para reafirmar la necesidad de formalizar la aplicación de las ISO/IEC 27037:2012 como parte del procedimiento estándar para la provincia de Santa Fe, se presentan a continuación algunas circunstancias generadas en un allanamiento efectuado durante el mes de Julio del corriente año, en la mencionada provincia, sirviendo el mismo de ejemplo práctico: para el procedimiento de intervención en el lugar del hecho se hizo indispensable la presencia de personal policial especializado y de empleados judiciales permeables a la realidad que rodea los delitos actuales donde intervienen "herramientas" tecnológicas, en un porcentaje cada vez mayor, tanto de software como hardware, indistintamente. Es dable destacar que de cada investigación surgen diferentes particularidades. En esta oportunidad nos concentramos en una intervención originada en un informe proveniente de NCMEC[8] de carácter prioritario, por reincidencia, referente a material de pornografía infantil. La lógica de estas investigaciones hace fundamental el registro de los domicilios de los eventuales imputados para asegurar la evidencia digital y consecuentemente realizar el correspondiente informe forense. Este tipo de abordaje requiere particular atención por parte de los especialistas, tanto en la preservación del entorno inmediatamente después del acceso al lugar, como durante la requisa, haciéndose sumamente valiosa la presencia de un perito fotógrafo que realice las primeras vistas fotográficas y efectúe una minuciosa filmación de los detalles que puedan resultar de utilidad durante el análisis forense posterior. En este caso, se realizó la filmación correspondiente, luego ingresó el Jefe policial a cargo de procedimiento acompañado por el Especialista de Evidencia Digital de Primera Intervención (quién contó con ayuda de dos especialistas que trabajaron en el registro y documentación de los elementos recolectados). Debido a que se hallaron numerosos documentos con inscripciones varias, el aporte de los especialistas en informática forense se hicieron indispensables, pudiendo detectar contraseñas de encriptación, códigos y procedimientos minuciosamente documentados en cuadernos, hojas sueltas, revistas, posters y otros trozos de papel almacenados en grandes cantidades en la habitación del imputado. Sólo con esta pequeña reseña queda completamente comprobado que acceder al lugar del suceso siguiendo lo aconsejado por normas y estándares representa asegurar el éxito de la investigación. En este caso en particular, gracias a los conocimientos de los Especialistas se logró individualizar material de importancia

trascendental para iniciar nuevas investigaciones originadas en la intervención realizada en esa fecha. Determinase que es conveniente aplicar los amplios conocimientos del Especialista desde el primer acceso a la evidencia y que indefectiblemente podrá hallar información válida en cualquier lugar, en cualquier tipo de soporte y su participación es de carácter prioritario.

4 Conclusiones y líneas futuras

Nuestro resultado más importante ha sido el desarrollo con la empresa local Digilogic Ingeniería [9] de un prototipo de bloqueador de escritura por hardware SATA para la adquisición de evidencia digital de discos rígidos sin contaminar la evidencia. Este hardware es inédito en Argentina y está diseñado con tecnología de lógica programable (FPGAs).
Se han implementado dos cursos en concordancia con la ISO/IEC 27037:2012: Evidence First Responder (DEFRs) o *especialista en evidencia digital de primera intervención*, y Digital Evidence Specialist (DESs) o *especialista en evidencia digital*. Nuestro objetivo es extender estos cursos con personal policial y del poder judicial.
Luego de un muy extenso benchmarking se ha elegido la distribución Deft Linux como nuestra herramienta preferida por su facilidad de instalación, la cantidad y calidad de herramientas que incluye y su actualización permanente [10]. Para la extracción de evidencia de teléfonos móviles se escogió la distribución Santoku Linux [11].
A medida que se van obteniendo resultados, obviamente se generan nuevas preguntas e ideas. La línea de trabajo que asoma como más prometedora y en la que estamos trabajando actualmente es el análisis forense de teléfonos chinos shanzhai [12], teléfonos piratas que no se pueden acceder con las herramientas comerciales más conocidas como XRY[13] y UFED CHINEX[14]

Referencias

1. Guidelines for identification, collection, acquisition and preservation of digital evidence" ISO/IEC 27037:2012
2.- "Ley Provincial 13139-2010". Boletín Oficial de la Provincia de Santa Fe del martes 23 de noviembre de 2010.
3.- L. S. Gómez "Protocolo de Actuación para Pericias Informáticas" Poder Judicial de la Provincia de Neuquén.
4.- L. S. Gómez, "Pericias informáticas sobre telefonía celular" Poder Judicial de la Provincia de Neuquén.
5- "Guide to Integrating Forensic Techniques into Incident Response", National Institute of Standards and Technology Special Publication 800-86 Aug 2006
6.-"Recovering and Examining Computer Forensic Evidence", Forensic Science Communications October 2000, Federal Bureau of Investigation

7.- Auguste Kerckhoffs, "La cryptographie militaire", Journal des sciences militaires, vol. IX, pp. 5–83, Jan. 1883, pp. 161–191, Feb. 1883
8.-NCMEC, National Center for Missing and Exploited Children. USA
9.- www.digilogic.com.ar
10.- M. E. Casco; S. Roatta; N. Acosta; C. Kornuta; M. Marinelli "Análisis y Evaluación de Herramientas Libres Aplicadas a la Informática Forense" 40º Jornadas Científico – Tecnológicas UNaM, 2013.
11.- https://santoku-linux.com
12.- Junbin Fang, Zoe Jiang, Kam-Pui Chow, Siu-Ming Yiu, Lucas Hui, Gang Zhou, Mengfei He, Yanbin Tang "Forensic Analysis of Pirated Chinese Shanzhai Mobile Phones" IFIP Advances in Information and Communication Technology Volume 383, 2012, pp 129-142
13.- https://en.wikipedia.org/wiki/XRY_(software)
14.- http://lang.cellebrite.com/es/mobile-forensics/products/standalone/ufed-chinex

Probabilidades de referencia para aplicar en la detección de Infraestructuras de Clave Pública anómalas

Antonio Castro Lecthaler, Marcelo Cipriano, Eduardo Malvacio

{antonio.castrolechtaler; cipriano1.618; edumalvacio}@gmail.com,

CriptoLab. Escuela Superior Técnica – Instituto Universitario del Ejército Argentino – Cabildo 15. A1406CCC – Ciudad Autónoma de Buenos Aires, Argentina

Abstract: Este trabajo permite calcular la probabilidad teórica de hallar primos repetidos en una muestra determinada de certificados digitales emitidos, libres de sesgos. Tales valores pueden servir de referencia para elaborar un procedimiento estadístico que permita auditar y fiscalizar el comportamiento de una Infraestructura de Clave Pública (PKI), y así poder detectar anomalías, si existieran, en su funcionamiento y evitar vulnerabilidades de esa naturaleza.

Keywords: PKI, RSA**,** Certificados Digitales.

1 Introducción

Los certificados emitidos por una Infraestructura de Clave Pública (PKI por sus siglas en inglés: Public Key Infrastructure) en entornos y sistemas militares como del ámbito civil, redes Públicas o Privadas, Lan´s, o Wan´s o asimismo Internet, tienen amplia difusión. Entre otras aplicaciones de los mismos se pueden destacar: logueo y autenticación de usuarios, equipos y sistemas, cifrado y firma digital, no repudio, determinación de claves de sesión, etc.

Los certificados que emite una PKI, incluyen entre otros, un módulo *m* y un número *e* (generalmente 65537) conocidos como *"clave pública"* y un número *d* llamado *"clave privada"*. El valor *m*, que tiene un tamaño *t* (medido en bits) se obtiene por el producto de 2 valores primos. Este trío *(m,e,d)* es calculado por la PKI al momento de solicitar el certificado digital correspondiente y entregado a un usuario que será su poseedor.

Se produce una vulnerabilidad[1] si la PKI manifiesta alguna anomalía al calcular los valores *m* o se emiten certificados en los que dos o más usuarios comparten algún factor primo de sus respectivos módulos.

[1] La seguridad del *Sistema RSA* se basa en la dificultad de factorizar en tiempo aceptable, módulos *m*, (por ejemplo, t=1024, 2048 o 4096 bits como los empleados en la actualidad) y por ello preservar la clave secreta *d*. Con el conocimiento de uno de los factores primos de un determinado módulo, se puede calcular de manera sencilla el otro factor y con él la clave *d*, previo paso, que se calcula sin dificultad.

Dicha información permite eludir la seguridad ofrecida por RSA y obtener sin dificultad la clave privada, pudiendo entonces acceder a la información que se pretende proteger.

La complejidad de los sistemas actuales es muy grande y la detección de determinado tipos de errores no es sencilla [7]. El método tradicional es la lectura y fiscalización de las líneas de código que forman la PKI. La detección de errores, por otro lado, es una realidad que tiene muchos antecedentes, dentro de los cuales se pueden consultar [1,9].

Una interesante discusión puede darse acerca de la naturaleza de dichos errores: inocentes "bugs" que superaron las pruebas y se filtraron para ser detectados años después de su creación o fueron "plantados" con la intención de debilitar la seguridad.

Otros investigadores [8] han evaluado más de un millón de certificados de clave pública y descubrieron que cerca del 5% de los mismos compartían factores primos. ¿Este es un valor esperable, dada la magnitud de la muestra analizada o está fuera de las posibilidades en vista del tamaño de los módulos analizados y la cantidad de primos posibles?

Este trabajo y sus antecedentes[2-6], determina la *Función de Probabilidad* de encontrar o no colisiones de factores primos sobre un *espacio muestral* variable, compuesto de certificados. Tales probabilidades (que se encuentran libres de anomalías y sesgos pues fueron calculados teóricamente) podrán utilizarse como valores de referencia, para auditar y fiscalizar el comportamiento de una PKI determinada.

En los puntos 2 y 3 se presenta un modelo probabilístico para la realización del experimento E, sus posibles resultados. La metodología para determinar los resultados antes mencionados y finalmente se obtendrán las fórmulas que permitan calcular las probabilidades en los que en muestras de tamaño *mu*, haya o no primos repetidos de tamaño *t*.

En el punto 4 se presenta la Función de Probabilidad que rige en una PKI teórica, libre de anomalías y sesgos.

En el punto 5 se presentan diferentes fórmulas para calcular factoriales grandes, pues las fórmulas obtenidas en los puntos anteriores los requieren.

En el punto 6 se presentan las conclusiones y posibles continuaciones de esta investigación: determinación del modelo probabilístico oculto en una PKI a analizar, empleando herramientas provenientes de la *inferencia estadística*. Y finalmente la comparación entre los valores teóricos y los obtenidos empíricamente, que permitirán determinar la presencia de sesgo o anomalías. El cierre final sería la elaboración de un software que permita auditar PKI.

2 Cantidad de certificados emitibles por una PKI

Sea la *hipótesis*[2] *H1: el tamaño de los valores primos generados por la PKI, llamado aquí b, es la mitad del valor del tamaño t de los módulos.* Por ejemplo, si *t=1024* entonces los valores primos serán de *b=512* bits de tamaño.

Sea P_1 el conjunto de números primos de tamaño *b*.

$$P_1 = \{p \,/\, p \text{ primo};\ 2^{b-1} < p < 2^b\}. \tag{1}$$

El cardinal o cantidad de elementos de P_1 -llamado aquí p_1 - puede calcularse con una fórmula asociada al Teorema de los Números Primos[3]:

$$p_1 = Card(P_1) \approx \pi(2^b) - \pi(2^{b-1}). \tag{2}$$

$$p_1 \approx \frac{2^b}{ln\,2^b} - \frac{2^{b-1}}{ln\,2^{b-1}} = \frac{2^{b-1}}{ln\,2}\left(\frac{2}{b} - \frac{1}{b-1}\right). \tag{3}$$

Sea M_1 el conjunto de todos los módulos públicos que se pueden determinar a partir de los elementos del conjunto P_1:

$$M_1 = \{m/m = pq;\ p \neq q;\ p,q \in P_1\}. \tag{4}$$

Se asume aquí otra hipótesis de trabajo *H2: la PKI no emitirá certificados obtenidos como el producto del mismo factor primo*. Es decir que el módulo público no será un número cuadrado.

El cardinal de M_1 (que se indicará por m_1) es la cantidad de subconjuntos de 2 elementos del conjunto P_1, ya que cada módulo público es el producto de 2 valores primos y por la conmutatividad del producto, no importa el orden en el que se los multiplique.

$$m_1 = Card(M_1) = \binom{p_1}{2} = \frac{p_1(p_1 - 1)}{2}. \tag{5}$$

---

[2] A lo largo de este trabajo se irán asumiendo diferentes hipótesis respecto al contexto y entorno. Se irán describiendo y numerando medida que se vayan presentando.

[3] Conjeturado por el matemático alemán *Carl Gauss* (1777-1855) y demostrado de forma independiente por el matemático belga *Charles-Jean de la Vallée Poussin (1866-1962)* y el matemático francés *Jacques Hadamard* (1865-1963).

3. El experimento E, modelo probabilístico y su Función de Probabilidad

3.1 Definición del experimento E

Se propone la realización del experimento *E*:

Experimento E: *solicitar a la PKI la cantidad de* ***mu*** *certificados y con ellos formar el conjunto MU llamado "muestra".*

$$MU = \{m \,/\, m \text{ es un módulo público de tamaño } t\}. \tag{6}$$

$$Card(MU) = mu. \tag{7}$$

Se asume la *Hipótesis 3: la obtención de los módulos m está libre de sesgos*. Por lo que esto determina un modelo probabilístico en el que la probabilidad de obtener cualquier módulo es equiprobable.

3.2 Resultados del experimento

Este experimento puede tener 2 *resultados* o *eventos:*

$$R = \{r_1 ; r_2\}. \tag{8}$$

- r_1: que en el conjunto *MU* no haya módulos *m* que compartan algún factor primo. En cuyo caso se dirá que *no hay colisiones de factores primos*.
- r_2: que en el conjunto *MU* haya 2 o más módulos *m* que repitan factores primos. En cuyo caso se dirá que *sí hay colisiones de factores primos.*

Para determinar cuál es el resultado del experimento[4] se empleará el Máximo Común Divisor de todos los módulos, tomados de a pares:

$$\forall m_i, m_j \in MU (i \neq j); mcd(m_i, m_j) \in \{1; p\}, p \in P_1. \tag{9}$$

Por lo tanto, si todos los valores obtenidos del *mcd* es 1, el experimento tuvo resultado r_1 pues no se verifica que haya primos repetidos en dichos módulos. Caso contrario, el experimento tuvo resultado r_2.

[4] Se puede solicitar a la PKI los factores primos del módulo, junto con el resto de la información del certificado. Luego bastará con revisar si en la muestra hay o no primos repetidos. Ambas pruebas, el cálculo del mcd y como esta última descripta, tienen costo computacional que en el marco de este trabajo no se analizará. Lo recomendable es elegir la que tenga menor complejidad de las dos.

El proceso de aplicar el experimento *E* a todos conjuntos posibles *MU*, determina 2 conjuntos: R_1 y R_2:

$$R_1 = \{ MU / R(MU) = r_1 \}. \quad (10)$$

$$R_2 = \{ MU / R(MU) = r_2 \}. \quad (11)$$

Tales conjuntos tienen las siguientes propiedades:

a) Disjuntos:

$$R_1 \cap R_2 = \varnothing. \quad (12)$$

b) Complementarios:

$$R_1 \cup R_2 = EM(E). \quad (13)$$

Donde *EM(E)* es el *Espacio Muestra del experimento E*, es decir todos los conjuntos *MU* de tamaño *mu*.

$$EM(E) = m_1^{mu} \quad (14)$$

Sea la *hipótesis H4: la PKI no "recuerda" los certificados emitidos*. Por lo tanto podría repetir módulos en su espacio muestral.[5]

Definimos la Función de Probabilidad, de acuerdo a las propiedades de R_1 y R_2

$$p(R_1) + p(R_2) = 1. \quad (15)$$

$$p(R_2) = 1 - p(R_1). \quad (16)$$

Para determinar el valor de estas probabilidades utilizaremos la teoría clásica[6]. Para tal fin se deberá calcular el cardinal de cada conjunto y el total del espacio muestral.

3.3 Cardinalidad de R1 y R2

El cardinal de R_1 es la cantidad de conjuntos *MU*, de tamaño *mu*, formado por módulos de tamaño *t*, en los que se verifica que *no* haya colisiones de primos.

---

[5] En caso de asumir otra hipótesis respecto a la PKI, esto conllevaría el consecuente cambio de tamaño del espacio muestral.

[6] La Teoría de Probabilidades fue iniciada por *Pierre de Fermat (1601-1665)* y *Blaise Pascal (1623-1662)*. Sin embargo se debe *a Pierre-Simon Laplace (1749-1827)* la primera definición axiomática de probabilidad: sea la probabilidad de un suceso el cociente entre cantidad de resultados favorables del suceso y la cantidad total de resultados posibles.

Dados los conjuntos P_1 y M_1 indicados en las fórmulas (1, 3) respectivamente, se tiene que el 1er elemento de *MU* puede ser cualquiera de los m_1 elementos de M_1.

El segundo elemento, debiera ser un módulo coprimo con el primero de la muestra. Para tal fin determinamos al conjunto P_2 como el conjunto de números que resulta de quitar los números *p* y *q* que forman al primer elemento del conjunto P_1.

$$P_2 = P_1 - \{p; q\}. \tag{17}$$

$$p_2 = Card(P_2) = Card(P_1) - 2. \tag{18}$$

Sea M_2 el conjunto de todos los módulos que se puedan generar con P_2, cuyo cardinal m_2 se calcula como el combinatorio de todos los elementos de P_2, tomados de a 2.

$$M_2 = \{m/m = pq; p \neq q; p,q \in P_2\}. \tag{19}$$

$$m_2 = Card(M_2) = \binom{p_2}{2} = \frac{p_2(p_2 - 1)}{2}. \tag{20}$$

$$m_2 = \binom{p_1 - 2}{2} = \frac{(p_1 - 2)(p_1 - 3)}{2} \tag{21}$$

Con el mismo razonamiento, el 3er elemento de la muestra será un módulo coprimo con el primero y con el segundo elemento, tomados de a dos. Para ello se determina el conjunto de números primos P_3 que resulta de quitar de P_2 los factores *p* y *q* que determinan al segundo elemento.

$$P_3 = P_2 - \{p; q\}. \tag{22}$$

$$p_3 = Card(P_3) = Card(P_2) - 2 = Card(P_1.) - 4. \tag{23}$$

Sea M_3 el conjunto de todos los módulos que se puedan generar con P_3.

$$M_3 = \{m/m = pq; p \neq q; p,q \in P_3\}. \tag{24}$$

$$m_3 = Card(M_3) = \binom{p_3}{2} = \frac{p_3(p_3 - 1)}{2}. \tag{25}$$

$$m_3 = \binom{p_1 - 4}{2} = \frac{(p_1 - 4)(p_1 - 5)}{2}. \tag{26}$$

En general, para cualquier valor *i* entre *1* y *mu*, se tiene:

$$Pi = P_{i-1} - \{p; q\}. \tag{27}$$

$$p_i = Card(P_i) = Card(P_{i-1}) - 2 = Card(P_1.) - 2(i-1). \tag{28}$$

Sea M_i el conjunto de todos los módulos que se pueden obtener como producto de elementos de P_i:

$$M_i = \{m/m = pq; p \neq q; p,q \in P_i\} \tag{29}$$

$$m_i = Card(M_i) = \binom{p_i}{2} = \frac{p_i(p_i - 1)}{2}. \tag{30}$$

$$m_i = \binom{p_i - 2(i-1)}{2} = \frac{[p_1 - 2(i-1)][p_1 - 2(i-1) - 1]}{2} \tag{31}$$

.

Se puede observar como cada cardinal de los conjuntos M_i se van expresando en función del cardinal de M_1.

Siguiendo este procedimiento hasta llegar al último módulo de la muestra. El módulo número *mu* que es la cantidad prevista en el *experimento E*.

$$P_{mu} = P_{mu-1} - \{p; q\}. \tag{32}$$

$$p_{mu} = Card(P_{mu}) = Card(P_{mu-1}) - 2 = Card(P_1) - 2(mu - 1). \tag{33}$$

Sea M_{mu} el conjunto de todos los módulos que se puedan generar con P_{mu}

$$M_{mu} = \{m/m = pq; p \neq q; p,q \in P_{mu}\}. \tag{34}$$

$$m_{mu} = Card(M_{mu}) = \binom{p_{mu}}{2} = \frac{p_{mu}(p_{mu} - 1)}{2}. \tag{35}$$

$$m_{mu} = \binom{p_1 - 2(mu - 1)}{2} = \frac{[p_1 - 2(mu - 1)][p_1 - 2(mu - 1) - 1]}{2}. \tag{36}$$

Todos los módulos del conjunto *MU*, son coprimos, tomados de a 2.

Finalmente, el cardinal del conjunto de todas las muestras de *mu* módulos, en las que no hay colisiones de primos, queda determinado por:

$$Card(R_1) = \prod_{i=1}^{mu} m_i = \prod_{i=0}^{mu-1} \binom{p_1 - 2i}{2}. \tag{37}$$

$$Card(R_1) = \frac{\prod_{i=0}^{2(mu-1)} (p_1 - i).}{2^{mu}}. \tag{38}$$

Luego,

$$Card(R_1) = \frac{p_1!}{2^{mu}(p_1 - 2(mu-2))!} \quad (39)$$

3.4 Cardinalidadp de R2

Dado que R_1 y R_2 son disjuntos y complementarios tal como se ha mostrado en (12-14), se tiene que:

$$Card(R_2) = EM(E) - Card(R_1). \quad (40)$$

$$Card(R_2) = m_1^{mu} - \frac{p_1!}{2^{mu}(p_1 - 2(mu-2))!}. \quad (41)$$

Siendo m_1 la cantidad de módulos que se pueden calcular con los factores primos del conjunto P_1, cuyo cardinal es el valor p_1 y mu que es la cantidad de módulos en cada muestra.

3.5 Función de Probabilidad

Tal como se expresa en (16-17), entonces:

$$p(R_1) = \frac{card(R_1)}{m_1^{mu}}. \quad (42)$$

$$p(R_1) = \frac{\frac{p_1!}{2^{mu}[p_1 - 2(mu-2)]!}}{m_1^{mu}}. \quad (43)$$

Luego, por (5):

$$p(R_1) = \frac{\frac{p_1!}{2^{mu}[p_1 - 2(mu-2)]!}}{\frac{[p_1(p1-1)]^{mu}}{2^{mu}}}. \quad (44)$$

$$p(R_1) = \frac{p_1!}{[p_1 - 2(mu-2)]![p_1(p1-1)]^{mu}}. \quad (45)$$

$$p(R_2) = 1 - \frac{p_1!}{[p_1 - 2(mu-2)]![p_1(p1-1)]^{mu}}. \quad (46)$$

4 Cálculo de factoriales grandes

Estas fórmulas requieren resolver factoriales muy grandes, cuya complejidad computacional dificulta dicho cálculo. A modo de ejemplo, se darán algunas fórmulas para aproximar el valor de los factoriales:

$$n! = e^{\ln n!} \approx e^{n(\ln n - 1)}. \tag{47}$$

$$n! \approx n^n e^{-n} \sqrt{2\pi n}. \tag{48}$$

$$n! \approx n^n e^{-n} \sqrt{\pi} \sqrt[6]{8n^3 + 4n^2 + n + \frac{1}{30}}. \tag{49}$$

Conociéndose (47) y (48) como las fórmulas de *Stirling*[7] y (49) de *Ramanujan*[8].

$$n! \approx \sqrt{2\pi} \left(\frac{n + \frac{1}{2}}{e} \right)^{n+\frac{1}{2}}. \tag{50}$$

$$n! \approx n^n e^{-n} \sqrt{\pi} \sqrt{2n + \frac{1}{3}}. \tag{51}$$

$$n! \approx n^n e^{-n} \sqrt{2\pi \left(n + \frac{1}{6} + \frac{1}{72n} - \frac{31}{6480n^2} - \frac{139}{155520n^3} + \frac{9871}{6531840n^4} \right)}. \tag{52}$$

Conocidas estas expresiones como las fórmulas de *Burnside*[9], *Gosper*[10] y *Batir*[11], respectivamente.

5 Conclusiones y futuros trabajos

Se han presentado las fórmulas para calcular las probabilidades matemáticas de hallar en una muestra colisiones de primos, calculados a partir de una fuente libre de sesgos y anomalías.

Se puede asumir como hipótesis la existencia de *permanencia estadística*, es decir que a lo largo de la realización del experimento E, aplicado a una PKI determinada, su modelo probabilístico desconocido puede ser revelado a través de herramientas estadísticas.

---

[7] *James Stirling* (1692-1770). Matemático escocés.

[8] *Srinivasa Ramanujan* (1887-1920). Matemático indio. No dejó una demostración de su fórmula. Fue demostrada en 2000 por la matemática rusa *Ekatherina Karatsuba*.

[9] *William Burnside* (1852-1927). Matemático inglés.

[10] *Ralph Gosper, Jr.* (1943-). Matemático y científico de computadoras estadounidense.

[11] *Necdet Batir* (1959 -). Matemático turco.

Habría anomalía en el comportamiento de la PKI si se verifican divergencias entre los valores de referencia propuestos en este trabajo y los obtenidos por "experiencia directa"

Quedan así planteadas cuestiones para continuar la investigación y poder elaborar un software que pueda auditar y fiscalizar anomalías en Infraestructuras de Clave Pública.

6 Referencias

[1] Bello L, Bertacchini M. *"Generador de Números Pseudo-Aleatorios Predecible en Debian"*. III Encuentro Internacional de Seguridad Informática. Manizales, Colombia. Octubre 2009.

[2] Benaben, A; Castro Lechtaler, A; Cipriano, M; Foti, A. *"Development, testing and performance evaluation of factoring algorithms whit additional information*" XXVIII Conferencia Internacional de la Sociedad Chilena de Computación. Santiago de Chile. 2009.

[3] Castro Lechtaler, C; Cipriano, M; Benaben A; Quiroga, P. *"Study on the effectiveness and efficiency of an algorithm to factorize N given e and d*". IX Seminario Iberoamericano en Seguridad de las Tecnologías de la Información, La Habana, CUBA. 2009.

[4] Castro Lechtaler, A; Cipriano, M. "*Detección de anomalías en Oráculos tipo OpenSSL por medio del análisis de probabilidades*". XVII Congreso Argen-tino de Ciencias de la Computación CACIC 2011. La Plata, Buenos Aires, Octubre 2011.

[5] Castro Lechtaler, Antonio, Cipriano Marcelo; Malvacio Eduardo; Cañón, Sebastián; *Procedure for the Detection of Anomalies in Public Key Infrastruture (RSA Systems)*. XIII Simposio Argentino de Tecnología, 41 Jornadas Argentinas de Informática e Investigación Operativa JAIIO – SADIO. La Plata, Buenos Aires, Agosto 2012.

[6] Castro Lechtaler, Antonio; Cipriano, Marcelo; Malvacio, Eduardo. *Experimental detection of anomalies in public key infrastructure*. XVIII Congreso Argentino de Ciencias de la Computación CACIC 2012. Bahía Blanca, Buenos Aires, Octubre 2011.

[7] Glass, Robert *"Facts and Fallacies of Software Engineering*". Addison-Wesley Professional, 2003

[8] Lenstra, A; Hughes, J; Augier, M y otros. Ron was wrong, Whit is right. e-print International Association for Cryptologic Research. 15 Feb 2012. http://eprint.iacr.org/2012/064.

[9] Young A and Yung M. *An Elliptic Curve Asymmetric Backdoor in Open-SSL RSA Key Generation.* Chapter 10. Cryptovirology. 2006.

Controles de Seguridad propuesta inicial de un Framework en el Contexto de la Ciberdefensa.

Pablo Gastón Sack , Jorge S Ierache.

Facultad de Informática Ciencias de la Comuicación y Técnicas Especiales
Universidad de Morón Cabildo 134 Morón Argentina
jierache@unimoron.edu.ar

Abstract.: El presente trabajo explora un conjunto de definiciones de ciberespacio, ciberguerra, ciberdefensa, y de características de controles de seguridad a fin de realizar una propuesta inicial de un Framework que facilite la gestión y diagnóstico de seguridad en el contexto de la ciberdefensa.

Keywords: Ciberespacio, Ciberguerra, Ciberdefensa, Seguridad Informática, Controles de Seguridad.

1 Introducción

La constante evolución humana ha llevado a la generación de un nuevo espacio artificial que resulta transversal a los espacios naturales (terrestre, marítimo, aéreo, espacial) en los que la humanidad se desarrolla naturalmente. El nuevo ambiente o espacio denominado ciberespacio está plenamente integrado en las actividades humanas, no reconoce fronteras físicas ni estados naciones, permite la evolución de las operaciones en términos de interoperabilidad de los sistemas en los distintos ambientes naturales. En la Primera Guerra Mundial se introdujeron los aviones, una nueva forma de sorprender y atacar a las fuerzas en oposición, algo similar (salvando las distancias tecnológicas) ocurre en la actualidad con el ciberespacio. Desde este nuevo espacio artificial se presenta un campo de batalla , presente en la puerta de cada computadora infectada por un software malicioso (malware) puede ser un elemento de ataque a gran escala a un país/agencia o empresa sin que su dueño esté enterado. Internet no es un lugar seguro, ya que hay personas que buscan delinquir en la red ya sea por diversión, por dinero, por motivos políticos, etc. El ciberespacio se presenta como el nuevo ambiente artificial creado por el hombre, que actúa como integrador de los ambientes naturales donde actúa la humanidad. Desde el principio de la historia de la humanidad, los conflictos y las guerras, aumentaron sus alcances y efectos acompañados por los avances científicos tecnológicos. El hombre ha luchado entre sí por diferencias culturales, disputas territoriales, políticas, por recursos, sometimiento, hoy lucha por el dominio del ciberespacio. Se presenta en la Fig. 1 la integración de ambientes a través del ciberespacio, se detalla a continuación cada uno de ellos: a) Tierra (1) y Mar (2): Con el paso del tiempo el hombre ha ampliado y dominado nuevos campos de batalla, siendo sus primeros comienzos el campo terrestre y marítimo; b) Aire (3): Con la llegada de la Primera Guerra Mundial se comenzó con

el dominio aéreo teniendo una vital importancia para la Segunda Guerra Mundial cambiando drásticamente la forma de hacer una guerra; c) Espacio (4): Ya avanzada la década del 60 fue el auge de la carrera espacial dominando así un nuevo campo de batalla muy diferente a los anteriores los cuales están definidos en gran medida por la geografía o radio de operación; d) Ciberespacio (5): Con el advenimiento de Internet desde hace varias décadas, los países (principalmente los desarrollados) se han volcado de lleno a esta nueva era digital en la que todo está interconectado con todo.

Fig. 1. Integración de Dominios

1.1 Ciberespacio

La palabra ciberespacio surge de la conjunción de la palabra "cibernao" -proveniente del griego que significa "pilotear una nave" y es utilizado comúnmente en el ámbito de las redes -, y espacio dando así la idea de estar piloteando o navegando sobre un mundo virtual. Éste término fue utilizado por primera vez en la década de los '80 por el escritor William Gibson para describir una red de computadoras ficticia que contenía enormes cantidades de información que podría explotarse con el fin de adquirir riquezas y poder [1]. En la actualidad el ciberespacio se le da un significado más amplio al que se lo aglomera en la conjunción de toda la información disponible (digitalmente) junto con el intercambio de la información y las comunidades electrónicas que surgen en base al uso de esa información [2]. El ciberespacio no debe confundirse con el Internet real, el término se refiere a menudo a los objetos e identidades que existen dentro de la red informática. Esto quiere decir que una página Web (por ejemplo), se encuentra en el ciberespacio. Según esta interpretación, los acontecimientos que tiene lugar en Internet no están ocurriendo en los países donde los participantes o los servidores se encuentran físicamente, sino en el ciberespacio [3]. Otro punto de vista que vale la pena mencionar es una definición de ciberespacio que abarca todo el espectro electromagnético y propagación de energía. En otras palabras, todo lo que fluya a través del espectro electromagnético (como celulares, Internet, etc); Si emite, transmite, usa el ciberespacio. En septiembre de 2006, Jefes de Estado Mayor de los Estados Unidos lo definieron como "dominio caracterizado por el uso de electrónica y espectro electromagnético para el almacenamiento, modificación e intercambio de información vía sistemas en red e infraestructuras físicas asociadas" [4]. El ciberespacio sobrepasa los límites de cómo

y cuándo interactuar. Entre las características del ciberespacio están las siguientes [5]: a) Identidad, flexibilidad y anonimato: La falta de interacción física cara a cara causa un impacto en cómo la gente presenta su identidad o quizás quedarse en el anonimato, incluso se puede tener una identidad imaginaria o falsa; b) En el ciberespacio todos tenemos la misma oportunidad de comunicación; c) Trasciende los límites espaciales: Las distancias geográficas no limitan quién pueda comunicarse con quién; d) Tiempo extendido y condensado: puede haber una comunicación cualquiera vía Internet, pueden haber varias personas sentadas en su computadora al mismo tiempo. Este tipo de comunicación crea un espacio temporal donde el estar, como tiempo interactivo se extiende. Es decir, se tiene tiempo para pensar cosas y dar una respuesta. El Ciberespacio en el contexto del campo de batalla ha ido creciendo y convirtiéndose en algo más difícil de definir y defender, su ámbito de actuación propio de Internet proveyó un mecanismo de acción que incrementa la velocidad, difusión y poder en un ataque [6], a diferencia de nuestro entendimiento histórico de la guerra, el ciberespacio como campo de batalla posee ventajas asimétricas a favor del atacante. En este orden resulta oportuno considerar que: a) La inteligencia y engaño serán los principios críticos de la guerra en el ciberespacio; b) El ciberespacio es vasto y es fácil de esconderse en el mismo; c) Los efectos de los ataques son desproporcionados a los costos de los mismos; d) El uso del ciberespacio no requiere un uso intensivo de materiales o capital; e) Se puede lanzar ataques desde cualquier parte del mundo, incluso el mismo país o red del atacado proporcionando un anonimato. Éste punto representa una dificultad extra ya que no es fácil ubicar al agresor e incluso pueden ser grupos independientes (no gubernamentales) o incluso mismos ciudadanos; f) El atacante puede tomarse todo el tiempo necesario para la adquisición de recursos u oportunidades para garantizar la conquista de su objetivo. Diferentes definiciones del Ciberespacio se pueden presentar. En este orden la **Real Academia Española** lo define como el: Ámbito artificial creado por medios informáticos. Esto quiere decir que para implementar el ciberespacio se necesita de una infraestructura física de computadoras y líneas de comunicaciones que las mantengan interconectadas. Para el **National Institute of Standards and Technology (NIST)** Ciberespacio se define como **el** dominio global dentro del entorno de la información que consta de redes interdependientes de infraestructuras de sistemas de información que incluyen: internet, redes de telecomunicaciones, sistemas informáticos, procesadores embebidos y controladores. Para la **Unión Europea** se define como el: espacio virtual por donde circulan los datos electrónicos de los ordenadores del mundo. Para la **Unión Internacional de Telecomunicaciones** se define como: un lugar creado a través de la interconexión de sistemas de ordenador mediante Internet. **España** lo define como el: Conjunto de medios físicos y lógicos que conforman las infraestructuras de los sistemas de comunicaciones e informáticos. **Estados Unidos** (DoD) lo define como: un dominio global dentro del entorno de la información, consistente en la red interdependiente de las infraestructuras de tecnología de la información incluida la Internet, redes de telecomunicaciones, sistemas informáticos, los procesadores y controladores embebidos. **Estados Unidos** (National Military Strategy for Cyberspace Operations) lo define como: El dominio que se caracteriza por el uso de la electrónica y el espectro electromagnético para

almacenar, modificar e intercambiar datos a través de sistemas de redes e infraestructuras físicas asociadas. Para **Alemania** el ciberespacio se define como: espacio virtual de todos los sistemas informáticos vinculados a nivel de datos a escala global. La base para el ciberespacio es el Internet como una red de conexión y transporte universal y accesible al público que puede ser complementada y más expandido en cualquier número de redes de datos adicionales. Sistemas de informáticos en un espacio virtual aislado no son parte del ciberespacio.. Para el **Reino Unido**: Todas las formas de actividades en redes digitales; esto incluye el contenido y acciones realizadas a través de redes digitales. En una visión particular de **ciberespacio** por parte de los autores propone como definición del ciberespacio: al ámbito artificial transversal a los ambientes naturales (terrestre, marítimo, aéreo y espacial) que conforma el espacio virtual de interacción en el que se desarrollan actividades propias de humanos y máquinas relacionadas con la creación, procesamiento, publicación, almacenamiento, modificación y explotación de datos, información y conocimiento digitales, en un contexto distribuido (computación en nube) a través de redes interdependientes e interconectadas globales, públicas, privadas, híbridas, software y firmware de máquinas, cuyo carácter distintivo está dado por el empleo de las tecnologías de información y comunicaciones.La **ciberguerra** es definida por Richard Clarke como las acciones realizadas por un estado Nación que penetra computadoras o redes de otras naciones con el propósito de causar daño o ruptura de las mismas. Wisegeek la define como la forma de guerra que toma lugar en las computadoras e Internet a través de mecanismos electrónicos por sobre los físico. **China** define Ciberguerra como: las acciones adoptadas para lograr la superioridad de información al afectar la información adversario, sus procesos basados en la información, sistemas de información y redes informáticas, mientras que se realiza la defensa de la propia información, los procesos basados en la información, sistemas de información y redes informáticas.**Estados Unidos (DoD)** Cyber warfare (CyW) — Cualquier acto destinado a obligar a un oponente para cumplir nuestra voluntad nacional , ejecutado contra el software de control de procesos dentro del sistema de un oponente. CYW incluye los siguientes modos de ataque cibernético : la infiltración cibernética , la manipulación cibernética , asalto cibernético, y la incursión cibernética..La **Ciberdefensa** es definida por La OTAN como: El desarrollo de la capacidad de prevenir, detectar, defenderse y recuperarse de los ataques cibernéticos. Por lo tanto, la defensa se centra en el uso de métodos tecnológicos para identificar una intrusión no autorizada, localizar el origen del problema, evaluar los daños, evitar la propagación de los daños dentro de la red, y en la medida necesaria, la reconstrucción de los datos y de los equipos que se encontraban dañados. Defensa implica [7] la capacidad de colocarse en el camino de penetración, identificar tal intento, y frustrar a través de la interrupción y suspensión de la tareas. Para este propósito, los sistemas informáticos se utilizan para supervisar las actividades y las comunicaciones; bloquear vías de acceso; limitación de permisos; verificación de identidad; proporcionar cifrado y habilitar la copia de seguridad y recuperación de desastres. Para **Colombia Ciberdefensa se define como la:** Capacidad del Estado para prevenir y contrarrestar toda amenaza o incidente de naturaleza cibernética que afecte la soberanía nacional. Para **Estados Unidos**

(Comprehensive National Cybersecurity Initiative (CNCI)): La defensa de todo el espectro de amenazas mediante la mejora de las capacidades de contrainteligencia de EEUU y el incremento de la seguridad de las cadenas claves de suministro de información.

2 Controles de Seguridad

En el contexto de la Ciberdefensa para su estudio se propone armar una base sólida y progresiva de controles de seguridad tomando como esqueleto y estructura el documento "The Critical Security Controls for Effective Cyber Defense" [8]. Complementar la base de controles de seguridad con lo planteado en "Strategies to Mitigate Targeted Cyber Intrusions" [9] y en el documento de "Security and Privacy Controls for Federal Information Systems and Organizations (800-53 Rev.4)" [10]. Se toma como punto de partida y de referencia el documento "Framework for Improving Critical Infrastructure Cybersecurity" de National. [11]

2.1 The Critical Security Controls for Effective Cyber Defense

El documento de Critical Security Controls for Effective Cyber Defense [8], fue realizado por una comunidad de más de 100 agencias gubernamentales, compañías privadas y expertos. Estos controles actúan como una plataforma sólida para construir otros estándares, puede ser utilizado para crear un mapa de ruta o punto de partida para quienes no saben por dónde empezar. El objetivo principal de este documento es reducir los ataques iniciales incrementando los niveles de seguridad al configurar los dispositivos, la identificación de dispositivos comprometidos, la interrupción del control de los atacantes sobre el código malicioso implantado y encontrar la forma de que todo esté integrado en un circuito de retroalimentación (evolución continua). Los principios críticos en que se basa este documento son: a) La ofensiva informa a la defensa: Usar el conocimiento de ataques actuales para aprender de estos eventos y construir defensas más efectivas; b)Priorizar: Invertir en controles que proveerán un mayor grado de reducción de riesgo; c) Métricas: Establecer sistemas de medición comunes que puedan dar un lenguaje común para la gerencia, los auditores, el grupo de informática y el grupo de seguridad de la organización; d) Diagnóstico: llevará a cabo mediciones continuas para evaluar y validar la efectividad de las mediciones de seguridad actuales; e)Automatización: Automatizar defensas para que las organizaciones puedan obtener mediciones confiables, escalabres y continuas .Por otro lado, este documento está divido en veinte controles críticos de seguridad que contienen a su vez un grupo de subcontroles. Dichos subcontroles se categorizan con el fin de implementar controles de manera progresiva y escalonada de la siguiente forma: a) **Logros Rápidos**: Estos subcontroles ofrecen una gran reducción del riesgo sin grandes inversiones financieras, de arquitectura o técnicas (orientado a ataques comunes); b) **Medidas de Visibilidad y Atribución**: Estos subcontroles ofrecen mejorar el proceso, la arquitectura y las capacidades técnicas de las organizaciones para monitorear sus redes y sistemas informáticos con el objetivo de detectar intentos

de ataque, localizar puntos de entrada, identificar dispositivos ya comprometidos, interrumpir actividades de atacantes infiltrados y obtener información sobre las fuentes del ataque; c)**Mejora de la Configuración de Seguridad de la Información**: Estos subcontroles ofrecen reducir el número y magnitud de las vulnerabilidades de seguridad y mejorar las operaciones de los sistemas informáticos en red; d) Subcontroles Avanzados: Estos subcontroles son usados en nuevas tecnologías o procedimientos que provee máxima seguridad pero son difíciles de implementar, más caro o requiere de personal altamente capacitado.

2.2 Strategies to Mitigate Targeted Cyber Intrusions [9]

El documento Strategies to Mitigate Targeted Cyber Intrusions" [9] está dividido en tres documentos desarrollados por el Departamento de Defensa de Australia y se compone de la siguiente manera: a) Mitigation Strategies 2014: contiene una breve introducción y un poster con el resumen de las treinta y cinco estrategias para mitigar ciberataques; b)Mitigation Strategies 2014 Details: Describe cada una de las treinta y cinco estrategias para mitigar ciberataques. Se menciona un código de control recomendado; c)Information Security Manual 2014 Control: Describe los códigos de controles mencionados en el documento anterior. Si bien este documento plantea treinta y cinco estrategias para la mitigación de los ciberataques remarca que como paso primordial es la de desarrollar y aplicar las primeras cuatro estrategias a las que denominan como esenciales. Dichas estrategias consisten en la generación de listas blancas de aplicaciones, parches de aplicaciones, parches de vulnerabilidades de sistemas operativos y restricción de privilegios de administrador.

2.3 Security and Privacy Controls for Federal Information Systems and Organizations

El documento "Security and Privacy Controls for Federal Information Systems and Organizations (800-53 Rev.4)" [10], forma parte de un "ciclo de vida de la seguridad" propuesto por NIST formando un marco de trabajo de gestión de riesgos. El "ciclo de vida de la seguridad" propuesto está compuesto por seis etapas con su correspondiente documento: a) Categorizar Sistemas de Información: 800-60; b) Seleccionar Controles de Seguridad: 800-53 Rev.4, c) Implementar Controles de Seguridad: 800-160; d) Evaluar Controles de Seguridad: 800-53A; e) Autorizar Sistemas de Información: 800-37; f) Monitorear Estado de Seguridad: 800-137.

2.4 Framework for Improving Critical Infrastructure Cybersecurity

Este marco de trabajo denominado como "Framework for Improving Critical Infrastructure Cybersecurity" de National" [11], provee un lenguaje común para el entendimiento, gestión, expresando el riesgo en la ciberseguridad, tanto interna como externa. Este marco de trabajo está compuesto por tres partes: a) **Núcleo**: Está compuesto por un conjunto de actividades de ciberseguridad, resultados deseados y referencias aplicables que son comunes en todos los sectores de infraestructuras

críticas. Se compone de cinco funciones concurrentes y continuas: **Identificar, Proteger, Detectar, Responder y Recuperar**. Dichas funciones proveen una vista estratégica del ciclo de vida de la gestión de riesgos de ciberseguridad de una organización. A su vez, cada una de estas funciones principales se dividirán en Categorias, Subcategorias y Referencias informativas; b) **Niveles de Implementación:** Proporcionar un contexto sobre cómo una organización ve los riesgos de ciberseguridad y los procesos para gestionar ese riesgo. Se compone de cuatro niveles: **Parcial, Riesgo Informado, Repetible y Adaptativo.** Estos niveles reflejan una progresión de respuestas informales y reactivas a enfoques ágiles y riesgo informado. c) **Perfiles:** Representa los resultados en base a las necesidades del negocio que una organización ha seleccionado de las categorías y subcategorías del marco de trabajo. Pueden ser utilizados para identificar las oportunidades para mejora la postura de ciberseguridad mediante la comparación de un perfil "actual" con un perfil "objetivo. Para el desarrollo de un perfil, una organización puede revisar todas las categorías y subcategorías y, basándose en los objetivos del negocio y una evaluación del riesgo, determinar cuáles son los más importantes.

3 Modelo de Framework propuesto

Sobre la base de controles y el marco de trabajo presentados en la sección anterior, se presenta a continuación la propuesta de un Modelo de Framework que facilite la implementación de controles de manera progresiva y realice un seguimiento de avance a través de las métricas en cada uno de los controles, ofreciendo documentos de consulta por cada control involucrado y links sugeridos. Para esto utilizaremos la **división de los controles de seguridad** propuestos en Framework for Improving Critical Infrastructure Cybersecurity [11]. Dicho documento plantea una división de los controles de seguridad propuestos en cinco fases. Estas fases aglutinarán controles de seguridad que cumplan con el objetivo de cada fase (ver Fig. 2).

Fig. 2. Fases del "Framework for Improving Critical Infrastructure Cybersecurity

3.1 Iteraciones dentro del marco de trabajo

Adicionalmente a la **división de los controles de seguridad** en cada una de las fases, el marco de trabajo constará de **cuatro iteraciones** que cruzarán todas las fases del mismo (ver Fig. 3). Estas iteraciones tendrán como objetivo lo descripto en la categorización de los subcontroles comentados en la sección anterior del documento

The Critical Security Controls for Effective Cyber Defense [8], logrando así la conformación de una grilla que permitirá ubicar los controles de acuerdo a la fase (Identificar, Proteger, Detectar, Responder, Recuperar) e iteración. Las cuatro iteraciones se clasifican como: a) **Logros Rápidos** (primera iteración): Controles, Métricas y recomendaciones para brindar protecciones inmediatas sin un mayor costo o cambio de infraestructura; b) **Medidas de Visibilidad y Atribución** (segunda iteración): Mayores controles, métricas y recomendaciones para monitorear y controlar la efectividad de dichos controles; **Mejora de la Configuración de Seguridad de la Información (**tercera iteración): Reducir prácticas débiles de seguridad, vulnerabilidades y sus impactos, mantenimiento y configuración apropiada de los controles; d) **Subcontroles Avanzados** (cuarta iteración): Minimizar posibilidades de éxito de atacantes con determinación, instalación de controles de alto costo.

Fig. 3. Secuencia a seguir para el marco de trabajo propuesto

3.2 Función del Marco de Trabajo

Cada cuadrícula conformada entre una fase y una iteración contendrá una "X" cantidad de controles a implementar y cumplir. La sumatoria del cumplimiento de cada uno de los controles dará el grado de cumplimiento de cada cuadrícula. El Marco de trabajo también tendrá la posibilidad de calcular el grado de cumplimiento no sólo de cada cuadricula sino también por fase e iteración. La Fig. 4 muestra un ejemplo de la interfaz propuesta, donde se pueden apreciar una categorización por colores que indican los grados de cumplimiento.

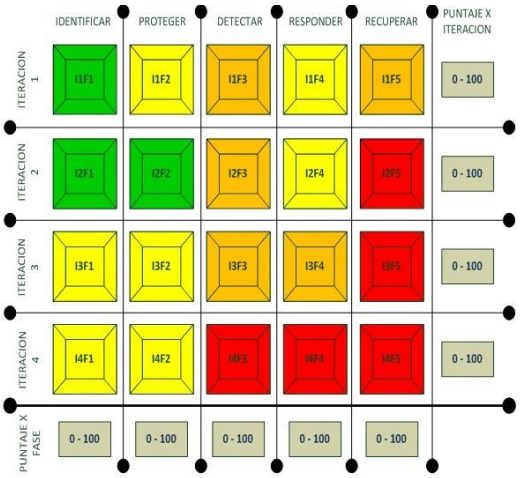


Fig. 4. Interfaz de explotación del Framework propuesta

3.3 Funcionamiento de la interfaz de carga.

Como se explicó anteriormente, cada cuadrícula estará compuesta por un listado de controles en la interfaz de carga. Dicha interfaz mostrará por cada control de seguridad un link a una documentación de ayuda y referencia para poder dar un valor de cumplimiento del control como así también links de sitios recomendados para mayor información. La interfaz de carga facilita parametrizar cada uno de los controles para la necesidad de cada usuario modificando la columna de Obligatorio / Recomendado, la ponderación que el usuario asigne o considere apropiada, como así también asignar un puntaje de cumplimiento. En la Fig. 5 se puede ver un ejemplo de la cuadrícula correspondiente a la fase 1 e iteración 1. Recordemos que el objetivo de la fase 1 corresponde a identificar "recursos permitidos de la empresa (SW, HW, PC's, etc.) y la iteración 1 corresponde a controles, métricas y recomendaciones para brindar protecciones inmediatas sin un mayor costo o cambio de infraestructura. El usuario al acceder a la interfaz de carga se encontrará con el listado de controles correspondientes a esta cuadrícula con una parametrización estándar recomendada que podrá ser modificada según la necesidad particular del mismo. El primer paso del usuario es considerar que controles son obligatorios y cuales son recomendados. Posteriormente se deberá asignar una ponderación de cada control y el valor a elegir estará contenido entre 0 y 1. Con respecto al puntaje el usuario irá completando el grado de cumplimento del control y el valor a ingresar estará contenido entre 0 y 100. El cálculo del total de la cuadrícula se determina de la siguiente manera: a) se calculará el Subtotal máximo de cada control de seguridad: Subtotal (Máximo) = Ponderación Ingresada X 100, b) se calculará el Subtotal real de cada control de seguridad: Subtotal = Ponderación Ingresada X Puntaje otorgado .c) Siendo el total de la cuadrícula igual a la relación entre la sumatoria de los subtotales y la sumatoria de los subtotales máximos, multiplicado por 100 para obtener el porcentaje. Siguiendo

con el ejemplo de la Fig. 5 el total de la cuadrícula sería igual a: Total Cuadricula (407,60/462)* 100= 88,22%.

INTERFAZ DE CARGA
PRIMERA ITERACIÓN, PRIMERA FASE

CONTROLES, MÉTRICAS Y RECOMENDACIONES PARA BRINDAR PROTECCIONES INMEDIATAS SIN UN MAYOR COSTO O CAMBIO DE INFRAESTRUCTURA

N.U.C.	NOMBRE DEL CONTROL	OBLIGATORIO / RECOMENDADO	PONDERACION	PUNTAJE	SUBTOTAL
I1F1-CCS01-01	Generar Inventario preliminar de dispositivos	OBLIGATORIO	0,7	80	56
I1F1-CCS01-02	Complementar el inventario preliminar de dispositivos con información DHCP	OBLIGATORIO	0,3	100	30
I1F1-CCS01-03	Automatizar la actualización del inventario de dispositivos	OBLIGATORIO	0,35	100	35
I1F1-CCS01-04	Cantidad de dispositivos no autorizados conectados en la red	RECOMENDADO	0,2	100	20
I1F1-CCS01-05	Tiempo de detección de nuevos SW instalados (min).	OBLIGATORIO	0,4	50	20
I1F1-CCS01-06	Tiempo de envío de alertas a los administradores por instalación de SW no autorizados(min).	RECOMENDADO	0,5	100	50
I1F1-CCS02-07	Tiempo para alertar que una aplicación de SW fue descubierta (min).	RECOMENDADO	0,49	100	49
I1F1-CCS02-08	Cantidad de aplicaciones de SW no autorizadas están instaladas en la red.	RECOMENDADO	0,27	100	27
I1F1-CCS02-09	Tiempo promedio para eliminar aplicaciones de SW no autorizadas.(min)	OBLIGATORIO	0,3	100	30
I1F1-CCS02-10	Es la política por defecto denegación de ejecución e un software (si o no)	OBLIGATORIO	0,51	60	30,6
I1F1-CCS02-11	Las carpetas de los ejecutables tienen restricciones de acceso a escritura (si o no)	OBLIGATORIO	0,6	100	60

Fig. 5. Interfaz de carga

4 Conclusiones y futuras líneas de trabajo

Se realiza un proceso exploratorio en término a las definiciones de ciberespacio, ciberguerra, ciberdefensa. Se propuso armar una base sólida y progresiva de controles de seguridad tomando como base inicial diferentes documentos de seguridad a fin de generar una propuesta inicial de un Framework de explotación, que facilite la gestión y diagnóstico de seguridad en el contexto de la ciberdefensa. En futuras líneas de trabajo se pretende articular la valoración de los controles y su análisis con el empleo de sistemas basados en conocimiento que asistan a la determinación del diagnóstico de seguridad de una unidad en el marco de la ciberdefensa.

5 Referencias

1. Gibson, W. (1984). Neuromancer. Ace Books(Gibson, 1984).
2. Umphress, T.C. (2007). Air&Space Power Journal. Obtenido de: “El Ciberespacio: ¿Un Aire y un espacio Nuevo? http://www.airpower.au.af.mil/apjinternational/apj-s/2007/3tri07/umphress.html
3 Gonzalez, J.A. (06/03/2010) Obtenido de: http://es.scribd.com/doc/32463107/grupo-de-el-ciberespacio

4.Wynne, H.M. (01/03/2007). *Air&Space Power Journal*. Obtenido de Flying and Fighting in Cyberspace: http://www.airpower.au.af.mil/airchronicles/apj/apj07/spr07/wynnespr07.html

5 ND. (2012). *La enciclopedia cubana en la red*. Obtenido de Ciberespacio: http://www.ecured.cu/index.php/Ciberespacio

6. Geers, K. (2010). A Brief Introduction to Cyber Warfare. *Common Defense Quarterly*, 16-17

7 Tabansky, L. (May de 2011). Basic Concepts in Cyber Warfare. Obtenido de INSS, The Institute for National Security Studies:http://www.inss.org.il/index.aspx?id=4538&articleid=2351

8 The Critical Security Controls for Effective Cyber Defense. Version 5 (CSC-5) del "Council on CyberSecurity. https://www.sans.org/media/critical-security-controls/CSC-5.pdf

9 Strategies to Mitigate Targeted Cyber Intrusions" del "Department of Defense – Intelligence and Security of Australian Government.http://www.asd.gov.au/infosec/mitigationstrategies.htm

10 Security and Privacy Controls for Federal Information Systems and Organizations (800-53 Rev.4) de National Institute of Standards and Technology (NIST).http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf

11 Framework for Improving Critical Infrastructure Cybersecurity de National Institute of Standards and Technology (NIST).http://www.nist.gov/cyberframework/upload/cybersecurity-framework-021214-final.pdf

Automatizando el descubrimiento de portales de autenticación y evaluación de la seguridad mediante ataques de fuerza bruta en el marco de una auditoría de seguridad

Gastón Traberg, Lía Molinari, Paula Venosa, Nicolás Macia, and Einar Lanfranco

LINTI, Facultad de Informática,
Universidad Nacional de La Plata,Buenos Aires, Argentina
{gtraberg}@cert.unlp.edu.ar
{lmolinari,pvenosa,nmacia,einar}@info.unlp.edu.ar
http://www.linti.unlp.edu.ar

Abstract. Este trabajo, derivado de la tesina de grado del alumno Gastón Traberg, plantea el desarrollo de la aplicación Swarming[1], la cual surge como una herramienta que encara de manera novedosa la automatización de algunas tareas que forman parte de la auditoría de seguridad de un servicio, mediante el descubrimiento de portales de autenticación y la obtención de credenciales de acceso mediante ataques de fuerza bruta, contemplando en este proceso los vacíos dejados por otras aplicaciones e incorporando tecnologías para las que existe escaso o ningún soporte.

Keywords: seguridad, auditoría, fuerza bruta, swarming, HTTP

1 Motivación

La utilización de ataques de fuerza bruta en el ámbito de las auditorías de sistemas de información, habitualmente presentan dos situaciones adversas:

- En cuanto a la modalidad, pues se realizan de una manera que desaprovecha gran parte de la información expuesta por los servicios que la red posee y que es relevante acerca de revelar cuáles son las conveniencias en la realización del ataque.
- En cuanto a la dificultad, ya que que se vuelven muy complejas en el momento de la configuración, sobre todo en servicios que ofrecen recursos de gran canttidad de alternativas en su utilización, como es el caso de HTTP.

En la práctica ocurre que la fuerza bruta durante una auditoría es poco practicada o se efectúa sin aprovechar su potencial, ya sea por cuestiones de tiempo o por las situaciones antes citadas, dejando así fuera del análisis importantes vulnerabilidades en los servicios que son objeto de auditoría.

2 Estado del arte

La Fuerza Bruta[2] es una modalidad de ataque mediante la cual se intenta obtener acceso no autorizado realizando intentos reiterados y sistemáticos de login sobre un servicio de red, a través de la utilización de credenciales potencialmente válidas. Las credenciales utilizadas durante la realización de este tipo de ataques, suelen provenir de conjuntos de credenciales que existen por defecto para ciertos dispositivos o aplicaciones, de credenciales comúnmente utilizadas por los usuarios o de credenciales que fueron comprometidas previamente en otros ataques[3]. En el contexto de las auditorías de seguridad, mediante la fuerza bruta se detectan todos o la mayoría de los casos en los cuales las credenciales utilizadas para el acceso a un servicio presentan alguna debilidad, las cuales potencialmente permitirían a un atacante acceder de forma no autorizada al sistema de información que está siendo evaluado.

3 Introducción

Entre las tres herramientas más destacadas que se utilizan en la comunidad de la seguridad informática para realizar ataques de fuerza bruta podemos mencionar: Hydra[4], Medusa[5] y Ncrack[6].

Sobre estas opciones es que se construyó el análisis que expone ventajas y limitaciones de cada una, constituyendo la base del desarrollo planteado.

Hydra es una de las aplicaciones de fuerza bruta más antiguas que existen en el contexto de la seguridad siendo desarrollada en sus inicios por el Alemán van Hauser en 2001. Hydra es Software Libre, y es popularmente, la primera opción al momento de hablar de herramientas de fuerza bruta sobre redes. Esta aplicación es activamente mantenida, habiendo sido actualizada la rama estable por última vez el 12 Mayo del 2014, mientras continúa su desarrollo utilizando github[7] como repositorio de código fuente. Entre las ventajas que se pueden destacar de Hydra se destaca:

- el gran número de protocolos que soporta para atacar
- el número de plataformas sobre las que se indica que funciona.

Por otro lado Hydra posee algunas características que le restan popularidad al momento de utilizar o de extender en funcionalidad, entre las que se destacan:

- Suele suceder que la aplicación se corrompe durante su ejecución u omite resultados producto de los errores que se generan. Y si bien algunos de estos problemas se fueron solucionando con el tiempo, otros aún persisten.
- Al estar implementada en lenguaje C y con un estilo de desarrollo muy particular de los autores del código, es difícil tener un entendimiento claro de lo que ocurre dentro de la aplicación, dificultando mucho la realización de extensiones por parte de gente que no forme parte del equipo principal de desarrolladores del proyecto.

Realizando estas mismas observaciones, es que miembros del grupo Foofus.net se avocaron al desarrollo de Medusa, subsanando de en este nuevo producto la mayoría de los problemas que Hydra presentaba. Comparándolos:

- Medusa posee un código fuente mucho más organizado y prolijo, además de otras características que lo hacen una interesante opción,
- Pero a diferencia de Hydra, Medusa tiene soporte sobre un número algo menor de protocolos .

Como tercera opción popular existe nCrack que es un proyecto nacido en el "Google Summer of Code" de 2009 pero que aún no es considerado como un programa completo ni terminado ni siquiera por los propios autores. En conclusión es válido decir que no es una opción muy competitiva contra las dos primeras:

- en principio porque no ofrece nuevas funcionalidades
- y en segundo lugar su desempeño es inferior con respecto a Hydra y a Medusa.

Concluyendo podemos decir que estas tres herramientas en conjunto poseen soporte para una gran cantidad de protocolos, buena velocidad en la realización de ataques de fuerza bruta y relativa simplicidad de uso[8][9].

Sin embargo, más allá de estas características positivas, lo que hace que estas herramientas sean la primer opción al momento de hablar de fuerza bruta sobre redes, es la inexistencia de alternativas completas. Esto se comprueba mediante la realización de simples casos de uso de estas aplicaciones sobre redes y servicios comúnmente encontrados hoy en día en cualquier red, quedando rápidamente en evidencia los muchos problemas que éstas presentan a lo largo de su utilización. Pudiendo mencionar:

- De existir un Firewall como iptables[10] o un IPS como fail2ban[11], los servicios solo se pueden acceder de a intervalos de tiempo regulares o un cierto número de veces por cantidad de tiempo. Esto genera situaciones que no siempre son bien manejadas por las herramientas, provocando que estas ignoren algunas credenciales o que las mismas sean rechazadas, dejando en desconocimiento si dichas credenciales son válidas o no.
- Existen protocolos como HTTP que presentan una gran complejidad en los procesos de login, ya que por ejemplo suelen utilizar para el manejo de dichos procesos Cookies de sesión o Tokens anti CSRF[12]. Estos mecanismos que no se encuentran considerados y soportados en forma completa en las en las herramientas antes mencionadas, sino que, o se lo hace de manera parcial o directamente ni se los considera. Esta ausencia de soporte lográ que para ellas HTTP sea un ejemplo de protocolo que queda excluido, aunque sea extremadamente utilizado hoy en día.
- Siguiendo con HTTP, a diferencia de otros servicios, en la actualidad este tiene un rol extremadamente destacado dentro de una red, en muchos de los aspectos posibles. Se ha vuelto un servicio extremadamente complejo, y aparejado a esa complejidad creció el número de potenciales problemas de seguridad existentes. Todas las herramientas mencionadas se limitan a

auditar sólo lo explícitamente indicado por su ejecutor, generando que la auditoría de un servicio con estas características sea una tarea imposible, debido a la enorme cantidad de contenido a analizar antes de poder generar una ejecución válida de las mismas.
- La velocidad con la que se realiza un número de intentos de login no es relevante al momento de un ataque. Este objetivo parece ser el buscado en principio por las aplicaciones listadas, no siendo el caso de la propuesta de Swarming. Si algo deja como experiencia la auditoría de redes y sus servicios es que en la mayoría de los casos, las credenciales que uno busca están al alcance de la mano o suelen ser de fácil inferencia, quedando la velocidad de ataque rezagada a un segundo plano ya que el problema radica en la obtención de las credenciales y no en la velocidad con la que se las utiliza.

Si bien se mencionan las expuestas anteriormente como las mas destacables, existen infinidad de características más que hacen de estas tres herramientas algo poco práctico al momento de hablar de procesos de fuerza bruta, más cuando se trata de auditorías de redes grandes.

4 Swarming

4.1 Los objetivos planteados

Partiendo de todos los problemas anteriormente analizados como así también de experiencias e ideas propias y sugeridas, es que se elaboró una lista de requerimientos que sirvieron como base para el diseño de la aplicación solución, entre los cuales se destacan:

- Realizar procesos de fuerza bruta, también conocidos como Cracking.
- Realizar un recorrido automático del servicio auditado, proceso conocido como Crawling.
- Crecer dinámicamente en el número de procesos que realizan los trabajos.
- Reconocer aplicaciones, dispositivos y cualquier característica que signifique un ahorro de trabajo en la auditoría.
- Reutilizar lo descubierto, iniciando de manera automática procesos de Crawling sobre recursos a los que se tuvo acceso, repitiendo el ciclo todas las veces que sea posible.
- Brindar una interfaz de usuario simple, basada en una API que permita la interacción de otras herramienta con la aplicación de manera sencilla.
- Seguir un modelo en el cual cada módulo encargado de procesar un protocolo en particular, funcione de la manera más aislada posible, permitiendo la carga y descarga de los mismos además de una fácil y flexible implementación.

De esta manera se establece un escenario que a futuro facilitará la tarea de agregar nuevas funciones, entre otras, los protocolos a soportar.

Si bien es imprescindible la implementación de la mayor cantidad de protocolos posible para un aprovechamiento máximo de la capacidad de la aplicación, en la primer instancia del desarrollo se apuntó a HTTP como primer protocolo a

contemplar. Esta decisión se justifica porque HTTP es uno de los protocolos con peor soporte por parte de las aplicaciones antes mencionadas, además de ser el protocolo que más problemas puede presentar debido a su complejidad y al gran número de diferentes situaciones existentes en su utilización.

4.2 Características del desarrollo

Swarming nace como una aplicación cuyo objetivo principal es el desarrollo automático de todas las actividades posibles en torno a los procesos de fuerza bruta, buscando con esto aprovechar al máximo toda la información obtenida del contexto, sin que se requiera intervención humana.

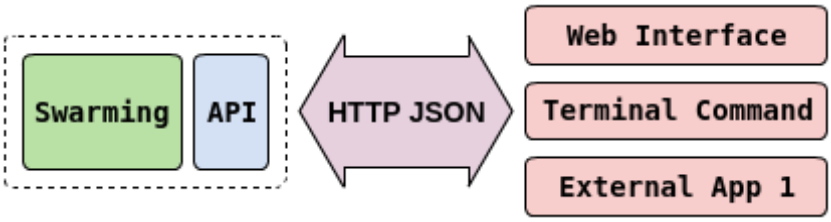


Fig. 1. Arquitectura de SWARMING

Para lograr esto, se siguió un modelo que presenta una única interfaz a través de la cual no solo el usuario controla la aplicación, sino que también permite que otras aplicaciones puedan acceder a los datos y controles, logrando de esta manera una fácil extensión en el uso del Swarming por otros programas. La figura 1 muestra una representación general de cómo el único medio de acceso, ya sea desde una interfaz de usuario o desde una aplicación externa, es mediante la API del programa, la cual es accedida a través del envío de mensajes en formato JSON[13].

Para la implementación, si bien existe un espectro interesante de lenguajes, se optó por la utilización del lenguaje de programación Python[14] en su versión 3, el cual presenta muchas cualidades dentro de las que podemos destacar:

- Lenguaje Interpretado. Esta característica es una gran ventaja al momento de desarrollar, ya que las pruebas pueden realizarse inmediatamente después que los cambios fueron realizados. Si bien el hecho de que sea un lenguaje interpretado no es óptimo con respecto a la performance, existe en Python la posibilidad de realizar módulos en algún lenguaje compilado (C++ por ejemplo), los cuales pueden ser importados y utilizados desde Python, característica que será aprovechada para dar soporte a protocolos como MySQL, SSH2, y otros.
- Soporte nativo de Unicode. Hoy en día, la capacidad de soportar la codificación Unicode en las comunicaciones es más un requerimiento que una opción. Es

por esto que se optó por Python 3 para la implementación del Swarming, ya que Python en sus versiones anteriores, como la 2.7, no soporta Unicode de manera simple.
– Gran Número de Librerías existentes que se pueden utilizar. Algo que siempre caracterizó a Python es el hecho de que posee un espectro muy interesante de librerías para distintas actividades, desde ORMs para el modelado de bases de datos siguiendo una orientación a objetos, pasando por Frameworks para el desarrollo de interfaces Web y utilización de JSON, hasta módulos para el manejo de protocolos como HTTP de forma muy simple pero no por eso menos potente.

5 Experiencia realizada

Para evaluar la implementación actual de SWARMING, se configuró un servidor WEB con el objeto de simular ser el portal de una organización real, accesible vía: http://www.miportal.unlp.edu.ar/.

Existe en el portal una sección http://www.miportal.unlp.edu.ar/admin/ cuyo ingreso está restringido con una autenticación tipo HTTP Basic, y que al autenticarse exitosamente permite acceder a un PHPMyAdmin, el cuál posee una autenticación propia que utiliza un formulario HTTP.

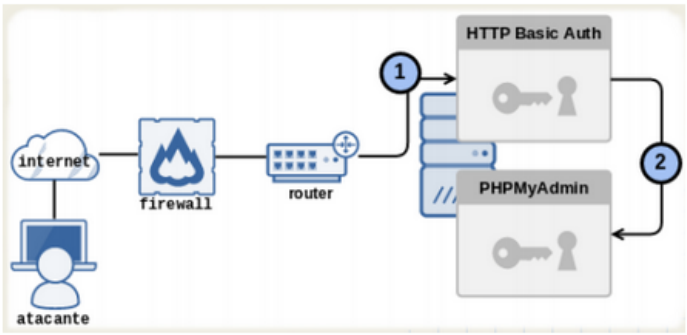


Fig. 2. Entorno de prueba

Se ejecutó la herramienta especificando la url del sitio web y los diccionarios de usuarios y contraseñas a utilizar a fin de:

– Descubrir posibles portales de autenticación como ser: HTTP Basic, HTTP Digest, Formularios HTML u otro.
– Ejecutar ataques de fuerza bruta sobre los mismos para una posterior evaluación de la información encontrada en los mismo.

Como resultado de la etapa (1), SWARMING encontró el portal *http://www.miportal.unlp.edu.ar/admin/* el cual estaba protegido con autenticación HTTP tipo Basic y creó una “tarea”

para la ejecución del ataque de fuerza bruta utilizando los diccionarios previamente cargados (etapa 2).

Finalizada la etapa (2), en caso de haber logrado el acceso SWARMING se encarga de iniciar nuevamente el ciclo de ejecución (descubrimiento/ataque) dentro de la sección a la que antes no se tenía acceso.

En un segundo ciclo de ejecución SWARMING detectó la aplicación WEB PHPMyAdmin, la cual restringe el acceso mediante un formulario HTML sobre el cual luego de descubierto se inicia nuevamente el ataque.

Para lograr lo anteriormente descripto, antes de la implementación de SWARMING hubiera sido necesario que el analista de seguridad realizara en forma manual:

- El descubrimiento de portales de autenticación utilizando alguna herramienta adicional que permita enumerar los recursos en la red, no sólo una herramienta de fuerza bruta, por ejemplo usando Burp/Firefox[15].
- Una vez descubierto el portal si realizar la ejecución del ataque de fuerza bruta, usando alguna de las herramientas preexistentes.

Dicho proceso el analista debería volver a iniciarlo de forma manual, luego de obtener algún nuevo acceso. En cambio SWARMING, automatiza lo anteriormente descrito utilizando unos pocos pasos de configuración los cuales implican:

- Iniciar la aplicación
- Cargar los diccionarios
- Especificar la URL

Una de las ventajas que vale la pena destacar de SWARMING es que permite interrumpir el proceso de auditoría en cualquier momento, como si se estuviera pausando, ya que luego va a continuar a partir del punto en que fue interrumpido sin necesidad de arrancar desde cero. Otra aspecto interesante es que a medida que avanza el análisis el usuario puede acceder a resultados parciales, ya sea a la información sobre portales de autenticación encontrados o accesos a los mismos logrados sin necesidad de que la herramienta finalice completamente su ejecución.

6 Conclusiones

Mediante lo experimentado se pudo observar el grado de automatización que el Swarming ofrece, no sólo en el descubrimiento de recursos a auditar, sino también retroalimentandose a partir de la generación de nuevas tareas a desarrollar sobre el servicio a partir de lo parcialmente descubierto, con el fin de lograr el máximo nivel de acceso posible.

De haberse ejecutado a través de aplicaciones como Hydra o Medusa cada etapa analizada a lo largo de la ejecución del Swarming, hubiese resultado en una repetida ejecución y análisis de resultados de forma manual, desperdiciando así todo el tiempo que la buena performance de estas herramientas podrían llegar a ofrecer al usuario.

Esta aplicación logra sus objetivos al implementar de forma completa mecanismos de autenticación HTTP, brindando un soporte robusto de formularios HTML y manejo de sesiones HTTP. Si bien la herramienta aún no presenta todas las capacidades que se pretenden, está diseñada de manera extensible permitiendo en el futuro la incorporación de nuevas características o funcionalidades.

El desarrollo de soporte de otros protocolos se propone como trabajo a futuro.

References

1. Swarming - Repositorio GIT de código del proyecto https://github.com/samelat/swarming/
2. Penetration Testing and Network Defense by Andrew Whitaker,Daniel P. Newman - Capítulo 9 - Página 209 - Brute Force Attacks - Cisco Press
3. Hacking: The Next Generation by Nitesh Dhanjani, Billy Rios & Brett Hardin - Capítulo 3 - Página 74 - Brute-Forcing Your Way In.
4. Hydra - Sitio Oficial: https://www.thc.org/thc-hydra/
5. Medusa - Sitio Oficial: http://foofus.net/goons/jmk/medusa/medusa.html
6. nCrack - Sitio Oficial: https://nmap.org/ncrack/
7. https://github.com/vanhauser-thc/thc-hydra
8. Análisis de performance comparativo - Realizado por el proyecto Hydra: https://www.thc.org/thc-hydra/network_password_cracker_comparison.html
9. Análisis de performance comparativo - Realizado por el proyecto Medusa: http://foofus.net/goons/jmk/medusa/medusa-compare.html
10. The netfilter.org project - IPTABLES http://www.netfilter.org/projects/iptables/index.html
11. Fail2ban HomePage http://www.fail2ban.org/wiki/index.php/Main_Page
12. Cross-Site Request Forgery. Web Security: A WhiteHat Perspective by Hanqing Wu y Liz Zhao. Capítulo 4, pag. 123.
13. Ajax, Data Formats, JSON. High Performance JavaScript by Nicholas C. Zakas - Capítulo 7, pag. 138.
14. Mark Lutz. Learning Python. 5ta edición.
15. BURP Proxy - https://portswigger.net/burp/proxy.html

IV Workshop de Innovación en Educación en Informática (WIEI)

7405 | Sistemas de información: Competencias profesionales 2020
Marisa Tumino (UAP), Juan Manuel Bournissen (UAP) y Karen Barrios (UAP)

7552 | Promoción del ritmo de estudio por feedback colectivo de progreso en trabajos prácticos
Diego Fontdevila (UNTREF), Mariano Tugnarelli (UNTREF), Sebastian Ismael (UNTREF) y Lucas Videla (UNTREF)

7571 | Estrategias Didácticas Innovadoras en la enseñanza de TICs para Ingeniería en Informática
Alicia Laura Mon (UNLaM), Horacio Del Giorgio (UNLaM) y Bettina Donadello (UNLaM)

7584 | Experiencia en la construcción de Objetos de Aprendizaje para árboles AVL usando CROA
Catalina Mostaccio (UNLP) y Gabriela A. Pérez (UNLP)

7605 | Requerimientos de competencias en el mercado laboral actual
Claudia Russo (UNNOBA), Monica C. Sarobe (UNNOBA), Marcela Dillon (UA), Fernando López Gil (UNNOBA), Martín Calcaterra (UNNOBA), Pedro Ochipinti (UNNOBA) y Hugo Ramón (UNNOBA)

7424 | Entorno para Experimentación de Vulnerabilidades en la Enseñanza de Buenas Prácticas de Programación
Yael Uri (UNA), Benjamín Barán (UNA)

7626 | Pedagogical Practices with Social Impact in the Computer Science School of the National University of La Plata: Interdisciplinary Articulation for the Development of a Digital Communicator for Autistic Children
Claudia Queiruga (UNLP), Ivana Harari (UNLP) y Jorge Rosso (UNLP)

7451 | Formación de docentes y alumnos en el desarrollo de actividades de programación colaborativa en línea
Alejandro González (UNLP), Beatriz Depetris (UNTDF), Maria Cristina Madoz (UNLP), Daniel Aguil Mallea (UNTDF)

7495 | Caracterización de alumnos de una materia mediante datos multivariados obtenidos de un cuestionario de satisfacción
Myriam Herrera (UNSJ), María Inés Lund (UNSJ), Laura Aballay (UNSJ)

7523 | Programación Estructurada en un Curso Introductorio. Una Experiencia Explorando Python
Edith Lovos (UNRN), Tatiana Gibelli (UNRN), Rodolfo Bertone (UNLP)

7661 | Incorporando conceptos en la enseñanza de Concurrencia y Paralelismo utilizando el entorno CMRE
Laura De Giusti (UNLP), Fabiana Leibovich (UNLP), Franco Chichizola (UNLP), Marcelo Naiouf (UNLP), Armando E. De Giusti (UNLP)

IV Workshop de Innovación en Educación en Informática (WIEI)

7558 | Propuesta didáctica innovadora para el aprendizaje de los sistemas de información administrativa

Laura Rita Villarrubia (UNJu), Lia Gabriela Rico (UNJu)

7569 | Simuladores Aplicados en laboratorios de Energía Nuclear

Lucia Pérez Lucero (UNC), Norma Adriana Chautemps (UNC), Rodolfo Bertone (UNLP), Laura Cecilia Díaz (UNC)

7594 | Enseñanza Universitaria. Convocatoria a los docentes de la Facultad de Informática de la UNLP

Armando E. De Giusti (UNLP), Maria Alejandra Zangara (UNLP), Cecilia Sanz (UNLP), Lucrecia Moralejo (UNLP), Fernanda Barranquero (UNLP), Marcelo Naiouf (UNLP)

7627 | Una Comparativa del Algoritmo Particle Swarm Optimization con Restricciones entre los Métodos de Factor de Constricción y Neighborhood con Coeficiente de Inercia Dinámico

Miguel Augusto Azar (UNJu), Fabiola Paz (UNJu), Analía Herrera Cognetta (UNJu)

7653 | Enseñando a programar con RITA en escuelas secundarias

Vanessa Aybar Rosales (UNLP), Claudia Queiruga (UNLP), Isabel Miyuki Kimura (UNLP), Matías Brown (UNLP) y Soledad Gomez (UNLP)

Sistemas de información:
Competencias profesionales 2020

Marisa Cecilia Tumino, Juan Manuel Bournissen y Karen Barrios
Universidad Adventista del Plata, Libertador San Martín, Entre Ríos, Argentina
{marisatumino, juanbournissen}@doc.uap.edu.ar

Resumen. En el trabajo se propuso efectuar una búsqueda de las competencias profesionales vigentes en carreras afines de instituciones de reconocida trayectoria, compilando aquellas más representativas del desempeño profesional en esta disciplina. Se analizaron las competencias seleccionadas a la luz de los objetivos de la carrera con el propósito de someterlo al juicio de empleadores, docentes, estudiantes y egresados. Posteriormente se definieron las competencias profesionales que reunieron los requisitos identificados en el estudio y se extrajeron sus conclusiones.

Palabras clave: Competencias profesionales, Formación por competencias, perfil profesional, sistemas de información.

1. Introducción

En la actualidad los currículos en su gran mayoría no poseen la flexibilidad necesaria para profundizar algunos temas o para incluir otros que enriquezcan la vocación del estudiante con opciones de estudio, oportunidades de investigación y de ejercicio profesional. De acuerdo con lo que señalan Duk y Loren [1]

> *"El currículum oficial de un país proyecta la visión de futuro de la sociedad y sus aspiraciones para con las nuevas generaciones; asimismo concreta las finalidades de la educación a través de la selección de las competencias que permitan a las personas desarrollarse y participar en las distintas esferas de la vida."*

En el proyecto Tuning América Latina [2], se señala que obtener un perfil de egreso para el área informática tiene dificultades especiales, pues el origen reciente de la informática y la rápida evolución de las tecnologías asociadas hace que los conocimientos y técnicas que la sustentan estén en permanente evolución. Por otro lado el uso de la informática no cesa de crecer en prácticamente todos los ámbitos del quehacer humano, generando una gran cantidad y variedad de escenarios para el ejercicio de la profesión.

Tal como lo señala el Departamento de Ingeniería en Sistemas de la Universidad de Antioquia [3], el mundo es cada vez más demandante de medios avanzados para crear, difundir, adaptar y usar el conocimiento como pilares para la generación de riqueza. Por ende, las sociedades deben educarse para permitir la diversificación de la formación de sus integrantes en aras de responder oportuna y eficazmente a las demandas del creciente volumen de conocimiento.

El documento mencionado plantea la importancia del funcionamiento del programa de sistemas de información sustentado en un modelo pedagógico de tipo investigativo

y centrado en los tres ejes conceptuales del ser, del saber y del hacer, orientado a la solución de problemas y al logro de competencias. Este modelo pedagógico concibe a la educación como un subsistema esencial del sistema social, en tanto que forma integralmente a los individuos para interactuar en dicho sistema. La formación integral conlleva tres dimensiones: la formación como persona, el desarrollo intelectual y la capacidad de intervenir de manera competente, responsable y solidaria a las realidades sociales.

La formación en estas dimensiones es soportada por un currículo centrado en la solución de problemas y orientada al desarrollo de las competencias para intervenir en las realidades. Unas competencias son desarrolladas por áreas profesionales que utilizan conocimientos especializados para resolver problemas específicos. Otras competencias son generadas por áreas básicas que soportan las competencias de las áreas profesionales y contribuyen al desarrollo de destrezas y estrategias para comprender las realidades y formular y resolver problemas.

En este sentido Garcia Almiñana, Sánchez Carracedo, y Gavaldà Mestre [4] clasifican las competencias profesionales de una titulación en tres categorías: (a) competencias técnicas, relativas a la titulación, (b) competencias transversales, necesarias para ese nivel académico pero no relacionadas con los conocimientos técnicos de la titulación y (c) competencias deontológicas, relacionadas con la formación personal social. A los fines prácticos de este trabajo, se han considerado las competencias técnicas, no sin descuidar aquellos aspectos de las competencias transversales y ontológicas que agregan valor natural a las competencias técnicas.

Fuente, De Andrés, Nieto, Suárez, Pérez, Cernuda, Luengo, Riesco, Martínez, Lanvín y Fondón [5], señalan que uno de los objetivos propuestos en el marco del Espacio Europeo de Educación Superior (EEES) consiste en acercar de manera real la preparación obtenida en el entorno universitario a las necesidades profesionales que la sociedad demanda, por lo que se propone tener muy en cuenta las competencias profesionales a la hora de diseñar las nuevas titulaciones. Sin embargo lo cierto es que en la actualidad muchas de las titulaciones que se imparten en las universidades europeas no cumplen este requisito, manteniendo una distancia importante entre el perfil del egresado y el perfil profesional que demanda la sociedad.

Para avanzar en la integración universidad-empresa se han elaborado los libros blancos de ANECA, España, donde un comité de expertos ha definido el conjunto de competencias que constituyen los perfiles profesionales. Como resultado del estudio crítico de las propuestas presentadas en el libro blanco (Agencia Nacional de Evaluación de la Calidad y Acreditación) [6] , Fuente et. All desarrollaron el “libro azul” como alternativa que surge de la redefinición de perfiles y competencias disciplinares.

Openlibra [7] confeccionó un manual para la formación en competencias informáticas e informacionales que fue también una fuente valiosa para la elaboración de la propuesta, así como también la contribución de las competencias TIC en información de Romeu Fontanillas [8].

Estas experiencias se han emprendido considerando tanto los ámbitos académicos como profesionales a fin de facilitar esta integración.

Siendo que estos libros han surgido de las condiciones emergentes en el EEES, se considera oportuno contextualizar la propuesta adaptada a la realidad argentina vigente en proyección al 2020.

2. Definición del problema

Dada la necesidad de especificar las competencias profesionales Informáticos adaptadas al contexto histórico-geográfico de Argentina, se propone en este trabajo definir las competencias que emerjan de los perfiles profesiones vigentes para someterlas a la valoración de los colectivos que tienen incumbencia real en esta definición.

3. Justificación científica, académica-institucional y social

En los últimos años en América Latina se estuvo consolidando la mirada a la formación profesional por competencias dado el auge que ha tenido en los países europeos a partir del Proceso de Bolonia con la construcción del Espacio Europeo de Educación Superior (EEES). Esta tendencia no se ha manifestado con suficiente firmeza en las carreras de informática de los países latinoamericanos, manteniendo la distancia observada entre el perfil de egresado para el que forma la universidad y el perfil profesional que impone la demanda social. Recientemente en Argentina se ha expresado la voluntad de definir las competencias con la intención de instaurar el concepto en esta disciplina.
La UAP se encuentra preparando camino al respecto y se espera comenzar a diseñar planes de estudios y programas de asignaturas desde un enfoque de formación profesional por competencias.
La integración del concepto de competencia, que demanda evidencia del desarrollo holístico del estudiante desde la asimilación del conocimiento, reflejado en habilidades y actitudes, concuerda con la intención de las instituciones educativas de formar profesionales capaces de afrontar los desafíos que los avances de la ciencia y la tecnología imprimen en la comunidad organizacional del siglo XXI. Por esto se considera importante llegar a un consenso en la definición de las competencias que atiendan a estas tendencias.

4. Metodología

Los pasos seguidos se resumen a continuación:
1. Relevamiento. Durante el inicio del estudio se propusieron las competencias, basándose en el sondeo y la compilación de las tendencias y planes de estudio de otras universidades nacionales y latinoamericanas, analizando el perfil de profesionales informáticos que demandan las organizaciones y los perfiles de egresados informáticos de las carreras afines a la profesión. Asimismo se contemplaron las actividades reservadas del informático consideradas en la Res ME Nº 786/09 – Estándares de acreditación de los títulos de Licenciado en Ciencias de la Computación, Licenciado en Sistemas/Sistemas de Información/Análisis de Sistemas, Licenciado en Informática, Ingeniero en Computación e Ingeniero en Sistemas de

Información/Informática [9]. Se efectuó un análisis de los datos y se propusieron las competencias en forma preliminar.
2. Valoración de la propuesta en conjunto de los colectivos. Seguidamente los participantes del estudio las "Competencias del Informático 2020", incluyendo docentes, empleadores y graduados de las carreras de informática de diferentes facultades del país, analizaron la suficiencia de las competencias reflejadas en el listado, agregaron aquellas que a su juicio se hubieran omitido, eliminaron aquellas que consideraron repetidas o innecesarias y modificaron aquellas que ameritaban un cambio.
3. Ajuste y definición de las competencias del profesional informático a parir de las sugerencias recibidas de los participantes. Luego de recibir las sugerencias, analizarlas a la luz de los perfiles profesionales, e introducir las modificaciones correspondientes, se confeccionó el nuevo listado de las competencias del informático 2020. Seguidamente se confeccionó la encuesta en línea, utilizando una escala de respuestas que representa una evaluación en continuo desde una relevancia nula (0) hasta una relevancia óptima (7) para cada una de las competencias, organizadas por áreas, cada una de ellas con un conjunto de indicadores o competencias que varía en número de acuerdo al área.
4. Una vez confeccionado el instrumento, se les solicitó a los diferentes colectivos (docentes, empleadores, graduados y estudiantes) la evaluación del nivel de relevancia de las competencias propuestas para la formación profesional de sistemas de información. Se obtuvo la valoración en cuanto al grado de relevancia de cada una de las competencias definidas.

5. Procesamiento y análisis de datos

El análisis de los resultados se adecuó a la metodología utilizada por Maldonado Rojas (2007: 235-237) [10]. La escala de respuestas se codificó según el esquema que muestra la Tabla 1. Para ello se utilizaron las frecuencias observadas en cada valor de la escala. El criterio para analizar los resultados obtenidos se sustentó en las categorías interpretativas diseñadas a partir de la escala de cuartiles y considerando los porcentajes obtenidos sólo en el nivel Alto, luego de realizada la codificación en base a la escala de respuestas mostrada en la Tabla 1. Las categorías finales se muestran en la Tabla 2.

Tabla 1: Escala de respuesta y su codificación

Escala	Grado
0	Bajo
1	
2	
3	Medio
4	
5	Alto
6	
7	

Las competencias que formaron parte de la propuesta final fueron aquellas que se ubicaron en las categorías interpretativas iguales o superiores al nivel "Satisfactorio".

El estudio describe el porcentaje de sujetos que adhiere a cada una de las opciones de la escala.
Se espera que como resultado de esta iniciativa, cada institución participante discrimine las competencias propuestas entre las carreras de sistemas, incluyendo las licenciaturas en sistemas, licenciaturas en computación, licenciaturas en informática, ingenierías en sistemas e ingenierías en computación.

Tabla 2: Categorías interpretativas utilizadas como criterios de análisis

% de sujetos que valora la relevancia de la competencia en nivel Alto	Categoría interpretativa
Menor o igual que 25%	Deficiente
Mayor que 25% y menor o igual a 50%	Medianamente satisfactorio
Mayor a 50% y menor o igual a 75%	Satisfactorio
Mayor del 75%	Óptimo

6. Resultados

Es importante destacar que los expertos que participaron en el estudio reunían las condiciones descritas en las Tablas 3 y 4, en relación con el rol que desempañaban y la experiencia en su ejercicio profesional. Como se advierte el 59% de los involucrados contaba con más de 10 años de experiencia, lo que permite calificar los resultados obtenidos como información confiable.
Con respecto al rol desempeñado, se advierte que, dado que este campo permitía respuestas múltiples, el porcentaje total supera el 100% ya que algunos de los encuestados cumplían más de un rol; lo que se refleja en el 54% correspondiente a los graduados de informática, valor esperable para el cumplimiento de las funciones propias de la profesión.

Tabla 3: Distribución de roles de los participantes

Rol	Cant.	%
Empleador	7	17%
Graduado en informática	22	54%
Docente de informática	14	34%
Estudiante de informática	7	17%

Tabla 4: Distribución de tiempo de experiencia de los participantes

Tiempo de experiencia	Cant.	%
menos de 6 años	9	22%
6 a 10 años	7	17%
11 a 15 años	9	22%
16 a 20 años	4	10%
21 a 25 años	3	7%
más de 25 años	8	20%

Una vez compiladas las competencias profesionales del informático, obtenidas del sondeo efectuado sobre las universidades nacionales y latinoamericanas que incluyen en su oferta académica títulos afines, se procedió a someterlas a la examinación de los participantes del estudio, quienes evaluaron a partir de su experticia cada competencia según su adecuación u omisión, sugirieron modificaciones y añadieron aquellas que ameritaban su inclusión.

El listado obtenido fue reenviado a los diferentes colectivos, quienes valoraron la relevancia de cada una de las competencias incluidas en él. Se procesaron los datos recibidos con el propósito de obtener los porcentajes totales de individuos que puntuaron la relevancia de cada competencia en el rango comprendido entre los valores 5 y 7 de la escala que representa el grado Alto de relevancia, conforme a lo descrito en la Tabla 1. Dichos porcentajes se muestran en la Tabla 5, organizada en orden decreciente.

Tabla 5: Porcentajes sumativos del grado Alto de la escala de relevancia

	Competencias	%
1	Análisis de las características, funcionalidades y estructura de los Sistemas Operativos para la implementación de aplicaciones.	92,68%
2	Aseguramiento del sistema y los datos según las necesidades de uso y las condiciones de seguridad establecidas para prevenir fallos y ataques externos.	92,68%
3	Integración de equipos de comunicaciones en infraestructuras de redes, determinando la configuración para asegurar su conectividad.	90,24%
4	Planificación, diseño, organización, desarrollo, mantenimiento y dirección de proyectos de sistemas, servicios y aplicaciones en el ámbito de la informática, liderando su puesta en marcha, su mejora continua y valorando su impacto económico y social.	90,24%
5	Análisis, diseño, implementación, mantenimiento y utilización eficiente y segura de bases de datos adecuadas a las funcionalidades de aplicaciones informáticas.	90,24%
6	Dominio y aplicación de las herramientas necesarias para el desarrollo de software y administración de bases de datos.	90,24%
7	Servicio a sus semejantes y preparación para el liderazgo en la identificación de las necesidades de la comunidad y trabajo conjunto con el propósito de encontrar soluciones altruistas y solidarias, atendiendo a la responsabilidad social.	90,24%
8	Proactividad que impulsa el pleno desarrollo, sustentada en los valores universales, los derechos humanos, los valores culturales y democráticos, la responsabilidad medioambiental y el compromiso ético.	90,24%
9	Administración de usuarios de acuerdo a especificaciones determinadas para garantizar los accesos y la disponibilidad de los recursos del sistema.	87,80%
10	Asesoramiento y diseño de estrategias de seguridad de la información en redes de transmisión de datos.	87,80%
11	Evaluación y selección de plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas.	87,80%
12	Diseño y utilización eficiente de los tipos y estructuras de datos que más se adecuan a la resolución de un problema.	87,80%
13	Autonomía para investigar desde diversas fuentes confiables nuevos lenguajes o soluciones tendientes a responder las nuevas demandas profesionales.	87,80%
14	Administración de servicios de red (web, mensajería electrónica y transferencia de archivos, entre otros) e instalación y configuración del software en condiciones de calidad para responder a las necesidades de la organización.	85,37%
15	Administración apropiada de los recursos informáticos (almacenamiento, espacio, energía, dinero, tiempo), acorde a los compromisos asumidos con el cliente.	85,37%
16	Desarrollo de aplicativos para dispositivos móviles, conforme a las tecnologías y paradigmas vigentes.	85,37%
17	Diseño, desarrollo, evaluación y aseguramiento de la accesibilidad, ergonomía, usabilidad y seguridad de los sistemas, servicios y aplicaciones informáticas, así como de la información que gestionan, conforme a principios éticos y a la legislación y normativa vigente.	85,37%
18	Análisis, selección y aplicación de metodologías y ciclos de vida adecuados a las necesidades de la aplicación a construir.	85,37%
19	Desarrollo, mantenimiento y evaluación de servicios y sistemas software fiables y eficientes que satisfagan todos los requerimientos del usuario y cumplan normas de calidad, aplicando las buenas prácticas de la Ingeniería del Software.	85,37%
20	Integración responsable, colaborativa y respetuosa en equipos de desarrollo de software.	85,37%

21	Selección más adecuada del paradigma y de los lenguajes de programación en el análisis, diseño, construcción y mantenimiento de aplicaciones robustas, seguras y eficientes.	85,37%
22	Expresión clara de las ideas, con sustento en la reflexión e interpretación de temas relevantes de índole social, científica o ética, adaptadas a los problemas y soluciones que demanda la diversidad de opiniones y situaciones.	85,37%
23	Utilización del idioma inglés a fin de acceder a la información pertinente para la investigación y capacitación permanente.	85,37%
24	Análisis de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, fundamentos de su programación, y su aplicación para la resolución efectiva de problemas propios de la ingeniería.	82,93%
25	Selección, diseño, despliegue, integración y gestión de redes e infraestructuras de comunicaciones en una organización.	82,93%
26	Supervisión de la seguridad física según especificaciones y el plan de seguridad para evitar interrupciones en la prestación de servicios del sistema.	82,93%
27	Diseño y desarrollo de sistemas o arquitecturas informáticas centralizadas, o distribuidas, integrando hardware, software y redes de acuerdo con los conocimientos apropiados.	82,93%
28	Aplicación de estándares de seguridad, confidencialidad, integridad y privacidad, inherentes a los sistemas de información, dentro del marco de la legislación vigente, la normativa del colegio profesional correspondiente y las políticas de las organizaciones donde se desempeña.	82,93%
29	Selección, diseño, construcción, integración, administración y mantenimiento de sistemas de información que satisfagan las necesidades de la organización, según los criterios de costos, calidad e innovación tecnológica en todo el ciclo de vida del software.	82,93%
30	Utilización de herramientas de entornos de programación para crear y desarrollar aplicaciones.	82,93%
31	Análisis y valoración del impacto social y medioambiental del desarrollo de soluciones informáticas, atendiendo a las especificaciones, observando el cumplimiento de la legislación vigente y comprendiendo la responsabilidad ética y profesional de la actividad.	82,93%
32	Liderazgo en relaciones humanas, en negociación y resolución de conflictos, en hábitos de trabajo efectivo, valiéndose de las habilidades de comunicación y motivación en todos los ámbitos del desempeño laboral.	82,93%
33	Diseño e implementación de aplicaciones basadas en el análisis de las características, funcionalidades y estructura de los Sistemas Distribuidos, de las Redes de Computadores y de Internet.	80,49%
34	Diseño y evaluación de interfaz humano-computadora (HCI) que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.	80,49%
35	Diseño de soluciones informáticas, utilizando métodos de ingeniería del software que integren aspectos éticos, sociales, legales y económicos.	80,49%
36	Dominio adecuado de los aspectos de seguridad, privacidad y propiedad intelectual inherentes a los sistemas de información de una organización.	80,49%
37	Utilización de la argumentación y la prueba como medios para retener o rechazar conjeturas y avanzar hacia el camino de las conclusiones.	80,49%
38	Análisis de la organización y arquitectura de los computadores, así como los componentes básicos que los conforman.	78,05%
39	Conocimiento del funcionamiento de máquinas reales y virtuales que permitan incrementar la eficiencia de los algoritmos computacionales de los sistemas informáticos.	78,05%
40	Desarrollo de software orientado hacia la comunicación, según las normas de seguridad y confiabilidad en redes de transmisión de datos.	78,05%
41	Administración apropiada de los requerimientos y limitaciones del cliente, reconciliando objetivos en conflicto mediante la búsqueda de soluciones aceptables dentro de las restricciones derivadas del costo, del tiempo, de las tecnologías, de la existencia de sistemas ya desarrollados y de las propias organizaciones (factibilidad técnica-económica).	78,05%
42	Emprendedorismo innovador, autónomo, proactivo y creativo en pro de la resolución de problemas y la capitalización de nuevas oportunidades.	78,05%
43	Participación activa en la vida económica, social y cultural con actitud crítica y responsable.	78,05%
44	Evaluación de los criterios de compatibilidad hardware-software y software-software a fin de asegurar la correcta funcionalidad de los componentes.	75,61%
45	Desarrollo de nuevos métodos y tecnologías que contribuyan con versatilidad y adaptación a las nuevas situaciones generadas en el campo profesional.	75,61%

46	Gestión de las Tecnologías de la Información y Comunicación en los procesos empresariales que contribuyan con soluciones efectivas a las necesidades de información de las organizaciones, otorgándoles ventajas competitivas.	75,61%
47	Aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas, en el diseño de soluciones, analizando su idoneidad y complejidad.	75,61%
48	Desarrollo de soluciones algorítmicas que consideren las características de los equipos en los cuales se va a implementar.	75,61%
49	Elaboración de pliegos de condiciones técnicas de una instalación informática que cumpla con los estándares y normativas vigentes.	73,17%
50	Aplicación de los principios y técnicas fundamentales de los sistemas inteligentes.	73,17%
51	Planificación y desarrollo de cursos de actualización profesional y capacitación en general en sistemas de información para colegas, usuarios de sistemas y público en general.	73,17%
52	Obtención, formalización y representación del conocimiento humano, relacionado con aspectos de percepción y actuación, en una forma computable para la resolución de problemas mediante un sistema informático.	73,17%
53	Empleo apropiado de las metodologías de desarrollo centradas en el usuario, y en la organización, para la evaluación y gestión de aplicaciones y sistemas basados en tecnologías de la información que aseguren la accesibilidad, la ergonomía y la usabilidad de los sistemas.	70,73%
54	Planificación, organización, elaboración, desarrollo y firma de proyectos en el ámbito de la ingeniería del software con objeto de explotar sistemas, servicios y aplicaciones informáticos, gestionando las herramientas y técnicas disponibles, sus enfoques y paradigmas.	70,73%
55	Mediación entre las comunidades técnicas y de gestión de una organización, aplicando los principios y buenas prácticas de las comunicaciones organizacionales.	70,73%
56	Análisis y evaluación de arquitecturas de computadores, incluyendo plataformas paralelas y distribuidas, en proyectos de software que respondan a requerimientos particulares.	68,29%
57	Evaluación y optimización permanente del rendimiento del sistema, configurando los dispositivos hardware de acuerdo a los requisitos de funcionamiento.	68,29%
58	Diagnóstico responsable y solución eficiente de problemas técnicos en redes de transmisión de datos.	68,29%
59	Aplicación de los principios de organización, economía, gestión de recursos humanos, gestión de los riesgos, legislación y normalización en proyectos del ámbito informático.	68,29%
60	Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.	68,29%
61	Dominio de procedimientos y algoritmos matemáticos y conocimiento de cómo, cuándo y por qué usarlos de manera apropiada.	68,29%
62	Modelamiento adecuado de problemas específicos, bajo las restricciones impuestas por la relación software-hardware.	68,29%
63	Análisis objetivo del enfoque de la inteligencia artificial en la búsqueda de soluciones basadas en sistemas de control y robótica.	68,29%
64	Investigación científica en temas relacionados a los sistemas de información.	68,29%
65	Diseño, implementación y documentación de componentes encapsulados de programas	68,29%
66	Comprensión y evaluación de los componentes que conforman la computación de alto desempeño.	65,85%
67	Modelado, diseño y evaluación del comportamiento de redes de transmisión de datos, considerando el funcionamiento lógico de sus componentes activos.	65,85%
68	Administración de los recursos propios del área en función de las cargas de trabajo y el plan de mantenimiento.	63,41%
69	Planificación y ejecución eficiente de tareas de auditoría de los sistemas de información.	63,41%
70	Análisis objetivo del enfoque de la inteligencia artificial en la búsqueda de soluciones basadas en problemas clásicos y juegos inteligentes, sistemas expertos, redes neuronales y lógica difusa.	63,41%
71	Coordinación y colaboración en investigaciones que fortalezcan el desarrollo científico y tecnológico, divulgando los métodos y resultados de manera efectiva.	63,41%
72	Comprensión, aplicación y evaluación eficientes de los medios de almacenamiento físico.	60,98%
73	Optimización de los algoritmos computacionales de los sistemas informáticos, sustentada en la gestión de los recursos que utilizan las máquinas de estado finito.	60,98%
74	Tasación, peritaciones e informes de tareas o trabajos de informática, adecuadas a las necesidades.	60,98%

75	Comprensión y aplicación correcta de los principios de la gestión de riesgos en la elaboración y ejecución de programas de acción.	60,98%
76	Aplicación de herramientas y métodos de simulación en el modelado de diferentes tipos de sistemas informáticos.	60,98%
77	Conocimiento de los aspectos básicos de la teoría de máquinas de procesamiento de datos que permitan operar con propiedad los componentes software de un sistema informático.	53,66%
78	Conocimiento de los fundamentos de lenguajes de descripción de hardware y de la funcionalidad de los componentes hardware que otorguen mayor seguridad al momento de asesorar a los potenciales clientes.	53,66%
79	Interpretación y modelado de fenómenos a través de modelos matemáticos.	53,66%
80	Comprensión de los conceptos básicos de electricidad, magnetismo, electromagnetismo, ondas electromagnéticas campos eléctricos, magnéticos y electromagnéticos; concepto de electrónica (analógica y digital), circuitos y propiedades, semiconductores, dispositivos electrónicos y su aplicación apropiada en la resolución de problemas propios de la ingeniería.	51,22%
81	Construcción y representación de formulaciones matemáticas, aplicando métodos computacionales adaptados a los requerimientos específicos.	39,02%
82	Creación y depuración de programas simples, utilizando código ensamblador.	29,27%

El indicador que antecede a cada competencia, señala el área disciplinar a la que pertenece, tal como se detalla a continuación:
Arquitectura, Sistemas Operativos y Redes (ARSORE): 80, 24, 38, 66, 1, 33, 25, 3, 14, 26, 9, 68, 53, 56, 72, 77, 39, 78, 73, 44, 15, 57, 81, 67, 40, 10, 58 y 82
Ingeniería del Software, Bases de Datos y Sistemas de Información (ISBDSI): 4, 16, 17, 11, 27, 45, 74, 59, 54, 49, 28, 2, 29, 5, 6, 50, 18, 34, 19, 41, 46, 55, 69, 75, 76 y 20
Algoritmos y Lenguajes (AyL): 47, 12, 21, 60, 48, 30, 61 y 13
Aspectos Profesionales y Sociales (APyS): 22, 31, 35, 42, 36 y 51
Sistemas de control / inteligencia artificial: 52, 62, 70 y 63
Varios – Genéricas: 7, 32, 8, 23, 64, 65, 71, 79, 43 y 37n

7. Discusión y conclusión

La relevancia del presente proyecto radica en la valiosa contribución de los profesionales informáticos de experiencia que dedicaron tiempo y esfuerzo, durante las dos etapas del proceso de investigación, en pro de construir una propuesta curricular sustentada en la formación por competencias. Tal como se aprecia en la Tabla 5, cuarenta y ocho (48) competencias resultaron situadas en la categoría interpretativa Óptimo, dado que la suma de frecuencias pertenecientes a las puntuaciones altas, entre 5 y 7, representan un porcentaje superior al 75%.Por su parte, treinta y dos (32) competencias fueron apreciadas en la categoría interpretativa Satisfactorio, por quedar asociadas a un porcentaje comprendido entre 50% y 75%.Solo dos (2) competencias se encontraron relacionadas con un nivel Medianamente satisfactorio, ya que sus correspondientes porcentajes oscilaron entre 25% y 49%, por lo que se considera que del listado original, solo estas dos competencias quedan suprimidas y eliminadas.
Desde el enfoque investigativo adoptado por el estudio, se estima que el listado elaborado durante el proceso propuesto, en términos generales, ha resultado adecuado al perfil profesional del informático 2020, dado que el 97,6% de las competencias propuestas, es decir ochenta (80), se situaron en los niveles interpretativos de

relevancia Satisfactorio y Óptimo. A partir del listado definitivo compuesto por las 80 competencias profesionales, se espera contribuir con los diseños curriculares de las 5 terminales de las carreas de informática del sistema educativo argentino, o de aquellos correspondientes a las profesiones afines en otros países.

8. Referencias

1. C. Duk H. y C. Loren G., "Flexibilización del Currículum para Atender la Diversidad", Revista Latinoamericana de Educación Inclusiva, 4, 1 (2010): 187.
2. Tuning América Latina. Documento elaborado en la tercera reunión general del 2 al 4 de mayo de 2012 en Santiago de Chile; disponible en; http://www.tuningal.org /es/publicaciones/doc_download/71-documento-de-trabajo-de-la-reunion
3. Departamento de Ingeniería de Sistemas Universidad de Antioquia. "Transformación curricular programa Ingeniería de Sistemas". Medellín: Publicaciones Universidad de Antioquia (2006); disponible en; http://es.scribd.com/doc /54299048/Ingeniera-de-Sistemas
4. J. Garcia Almiñana, F. Sánchez Carracedo, y R. Gavaldà Mestre, "Recomendaciones para el diseño de una titulación de Grado en Informática", IEEE-RITA, 2, nº 2, (2007): 99
5. A. J. Fuente, J. De Andrés, C. Nieto, M. Suárez, J. R. Pérez, A. Cernuda, M. C. Luengo, M. Riesco, B. Martínez, D. F. Lanvín y M. D. Fondón, "El Libro Azul de La Ingeniería en Informática: una alternativa al Libro Blanco"; disponible en; http://di002.edv.uniovi.es/~cernuda/pubs/jide2005-c.pdf
6. Agencia Nacional de Evaluación de la Calidad y Acreditación. "Libro Blanco, Título de grado en Ingeniería Informática", 2004; disponible en; http://www.fic.udc.es/files/266 /libroblanco_informatica_0305.pdf; Internet (consultada el 07 de marzo de 2014).
7. Manual para la formación en competencias informáticas e informacionales (Madrid: Cardiff University, 2011), 160, disponible en; http://ci2.es/sites/default /files/documentacion/ manual_ci2_completo.pdf
8. Teresa Romeu Fontanillas, "Competencias TIC en Información y Documentación" (Catalunya, España: UOC, 2009).
9. Res ME Nº 786/09 – "Estándares de acreditación de los títulos de Licenciado en Ciencias de la Computación, Licenciado en Sistemas/Sistemas de Información/Análisis de Sistemas, Licenciado en Informática, Ingeniero en Computación e Ingeniero en Sistemas de Información/Informática"; disponible en; http://www.coneau.edu.ar/archivos /Res786_09.pdf
10. Mónica Maldonado Rojas, "Valoración de la formación recibida usando un perfil de referencia basado en competencias profesionales", Educación Médica, 10, nº 4, (2007): 233-243.

Promoción del ritmo de estudio por feedback colectivo de progreso en trabajos prácticos

Diego Fontdevila, Mariano Tugnarelli, Sebastián Ismael, Lucas Videla
Universidad Nacional de Tres de Febrero, Valentín Gómez 4752,
Caseros B1678ABH, Buenos Aires, Argentina
{dfontdevila, mtugnarelli, sismael, lvidela}@untref.edu.ar

Abstract. La mejora en el ritmo de estudio es una de las preocupaciones centrales de los docentes, en particular en ciencias básicas y computación. En particular, el desarrollo de trabajos prácticos dentro y fuera del aula es considerado una actividad clave de aprendizaje. En este trabajo, describimos un estudio exploratorio realizado en un curso introductorio de programación, en el cual se hizo visible cuántos ejercicios llevaban resueltos los alumnos, clase a clase, para la guía correspondiente. Los resultados muestran significativas mejoras relativas (comparando guía inicial con guía final) a medida que avanza el curso, en la cantidad de ejercicios resueltos. También realizamos una contrastación de los datos recogidos con los resultados del examen al final del período estudiado. En el cierre, describimos el alcance del estudio, los mecanismos para repetirlo y perspectivas de futuros trabajos.

Keywords: feedback, ritmo de estudio, mejora, guía de ejercicios.

1 Introducción

Como docentes nos encontramos muchas veces ante el desafío de promover las actividades de aprendizaje más allá de las realizadas en clase, y a veces con la frustración de no lograrlo. En particular, las guías de trabajos prácticos en los estudios universitarios, tan comunes y necesarias en las materias de ciencias básicas y técnicas, presentan serias dificultadas para los alumnos. Los docentes de la cátedra de Algoritmos y Programación I de la carrera de Ingeniería en Computación de la Universidad Nacional de Tres de Febrero nos propusimos realizar una experiencia para promover, y al mismo tiempo medir, el progreso en la realización de guías de trabajos prácticos por parte de nuestros alumnos.

Este trabajo es parte de nuestro proyecto de investigación aplicada sobre innovación en la enseñanza de la construcción de software.

1.1 El contexto de la carrera

La experiencia fue realizada en el marco del dictado de la materia Algoritmos y Programación I, que es la primera materia de programación y corresponde al primer cuatrimestre de la carrera.

La materia se dicta durante la segunda mitad del año ya que en la primera mitad los alumnos ingresantes participan del curso de ingreso. Como nuestros alumnos son nuevos ingresantes a la Universidad, en su mayoría no tienen experiencia universitaria previa y terminaron sus estudios secundarios el año anterior. Esto propone una dinámica particular entre docentes y alumnos ya que el desafío pedagógico no es sólo el desarrollo de los conocimientos propios de la materia sino también lograr que los alumnos adopten los hábitos de un estudiante universitario.

Sumado a esto, y como ocurre en las Universidades del conurbano bonaerense, la amplia mayoría de los alumnos son de Nuevo Ingreso, es decir, primera generación de estudiantes universitarios en su ámbito familiar. Lo que determina, entre otros aspectos, la ausencia de contexto acerca de los hábitos de estudios requeridos en la enseñanza universitaria, entre ellos la continuidad del ritmo de estudio.

En este contexto llevamos adelante esta experiencia para promover que los alumnos adopten un ritmo apropiado en sus estudios universitarios.

1.2 El contexto del curso

Como se trata de la primera materia de programación de la carrera los docentes asumimos que los alumnos no tienen conocimientos previos de programación.

Una de las ideas fundantes al momento de organizar los contenidos de la materia fue que los alumnos primero aprendan a usar objetos, y después a crearlos, a diferencia de otras formas más tradicionales de dictado. Este enfoque se basa en los conceptos desarrollados en torno al Curriculum Invertido [1] de que al adquirir los primeros conocimientos con un nivel de abstracción alto, se forma una estructura mental más apropiada para el desarrollo de software. Además, es más intuitivo bajar el nivel de abstracción de los conocimientos a medida que se los adquiere, en lugar de hacerlo a la inversa (ver el patrón “Abstraction Gravity” en [2]).

Partiendo de estas premisas, se organizó la materia de manera que un primer tercio consiste en manipular objetos de un juego llamado “Batalla espacial” desarrollado por uno de los docentes de la cátedra. A través de estas actividades los alumnos aprenden progresivamente a pasarle mensajes a un objeto (invocación de métodos), esperar resultados de las operaciones del mismo (valores de retorno), el concepto de enviar datos de entrada a las operaciones (parámetros), etc.

El segundo tercio de la materia se enfoca principalmente en la creación de clases propias para la resolución de problemas. En esta etapa no se utiliza la

“Batalla Espacial”, sino que los alumnos aprenden cómo construir sus propios objetos de acuerdo al problema que deben resolver.

La última parte de la materia tiene los contenidos más avanzados del curso: cómo realizar pruebas, algoritmos simples de búsqueda y ordenamiento, y conceptos de referencias y manejo de memoria.

La carga horaria es de ocho horas semanales y se dicta en dos clases teórico-prácticas por semana, con una asistencia promedio por encima de 30 alumnos, sobre un total de aproximadamente 35.

Para mantener el foco y la atención de los alumnos, los docentes procuran que la exposición teórica no exceda los 20/30 minutos continuos por clase. En función de ello, se promueve el aprendizaje kinestésico con actividades lúdicas, principalmente como disparadores de las exposiciones teóricas.

La materia se dicta en un aula-taller con computadoras y se dedica gran parte de la clase a trabajar en ellas para aplicar lo aprendido y ejercitar sobre los temas teóricos.

La forma de evaluación son dos exámenes parciales y un examen final. En el primer examen parcial se incluyen los temas de las primeras 5 guías de trabajos prácticos, y el resto en el segundo. Se proveen guías de ejercicios para el trabajo en clase y fuera de ella, las cuales son el objeto del presente estudio. Las guías de trabajos prácticos tienen los siguientes temas:

Guía 1: Usando objetos
Guía 2: Lógica
Guía 3: Estructuras de control
Guía 4: Repaso
Guía 5: Construyendo objetos
Guía 6: Pruebas Unitarias
Guía 7: Arreglos
Guía 8: Arreglos multidimensionales
Guía 9: Referencias
Guía 10: Ordenamiento
Guía 11: Búsqueda

Las primeras 5 guías son las alcanzadas por este estudio.

2 Marco conceptual e hipótesis de trabajo

El ritmo de estudio, entendido como una gestión exitosa del tiempo invertido y de los resultados obtenidos, es una de las preocupaciones centrales de los docentes y en general, de la pedagogía [3] [4]. Por otra parte, la necesidad de motivar a los alumnos para que tomen en sus manos y se comprometan con el proceso y las actividades de aprendizaje es una de las líneas centrales de muchas innovaciones pedagógicas, en particular de la enseñanza en tecnología. En [5] el autor plantea que “el ritmo y el control son una responsabilidad compartida entre el maestro y el alumno”.

En este contexto, y tomando los mecanismos de feedback como instrumento de mejora, consideramos que existen múltiples oportunidades de innovación pedagógica aplicando modificaciones sobre las prácticas tradicionales de feedback para la mejora del ritmo de estudio.

Las razones para enfocarnos en el feedback son múltiples:

- Es un concepto central en las innovaciones pedagógicas en la enseñanza de la computación [6] (en [2], *Feedback* es el título de una de las cinco secciones del libro).
- Aparece como una de las más básicas interacciones humanas [7].
- Es un concepto central en la literatura de desarrollo de software, en particular en el contexto de mejora continua ([8], ver páginas 206, 212 y 274).

En este caso, el feedback toma la forma de puesta en común del progreso colectivo y la posibilidad de visualizar el progreso de cada alumno con respecto a los demás, no tiene que ver con la corrección de los trabajos realizados (como en [6]). Este es un proceso de feedback no específico para cada alumno, puesto que no busca analizar lo que cada alumno hace sino tomarlo como parte de una realidad colectiva que involucra al resto de sus compañeros.

La hipótesis de trabajo es que crear amplia visibilidad del progreso logrado por el conjunto de alumnos produce en ellos la motivación a mejorar. En nuestra experiencia, además, se institucionalizó el feedback haciéndolo relativo al ritmo de progreso determinado por los docentes, que podemos considerar como el "ritmo de progreso de referencia". Buscamos generar así una presión moderada (no agresiva) y natural para motivar y mantener un ritmo de estudio eficaz.

A partir de poner en práctica este mecanismo, se presentan los resultados obtenidos durante el período que va desde el inicio de la cursada hasta la primera evaluación parcial.

3 La experiencia

Realizamos la experiencia durante el cuatrimestre agosto-diciembre de 2014. Los docentes presentamos a los alumnos la consigna durante la cuarta clase, en los siguientes términos:

- Cada semana de clase todos indican públicamente cuántos ejercicios de la guía correspondiente a esa semana (guía actual) han resuelto.
- Los resultados son visibles para todos durante la recolección de los datos.

Utilizamos un mecanismo simple para medir y generar feedback al mismo tiempo. En cada clase, un docente pregunta:

¿Cuántos han completado 0 ejercicios de la guía actual? ¿1? ¿2? … Y los alumnos levantan una mano cuando el número correspondiente es mencionado. De esa manera, tanto el docente como todos los alumnos pueden percibir de un golpe de vista la situación de progreso. Dicho de otra manera, el

recorrido por todos los números, desde 0 hasta la cantidad máxima de ejercicios de esa guía permite generar un histograma en muy pocos minutos. Además, genera feedback a los alumnos sobre su estado de avance comparado con el del resto, y presión de pares para avanzar.

Un concepto central del proyecto es el de *guía actual*. La misma representa el nivel de progreso esperado (o de referencia), es decir, la guía en la que deberían estar enfocados los alumnos en esa semana. Supone que la decisión de los docentes tiene en cuenta el ritmo necesario para completar la cursada de la materia exitosamente, y la importancia y dificultad relativas de cada guía. Además, este concepto permite simplificar las mediciones y el feedback, por cuanto no medimos el progreso en guías anteriores a la actual (que podría o no haber cambiado con respecto a la semana anterior). Una metáfora para la guía actual es el auto de seguridad en las carreras, que dicta el mínimo ritmo de progreso para seguir participando de la misma.

En la Tabla 1 se reproducen los datos recolectados:

Tabla 1. Datos recolectados sobre cantidad de ejercicios completados

Clase	5	6	7	8	9	10	11	12	13	14
Guía	1	2	3	3	3	4	4	5	5	5
Fecha	30/08	2/09	6/09	09/09	13/09	16/09	20/09	23/09	27/09	30/09
# Ejercicios	9	12	10	10	10	11	11	12	12	12
0	15	8	13	11	6	15	8	23	4	5
1	0	1	3	2	0	4	0	1	2	1
2	0	1	0	0	1	2	4	1	5	2
3	1	0	2	0	4	3	7	0	4	1
4	1	0	1	4	1	0	1	1	3	1
5	3	4	1	1	2	0	1	0	5	0
6	0	3	0	4	6	0	1	1	1	5
7	1	2	2	2	2	0	0	0	1	0
8	5	0	4	5	4	0	1	1	1	2
9	3	1	0	2	2	0	1	1	0	0
10		0	0	2	3	0	0	0	4	1
11		2				0	0	1	0	2
12		2						0	1	1
# Alumnos	29	24	26	33	31	24	24	30	31	21

Algunas aclaraciones sobre los datos recolectados:

- La primera clase es la 5 porque en las clases anteriores se trabaja con lecturas y ejercicios varios de resolución de problemas. En la 4 clase se inicia el trabajo con guías de ejercicios y a partir de ahí medimos, desde la clase siguiente, cuando los alumnos ya tuvieron tiempo de trabajar fuera de clase.
- Cada columna corresponde a una clase consecutiva de la cursada.

- Cada fila del cuerpo de la tabla corresponde a una cantidad de ejercicios resueltos.
- Cada celda del cuerpo de la tabla describe cuántos alumnos tienen resueltos, en esa clase, la cantidad de ejercicios definida por esa fila.
- El encabezado de la columna describe la clase, la guía considerada actual por los docentes, la fecha correspondiente a esa clase y la cantidad de ejercicios totales en esa guía.
- Algunas guías, por ser más largas o de temas más desafiantes, fueron consideradas "guía actual" durante más de una semana (dos semanas para la guía 4 y tres semanas para las guías 3 y 5).
- Las guías tienen distinta cantidad de ejercicios, por eso hay en la tabla celdas con 0 si ningún alumno tenía esa cantidad de ejercicios hechos, y en blanco si la guía no tenía esa cantidad de ejercicios.

Las mediciones han sido realizadas sobre los alumnos presentes sin identificar a cada uno individualmente. Por lo tanto, la cantidad de alumnos sobre la que se expresan los porcentajes varía de clase a clase.

4 Principales resultados

Se puede observar una marcada disminución en la cantidad de alumnos que no han comenzado (52% para la guía 1 y 24% para la guía 5) y un marcado aumento en la cantidad de alumnos que han completado más de la mitad de la "guía actual" (31% para la guía 1 y 52% para la guía 5). Aunque no hay evidencia de causalidad, puesto que aún no hemos usado mecanismos de control, en términos relativos la mejora observada es significativa (comparando las primeras guías medidas con las últimas).

Los resultados obtenidos fueron agrupados utilizando la siguiente estratificación:

- 0: alumnos que no hicieron ningún ejercicio de la guía actual.
- 1, 2: alumnos que hicieron uno o dos ejercicios de la guía actual.
- 3, 4, 5: alumnos que hicieron tres, cuatro o cinco ejercicios de la guía actual.
- 6 o más: alumnos que hicieron seis o más ejercicios de la guía actual.

La razón para hacer este agrupamiento es distinguir entre los extremos: no han comenzado (0) y tienen más de la mitad de las guías completadas (6 o más), y agregar alguna información sobre los intermedios, separando a los que han comenzado pero no han hecho casi ningún progreso (1,2) y los que están aproximadamente en menos de la mitad (3,4,5). Esta estratificación no representa estrictamente lo mismo para todas las guías dado que tienen distinta cantidad de ejercicios (entre 9 y 12) pero consideramos que es razonable para los objetivos del trabajo.

El análisis de tendencia a lo largo del tiempo fue realizado en dos escalas, clase por clase y agrupado por guías.

4.1 Ejercicios resueltos por clase

Considerando la estratificación antes descrita, en la Figura 1 se vuelca el porcentaje de alumnos que resolvieron N ejercicios por guía. Para las guías 1,2 y 3 se observan las mismas tendencias que para las guías 4 y 5, pero éstas se interrumpen entre las guías 3 y 4.

Las características de estas tendencias son:

- La proporción de alumnos que no hicieron ningún ejercicio de la guía actual (estrato “0”) disminuye conforme avanza la cursada.
- La proporción de alumnos que hicieron 6 o más ejercicios (estrato "6 o más") crece conforme avanza la cursada.
- Los decrementos del estrato "0" se corresponden mayoritariamente con incrementos del estrato "6 o más".

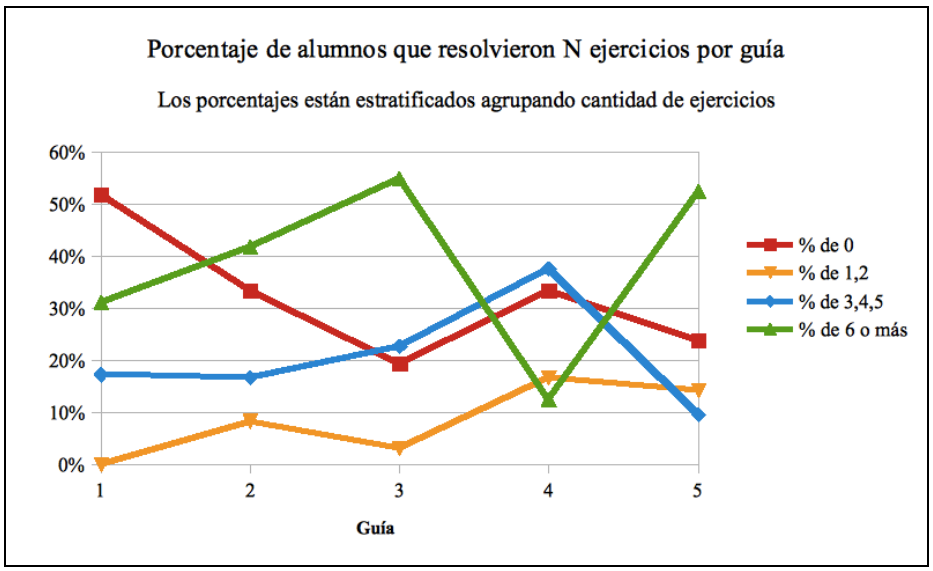


Fig. 1. Porcentaje de alumnos que resolvieron N ejercicios por guía.

En lo que respecta a la interrupción de estas tendencias entre las guías 3 y 4 se pueden destacar dos aspectos a considerar:

- Coincide con una marcada disminución en la cantidad de alumnos presentes, evidenciada por la cantidad de mediciones tomadas a partir de la clase 10 con la guía 4 (ver Tabla 1), consecuencia de la proximidad de exámenes parciales de esta y otras materias.
- La guía 4 reúne ejercicios más extensos que otras guías, que demandan mayor tiempo del alumno para su resolución

4.2 Ejercicios resueltos por clase

Debido a la complejidad de los ejercicios, las guías 3 y 5 tuvieron vigencia (fueron la guía actual) a lo largo de 3 clases, mientras que la guía 4 lo fue durante 2 clases.

El porcentaje de alumnos que resolvieron N ejercicios de la guía actual por clase se muestra en la Figura 2. En las mediciones progresivas de las guías 3, 4 y 5 se observa la misma tendencia entre clases de la misma guía que la observada entre guías consecutivas a lo largo de la cursada.

La única excepción es la última medición de la guía 5, en la que la proporción del estrato "0" aumenta. Dado que la cantidad de alumnos totales que no hicieron ejercicios de una guía determinada no puede incrementarse de una clase a otra (no es posible deshacer ejercicios), se evidencia que la medición no fue realizada sobre una población exactamente igual clase a clase. Este hecho también se condice con el incremento en el ausentismo de alumnos descrito anteriormente.

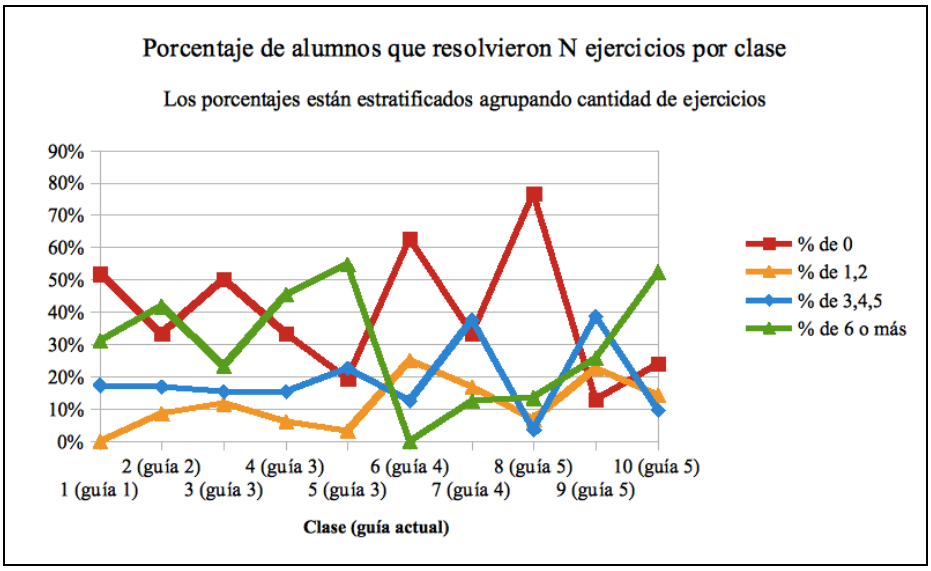


Fig. 2. Porcentaje de alumnos que resolvieron N ejercicios, por clase, indicando la guía actual entre paréntesis.

Los resultados obtenidos podrían indicar que insistir en dar feedback sobre una misma guía múltiples clases favorece el avance.

4.3 Relación con el desempeño en el examen parcial

Con el objetivo de contrastar la apropiación de conocimientos de los alumnos con la realización de las guías de ejercicios analizamos el desempeño en el primer examen parcial, correspondiente al final del período estudiado.

El examen parcial consta de 5 puntos, cada uno de los cuales se puede asociar a una temática correspondiente a una guía de ejercicios. La asociación de puntos del examen a guías de ejercicios se muestra en la tabla 2. Para evaluar cada uno de los puntos de un examen se utiliza la convención definida en la tabla 3.

Tabla 2. Asociación de guías a puntos del examen parcial

Punto	Temática	Guía
1	Uso de Objetos	1
2	Lógica en la Programación	2
3	Estructuras de Control	3
4	Estructuras de Control	4
5	Construcción de Objetos	5

Tabla 3. Calificación de cada punto del examen parcial

Nota	Descripción
-	En blanco
M	Mal
R	Regular
B	Bien

La distribución de notas observadas en la Figura 3 muestra mejores resultados (B + M) en los puntos (1, 2 y 3) cuyas guías de ejercicios asociadas tuvieron mayor nivel de resolución. Esto podría sugerir el impacto positivo del ritmo en los resultados de las evaluaciones.

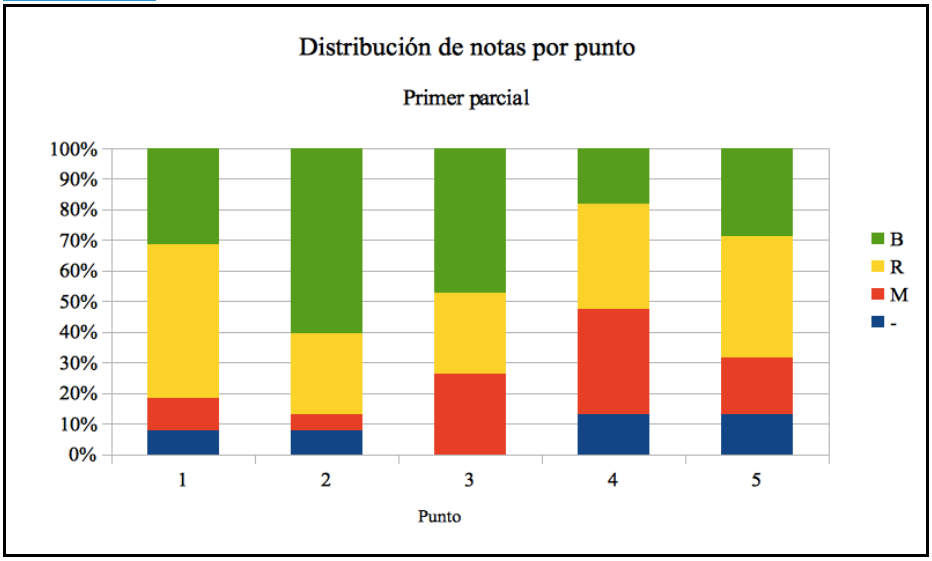


Fig. 3. Distribución de notas por punto para el primer parcial.

5 Conclusiones y trabajos futuros

Los resultados muestran claras tendencias de mejora relativa, con una singularidad en el caso de la guía 4. También muestran alguna coincidencia con los resultados de los exámenes, por cuanto la guía 4, la más conflictiva en términos de ritmo, aparece asociada al ejercicio del examen que tiene peores resultados.

Consideramos relevante compartir los resultados dado que la experiencia es simple y puede inspirar a otros docentes a experimentar con variaciones para intentar producir mejoras en sus propios contextos. En general, aunque no podemos identificar correlación entre el feedback y la mejora observada, lo reproduciremos siguiendo más estrechamente las reglas del diseño de experimentos.

Como perspectivas de investigación, consideramos interesante realizar el estudio con un grupo de control que no reciba feedback al aportar la información, para medición progreso sin feedback. Otra línea de trabajo consiste en explorar el efecto de medir múltiples veces cada guía (por lo menos dos), para estudiar si esto produce mejoras observables. Para ello, una alternativa es medir dos guías cada clase, la actual y a la anterior.

Agradecimientos. Agradecemos al Coordinador de la Carrera Alejandro Oliveros y la Directora de nuestro proyecto de Investigación Rosita Wachenchauzer por su acompañamiento metodológico y guía durante el trabajo.

Referencias

1. Meyer, B., A Touch of Class, Springer, (2012)
2. Bergin, J., Jutta, E., et al: Pedagogical Patterns: Advice for Educators. CreateSpace Independent Publishing, (2012)
3. de la Barrera, M. L., Donolo, D. S., Rinaudo, M. C.: Ritmo de estudio y trayectoria universitaria. Anales de psicología, vol. 24, no 1 (junio), 9--15, (2008)
4. Sposetti, A., Echevarría, H. D.: Tiempo dedicado al estudio y lentificación en la Universidad, Capítulo 5, Deserción y lentificación en el pensar y sentir de los alumnos universitarios. Investigación Evaluativa, (2004)
5. Rodríguez Acevedo, G. D.: Ciencia, Tecnología y Sociedad: una mirada desde la Educación en Tecnología, Revista Iberoamericana de Educación, Número 18 - Ciencia, Tecnología y Sociedad ante la Educación, Septiembre-Diciembre, (1998), http://www.rieoei.org/oeivirt/rie18a05
6. Falkner, N., Vivian, R., Piper, D., Falkner, K.: Increasing the effectiveness of automated assessment by increasing marking granularity and feedback units: Proceedings SIGSEA 2014, Páginas 9-14 doi 10.1145/2538862.2538896
7. Norman, D., The Design of Everyday Things. Basic Books, (1988)
8. Paez, N., Fontdevila, D., Suarez, P., Fontela, C., De Giovaninni, Molinari, A.: Construcción de software: una mirada ágil, EDUNTREF, 2014.

Estrategias Didácticas Innovadoras en la enseñanza de TICs para Ingeniería en Informática

Alicia Mon, Horacio Del Giorgio, Bettina Donadello

Universidad Nacional de La Matanza – Departamento de Ingeniería e Investigaciones Tecnológicas - Florencio Varela 1903 (B1754JEC) - San Justo, Buenos Aires, Argentina
alicialmon@gmail.com , hdelgiorgio@ing.unlam.edu.ar , bdonadello@ing.unlam.edu.ar

Abstract

El presente artículo expone los resultados de una investigación desarrollada sobre el uso de estrategias didácticas innovadoras para el dictado de Asignaturas orientadas a las TICs en la carrera de Ingeniería Informática de la Universidad Nacional de La Matanza.

Para desarrollar la investigación se realizó una encuesta entre los docentes de las materias de TICs de los últimos años de la carrera que permitió indagar sobre las diferentes estrategias utilizadas según los contenidos de las asignaturas y el perfil de los estudiantes. Asimismo, se realizó un estudio exploratorio con alumnos del último año de la carrera aplicando diferentes estrategias didácticas en grupos diversos, de manera que pudiera ser evaluado el impacto de la utilización de las mismas.

Los resultados que se presentan aquí demuestran que el uso del humor es percibido positivamente por el conjunto de los docentes, en tanto que mejora el ambiente de la clase, la relación entre el docente y los alumnos, las actitudes frente al estudio, la adquisición de conocimientos, la comprensión y fijación de conceptos, así como la transferencia por evocación.

No obstante, su utilización carece de formalismos en los docentes debido a una evaluación negativa del uso ordenado, planificado y sistematizado para el contexto universitario.

Keywords: Procesos Educativos, Enseñanza y Aprendizaje en las TICs, Estrategias Didácticas Innovadoras, el Humor

1 Introducción

La distancia subjetiva entre docentes y alumnos en la actualidad, especialmente de las carreras de áreas tecnológicas, pareciera generar una brecha en los modos de abstracción y adquisición del conocimiento. En este sentido, desmitificar o "humanizar" la enseñanza de los contenidos de las tecnologías de la información y las comunicaciones (TICs) permitiría acercar a los alumnos en la comprensión de los fenómenos de manera más sencilla, completa y realista.

En la práctica áulica cotidiana, especialmente en la Universidad, la enseñanza se posiciona desde un lugar donde cualquier manifestación que enfrente la solemnidad, del tipo del humor o las actividades lúdicas, no ingresa como parte de una estrategia para posibilitar un mayor acercamiento al alumno, ni como un ítem intencionalmente considerado en la propuesta o planificación del docente.

Podría pensarse que la raíz de la escisión entre enseñanza y humor radica en que se intenta transmitir los saberes científicos seriamente, recontextualizando los mismos de modo comprensible para quienes aprenden [1]. En este proceso, llamado

"Transposición Didáctica", pareciera quedar relegada la posibilidad de disfrutar de aquello que se aprende y se enseña en las prácticas educativas.

Sin embargo, el humor forma parte de la cotidianeidad humana y se halla implicado en numerosos actos comunicacionales.

En su "Tratado sobre el Alma", Aristóteles [2] menciona que "el hombre es el único ser viviente que ríe", a lo que Bergson [3] le agrega que "también es el único que hace reir".

Más allá de la Filosofía, el estudio del fenómeno denominado "humor" puede ser abordado desde muy diversas disciplinas: antropología, sociología, lingüística, psicología, fisiología, medicina o educación y constituye, por lo tanto, un campo interdisciplinario.

Existen muchos elementos reunidos bajo el concepto de "humor", entendido como lo que puede provocar risa o divertimento. En algunos casos, existen elementos gestuales que no necesitan palabras, dibujos, caricaturas, juegos de palabras que aprovechan los aspectos fónicos, palabras prohibidas insertas en un contexto distinto del habitual. El reír implica una relación con alguien, nadie se cuenta un chiste a sí mismo. El humor en sí, supone un encuentro social, un consenso de significados y actitudes, una intención y una respuesta [4].

Citado por Freud [5], Kant destaca que: "... constituye una singular cualidad de lo cómico el no podernos engañar más que por un instante", con lo que se resalta la fuerte relación entre el factor de desconcierto, seguido de esclarecimiento que están presentes en las situaciones humorísticas [4].

El estudio del uso del humor en la enseñanza y el aprendizaje parece interesante para un gran número de educadores, psicólogos y expertos en educación. Sin embargo, no pareciera tener un correlato académico en la formación docente y en la aplicación para la enseñanza de las tecnologías.

Por otra parte, en varias investigaciones [4] [6] que han indagado sobre la opinión de alumnos y docentes respecto de la inclusión del humor en el aprendizaje, es frecuente encontrar reflexiones acerca de las ventajas de incluir el humor en las clases, el tipo y frecuencia, la planificación y la espontaneidad, entre muchas otras situaciones. En ellas puede leerse una búsqueda constante de condiciones para mejorar el aprendizaje. Sin embargo, esta búsqueda queda limitada a lo que el docente hace y no se hace extensiva a las acciones del alumno, ni en forma individual ni en forma grupal.

En general, el tema del humor en la formación docente es un aspecto poco abordado por quienes investigan el uso del humor en las aulas. Así, éste queda, o bien librado a las características personales del docente, o bien planificado en un guión de tipo instruccional.

En el presente artículo se exponen los resultados de una investigación del Departamento de Ingeniería e Investigaciones Tecnológicas de la Universidad Nacional de La Matanza realizada entre docentes y alumnos de asignaturas relacionadas con TICs, en la cual se explora sobre el uso del humor y de estrategias innovadoras para la enseñanza de tales contenidos en la Carrera de Ingeniería Informática.

2 Metodología

Para el desarrollo de la investigación, se ha realizado un estudio exploratorio entre docentes responsables del dictado de materias de Tecnologías básicas y Tecnologías aplicadas de la carrera de Ingeniería Informática. Por otra parte, se ha realizado un experimento entre alumnos de diferentes cursos de una de las materias estudiadas a los cuales se les aplicaron estrategias didácticas diferenciadas de modo tal que se pudieran analizar los efectos en la adquisición de conocimiento de iguales contenidos, pero aplicando diferentes métodos de enseñanza.

En el presente artículo, se expondrán los resultados del estudio realizado entre los docentes a través de un cuestionario que incluía preguntas cerradas y preguntas abiertas, cuyas repuestas eran anónimas y la participación voluntaria. De 25 asignaturas de Tecnologías Básicas y Tecnologías Aplicadas que contiene el plan de estudios analizado, se encuestaron docentes de 21 materias, es decir que se tomó una muestra del 85% de las materias dictadas actualmente. Los Docentes encuestados fueron Responsables de Cátedra y/o Jefes de Trabajos Prácticos.

El cuestionario ha sido estructurado en dos partes. El primer apartado estaba compuesto por un conjunto de 19 preguntas cerradas que contemplaban posibles respuestas utilizando la escala Likert, todas ellas de carácter cualitativo, mientras que el segundo apartado estaba compuesto por 3 preguntas abiertas.

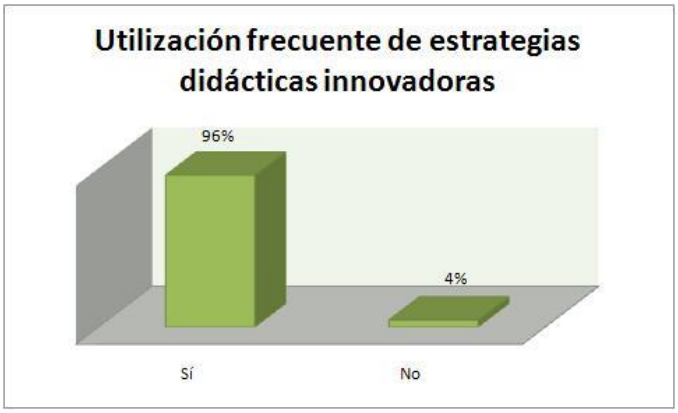
La organización del cuestionario se estructuró de la siguiente manera: la primera Pregunta (cerrada) resultaba fundamental para el desarrollo de todo el cuestionario, ya que la misma consistía en indagar si el docente consideraba que el uso de estrategias didácticas innovadoras (humor, chistes, ocurrencias, gestos, historias divertidas, dinámicas o juegos) en el dictado de la Asignatura que podía influir positivamente en las clases. Si la respuesta era positiva, se continuaba con el Cuestionario hasta la Pregunta 14 inclusive, siendo cerradas las Preguntas 2 a la 11, y Abiertas de la 12 a la 14. Para el caso de respuesta negativa estaban las Preguntas cerradas 15 a 19, ya que también resultaba importante conocer las causas de los desacuerdos, en el caso de que existieran.

3 Resultados

Del total de los docentes encuestados, el 100% respondió afirmativamente la Primera pregunta que considera como positivo el uso de estrategias didácticas innovadoras, tales como humor, chistes, ocurrencias, gestos, historias divertidas, dinámicas o juegos en el dictado de su Asignatura.

Por tal motivo, a continuación se presenta el análisis de las siguientes respuestas del cuestionario que consideran como positivo el uso dichas estrategias.

A la pregunta de si "¿utiliza frecuentemente estrategias didácticas innovadoras durante sus clases?", sólo el 96% de los encuestados respondió afirmativamente, lo que indica que del 100% de los docentes que consideran positiva su utilización, existe un 4% que no las utiliza, tal como se muestra en el Cuadro 1.

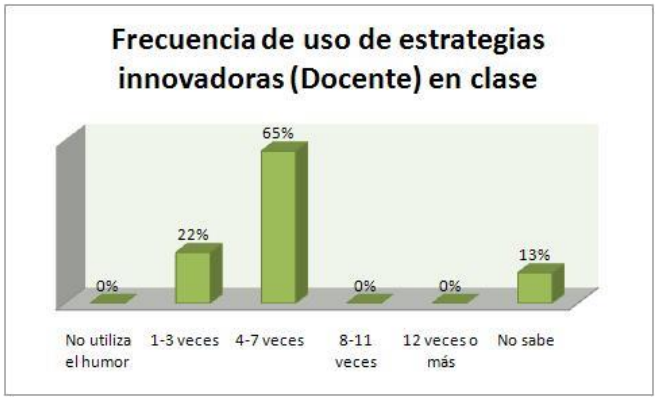


Cuadro 1

Es dable destacar que del 96% que utiliza estas estrategias, en algunos casos han mencionado explícitamente que "al menos lo intentan", mientras que otros señalaron que les solicitan a los ayudantes de sus cátedras que apliquen estas estrategias a través de charlas o anécdotas, en tanto que les llamaba la atención que los alumnos le

formularan preguntas referidas a tales experiencias en los exámenes, destacando la influencia del desarrollo afectivo y cognitivo conjunto con el docente y la transferencia por evocación [4].

Respecto a la pregunta "¿con qué frecuencia (de media) utiliza alguna estrategia didáctica innovadora (humor, chistes, ocurrencias, gestos, historias divertidas, dinámicas, juegos…) durante cada clase?", la respuesta más elegida ha sido la opción 3 (4-7 veces) con un 65%, seguida por la opción 2 (1-3 veces) con un 22%. Es de destacar que ningún docente encuestado ha considerado la opción 1 "No utilizo el humor", tal como se presenta en el Cuadro 2.



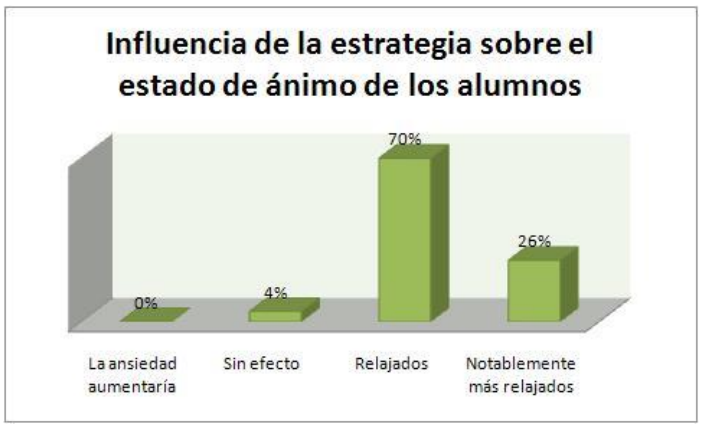
Cuadro 2.

A la pregunta sobre "¿qué aspecto de la estrategia didáctica innovadora está relacionado o es relevante para el tema abordado en el aula?", se observa que la gran mayoría de las respuestas de los docentes estuvieron orientadas en las opciones 4 (La Mayoría) y 5 (Todo). Se destaca que la opción 1 (Ninguna) no ha sido considerada por los encuestados, tal como se expone en el cuadro 3. En este sentido, se podría inferir que el uso de estrategias didácticas innovadoras está relacionad o es relevante a la hora de ponerlas en práctica con el tema dictado en cada clase.

Cuadro 3

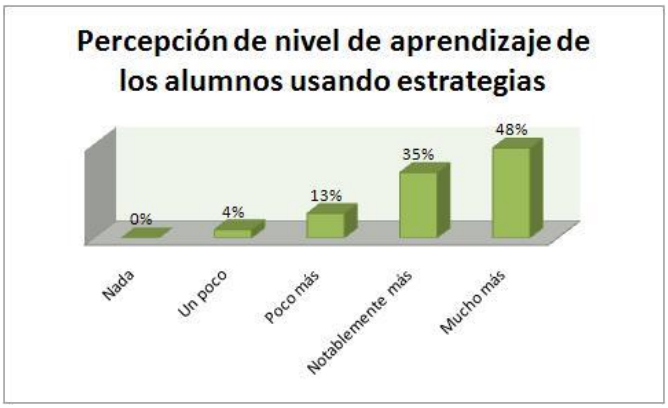
Respecto a la pregunta "¿hasta qué punto la estrategia didáctica innovadora hace (o cree que haría) que sus alumnos se sientan más relajados en sus clases?", se observa que, sobre todas las opciones, la respuesta 3 (Relajados) ha sido considerada por el 70% de los encuestados, seguida de la 4 (Notablemente más relajados) con un 26%, tal como se observa en el cuadro 4. En este sentido, pareciera que los docentes encuentran como positiva esta perspectiva de la utilización de estas estrategias para generar un clima distendido en la clase.

Asimismo, es dable destacar que la opción 1 (La ansiedad aumentaría) no ha sido considerada por ninguno de los encuestados, en tanto que la valoración 2 (Sin efecto) son evaluadas por el 4% de los docentes.



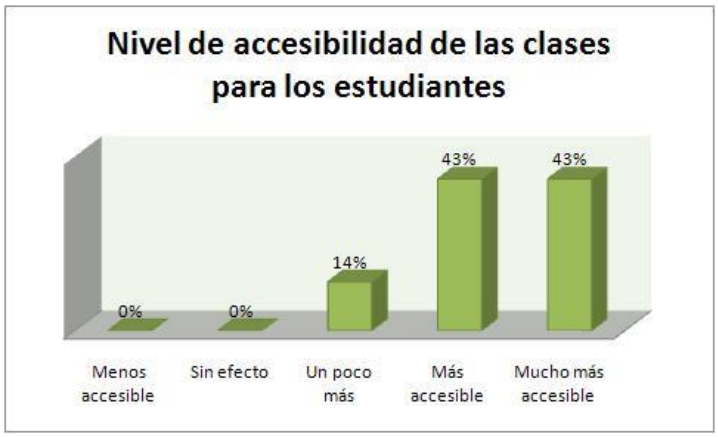
Cuadro 4

Al consultar a los docentes sobre si "¿cree que sus estudiantes aprenden ó aprenderían más sobre su materia al ser expuestos a situaciones distendidas o menos estructuradas durante la clase?", la respuesta más numerosa es la 5 (Mucho más) con un 48% del total de la muestra. Si se le suma el 35% de los que han elegido la opción 4 (Notablemente más), se detecta que el 83% de los docentes está de acuerdo con que éste es un tema que se tiene en cuenta. Por otra parte, aquellos docentes que han elegido las opciones 3 (Poco más) representan el 13% y la opción 2 (Un poco) constituyen el 4%, en tanto que la opción 1 "Nada" no ha sido tenida en cuenta por los encuestados, tal como se presenta en el cuadro 5.



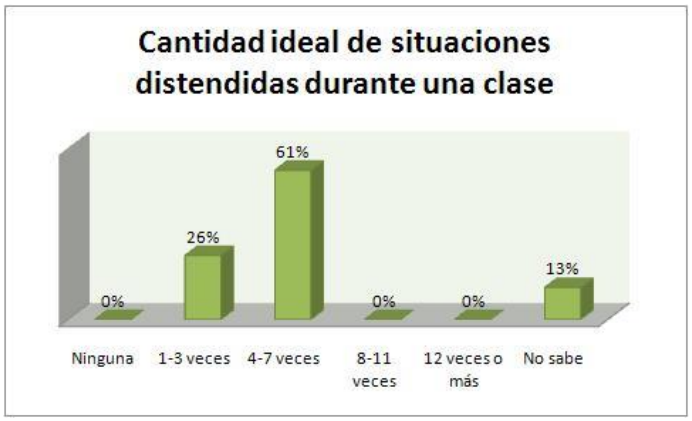
Cuadro 5

Respecto a la pregunta de si "¿cree que el uso de situaciones distendidas o menos estructuradas le hace más o menos accesible la clase a los estudiantes?", se destaca que las opciones 4 (Más accesible) y 5 (Mucho más accesible) representan un 86% de los encuestados, en tanto que un 14% optó por la opción 3 (Un poco más). Las opciones 1 (Menos accesible) y la 2 (Sin efecto) no han sido consideradas como válidas para los encuestados, tal como se muestran en el cuadro 6.



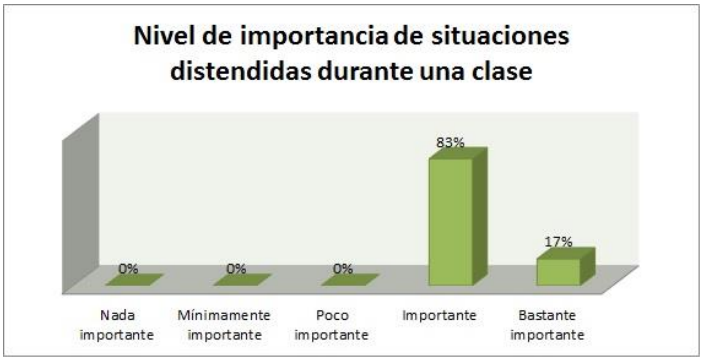
Cuadro 6

Sobre la opinión de "¿cuál es la cantidad ideal de situaciones distendidas o menos estructuradas para un entorno propicio para el aprendizaje durante una clase?", la opción más elegida ha sido la número 3 (4-7 veces) con un 61% de las respuestas. La opción 2 (1-3 veces) ha contado con un 26%, en tanto que las opciones 1 (Ninguna), 4 (8-11 veces) y 5 (12 veces o más) no han sido tenidas en cuenta, mientras que el 13% no se sintió en condiciones de proveer este dato, tal como se presenta en el cuadro 7.



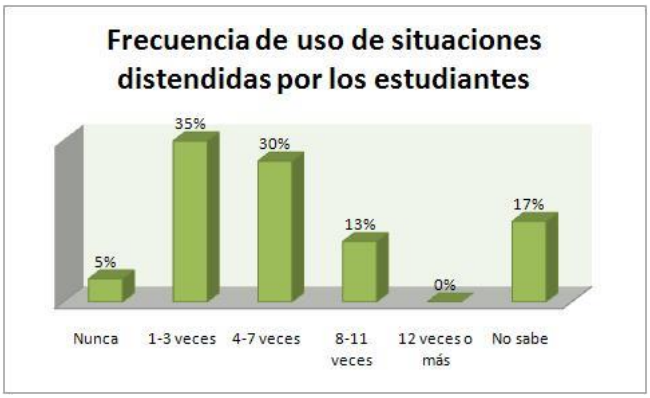
Cuadro 7

En relación a la opinión de los docentes sobre "¿cuánto de importantes son las situaciones distendidas o menos estructuradas para la asignatura que usted dicta?", el 83% de los docentes encestados considera que es "Importante", mientras que el 17% restante lo ve como "Bastante importante". Las respuestas 1 (Nada importante), 2 (Mínimamente importante) y 3 (Poco importante) no han sido tenidas en cuenta por los encuestados, tal como se muestra en el cuadro 8.



Cuadro 8

En tanto que al preguntar a los docentes "¿con qué frecuencia los estudiantes utilizan situaciones distendidas o menos estructuradas para comunicarse durante cada clase?", la opción más elegida ha sido la 2 (1-3 veces) con un 35% seguida de la 3 (4-7 veces) con un 30%, la opción 4 (8-11 veces) un 13% en tanto que la opción 1 (Nunca) ha sido considerada por el 5%. La opción 5 (No sabe) ha sido considerada en un 17%, tal como se expone en el cuadro 9.



Cuadro 9

En cuanto a la pregunta de si "¿utilizaría el humor sarcástico o irónico en sus clases?", entre las respuestas obtenidas, un 43% optó por una posición afirmativa y algunos mencionaron que "utilizan el absurdo", mientras que otros utilizan el sarcasmo en forma directa. A esta afirmación, se le agrega un 17% que manifestó que "No lo descartaría". Sin embargo, un 35% de los docentes respondió "Nunca" manifestando una percepción negativa sobre su utilización en clase, tal como se muestra en el cuadro 10.

Cuadro 10

A partir de tales preguntas, se exploraron algunos aspectos que permitieran explicar abiertamente las consideraciones de los docentes. Es así que a la pregunta "¿le parece interesante el hecho de utilizar alguna estrategia innovadora desde lo didáctico?", solicitándoles que además fundamentaran sus respuestas, la mayoría de las mismas apuntaban directamente hacia una percepción positiva en los alumnos y en la relación Docente-Alumno.

Se ha indagado sobre otras alternativas al uso del humor y las respuestas fueron muy variadas, entre las que es dable destacar la utilización de las siguientes estrategias didácticas: comentarios sobre la disciplina profesional; anécdotas de profesionales; juegos grupales; cambio de escenario del aula para el dictado; representaciones y teatralizaciones; stand-up; videos; juegos de magia; juegos de roles; rol playing por el absurdo/sarcasmo; lecturas de párrafos de filósofos; resolución de problemas con elementos no didácticos (ej. naranjas) y escenificación de conflictos.

Finalmente, se indagó con los docentes acerca de si tenían sistematizadas las estrategias didácticas innovadoras, y la respuesta general ha sido negativa. Es decir que, la mayoría de los docentes que utilizan diferentes estrategias didácticas, no las han sistematizado ni las incluyen en la propuesta curricular de la Asignatura. Entre las razones de la no sistematización existen diversas causas, de las cuales se desatacan la decisión personal y subjetiva de cada docente; la necesidad de adaptación a cada auditorio; a la situación personal de cada docente; o la convicción de que jamás sistematizarían ese tipo de estrategias por tener una percepción negativa sobre su evaluación formal. Por otra parte, un porcentaje menor de docentes manifestó que sí posee actividades sistematizadas, expresadas en las fichas curriculares, así como también utilizan foros de apoyo para alumnos.

4 Análisis de resultados

En relación a todas las respuestas obtenidas sobre el 85% de las materias relevadas en docentes de las temáticas de Tecnologías Básicas y Tecnologías Aplicadas, es posible analizar que el uso de estrategias didácticas innovadoras en el dictado de Asignaturas relacionadas con TICs se destacan los siguientes elementos:

- El profesorado encuestado utiliza frecuentemente estrategias didácticas Innovadoras durante sus clases.
- El profesorado encuestado utiliza una media de cuatro a siete veces este tipo de estrategias durante cada sesión de clase.
- Estas estrategias utilizadas están relacionadas la mayoría de las veces o bien todas las veces con el tema dictado en el aula.
- El profesorado considera que el uso de estas estrategias permite que los alumnos se sientan notablemente más relajados en la clase.
- El uso de estas estrategias en el proceso de enseñanza aumenta notablemente el interés de los estudiantes para aprender el tema dictado, según la percepción de los docentes.
- El uso de estas estrategias por parte de los encuestados en clase hace que la misma sea mucho más accesible a sus estudiantes.
- La cantidad ideal de situaciones distendidas que propicia el aprendizaje durante un período normal de clase oscila entre cuatro y siete veces según nuestros profesores.
- Para el aprendizaje de las materias dictadas, el uso de estas estrategias es considerado por parte de los profesores como un elemento importante.
- De acuerdo a la percepción de los docentes, sus estudiantes generan también ese tipo de situaciones de una a tres veces para comunicarse durante cada clase.

- Un notable porcentaje de los Profesores encuestados utiliza o utilizaría el humor sarcástico durante sus clases. Por otra parte, en las investigaciones tomadas como referencia [4] [6], el uso del sarcasmo fue considerado una forma negativa del humor. Sin embargo, en las respuestas de final abierto dadas por los estudiantes en esas mismas investigaciones, el sarcasmo fue ubicado en quinto lugar como forma apropiada de humor. Cuando se indaga a los alumnos qué tipo de humor usarían si ellos fueran profesores, ubicaron el sarcasmo como una de las tres opciones más elegidas. Estos hallazgos se contradicen con la concepción de que el sarcasmo es una forma hiriente de humor, y constituye una línea de investigación interesante.

5 Conclusiones y trabajo futuro

En el presente artículo se han presentado los resultados de una investigación sobre el uso de estrategias didácticas innovadoras en la enseñanza de TICs en carrera de Ingeniería Informática de la Universidad Nacional de La Matanza. La investigación se ha realizado con docentes y alumnos de asignaturas relacionadas con TICs, en la cual se ha explorado sobre el uso del humor y de estrategias innovadoras para la enseñanza de tales contenidos.

Como resultado podría evaluarse que la escisión entre enseñanza y humor radica en que se intentan transmitir los saberes científicos seriamente, y la percepción de los docentes es sumamente positiva, en tanto que su formalización y sistematización carece de importancia o se percibe como no adecuada para la enseñanza en el ámbito universitario. Es así como el proceso de "Transposición Didáctica" [1], pareciera quedar relegada de la posibilidad de disfrutar de aquello que se aprende y se enseña en las prácticas educativas de las TICs.

Queda pendiente en esta investigación el análisis del estudio realizado con los alumnos de las mismas asignaturas relevadas, en las cuales se han generado un conjunto de prácticas aplicadas en una misma materia en grupos diferentes para evaluar el impacto en los alumnos de tales actividades.

6 Referencias

[1] Chevallard, Yves (1997) – "La Transposición Didáctica: del saber sabio al saber enseñado", Editorial Aiqué, Buenos Aires, Argentina.

[2] Aristóteles (2010) – Acerca del Alma. Editorial Colihue. Santiago. Chile.

[3] Bergson, H. (1973) - La risa. Editorial Espasa-Calpe. Madrid. España.

[4] Guitart, Mónica (2009) – "Permitido reir… Estamos en clase!". [Fecha de consulta: 21 de Julio de 2015]. Disponible en http://www.academia.edu/917513/_PERMITIDO_REIR..._ESTAMOS_EN_CLASE_

[5] Freud, Sigmund (1905) – "El Chiste y su relación con el Inconsciente", Editorial Alianza, Madrid, España.

[6] Pirowicz, Denise (2010) – "El Humor en los Procesos de Enseñanza y Aprendizaje". [Fecha de consulta: 21 de Julio de 2015]. Disponible en http://repositorio.flacsoandes.edu.ec/bitstream/10469/3381/3/TFLACSO-01-2011DP.pdf

Experiencia en la construcción de Objetos de Aprendizaje para árboles AVL usando CROA

Catalina Mostaccio[1,2], Gabriela Pérez[1]

[1] LIFIA – Facultad de Informática de la Universidad Nacional de La Plata
{catty, gperez}@lifia.info.unlp.edu.ar
[2] Also CICPBA

Abstract. Los Objetos de Aprendizaje (OAs) son un recurso versátil y prometen ser una herramienta exitosa en el proceso de enseñanza-aprendizaje. Ofrecen a los estudiantes contenidos interactivos que despiertan su interés, los motivan y guían en ese proceso. La Facultad de Informática de la UNLP en su búsqueda para fomentar el desarrollo de OAs dentro de la comunidad educativa lanzó una convocatoria destinada a los docentes de los niveles de grado y posgrado para presentar OAs que hubieran producido o desearan producir para uso en sus cátedras. En este contexto, en la cátedra Algoritmos y Estructuras de Datos[1] se desarrolló un OA para enseñar un tema conflictivo por la gran cantidad y diversidad de material disponible en la web. Este trabajo cuenta la experiencia de crear un OA para la enseñanza de árboles AVL manteniendo los criterios propios de la cátedra y aplicando la metodología para la CReación de Objetos de Aprendizaje - CROA [1].

Keywords: Objetos de Aprendizaje, Metodología CROA, Estructuras de datos, Proceso enseñanza-aprendizaje, Materiales educativos.

1. Introducción

La informatización actual de la sociedad ha incrementado la demanda de profesionales informáticos. La formación que reciben los estudiantes en los primeros años de la Facultad permite que puedan insertarse tempranamente al mercado laboral antes de finalizar la carrera. Deben combinar el trabajo con los estudios, pero no siempre lo consiguen satisfactoriamente. En muchos casos no pueden responder a las exigencias de las materias que cursan, porque encuentran reducida su disponibilidad de tiempo, o no pueden cumplir con los horarios requeridos. Los docentes debemos hacer frente a esta situación utilizando las herramientas que estén a nuestro alcance, aplicando estrategias metodológicas apropiadas y usando nuevos recursos disponibles

[1] "Algoritmos y Estructuras de Datos" es una asignatura de grado (obligatoria) de la Facultad de Informática de la Universidad Nacional de La Plata. Dicha asignatura es semestral y corresponde al segundo año de las carreras Licenciatura en Informática, Licenciatura en Sistemas y Analista Programador Universitario.

(por ejemplo, entornos virtuales de aprendizaje) que les permita a los alumnos cubrir a distancia esas carencias.

Los objetos de aprendizaje (OA) son un recurso valioso en el proceso de enseñanza-aprendizaje. Han demostrado ser un recurso versátil y prometen ser una herramienta exitosa en este campo. Ofrecen a los estudiantes contenidos interactivos que despiertan su interés, los motivan y los guían en ese proceso. Facilitan el dictado de materias en la modalidad de estudio semipresencial y a distancia. Permiten que los docentes adapten los recursos didácticos de acuerdo con sus propias necesidades y estilos de enseñanza. Pueden ser accedidos en cualquier momento y proveen un camino de aprendizaje alternativo y adaptable a las necesidades específicas de cada estudiante para que puedan elaborar el conocimiento a través de la propia reflexión y experimentación en la práctica, evitando la simple memorización de información. Un OA es una unidad autónoma que permite alcanzar un determinado objetivo educativo y que combinada con otros OAs puede dar lugar a la construcción de estructuras más complejas tales como lecciones, módulos o cursos.

En la Facultad de Informática de la UNLP se llevan adelante distintos proyectos destinados a la innovación en la enseñanza. A fines del 2014, se lanzó una convocatoria destinada a los docentes para presentar OAs que hubieran producido o desearan producir para uso en la enseñanza de sus cátedras [2]. En este contexto, en la cátedra Algoritmos y Estructuras de Datos se desarrolló un OA para enseñar un tema que resulta conflictivo por la gran cantidad y diversidad de material disponible en la web. Esta materia tiene como propósito que el estudiante aprenda conceptos básicos de las estructuras de datos. Entre los contenidos de la materia se encuentran estructuras de datos como árboles binarios, árboles AVL, árboles generales, grafos. La cátedra contaba con presentaciones digitales sobre los temas de la currícula y además, utilizaba la plataforma Moodle que permitía publicar ese material e interactuar con los alumnos. Por lo tanto, la incorporación del OA en este entorno ya familiar, favorecería el proceso de aprendizaje.

Este trabajo cuenta la experiencia de crear un OA para árboles AVL, manteniendo los criterios propios de la cátedra y aplicando la metodología CROA definida por docentes de la Facultad. Nuestro objetivo fue, entonces, facilitar el acceso de los alumnos a estos conceptos para propiciar la construcción de una base sólida en su formación sobre el tema que les permita comprender otras terminologías y diferenciarlas de la utilizada en la cátedra.

El resto del trabajo está organizado de la siguiente manera: la sección 2 presenta una breve descripción del tema que será el objetivo de aprendizaje del OA, la sección 3 describe los principales lineamientos de la metodología CROA, la sección 4 muestra el desarrollo de los OAs correspondientes a los árboles AVL siguiendo la metodología y finalmente la sección 5 muestra las conclusiones y los trabajos futuros.

2. La estructura de Árboles AVL

A continuación presentamos brevemente las características de los árboles AVL [3]. Además, puntualizamos los criterios propios utilizados en la cátedra, seleccionados del material bibliográfico existente.

Los árboles AVL son árboles binarios de búsqueda (ABB) que satisfacen la condición de estar balanceados respecto a su altura. Por ser un ABB, respeta la propiedad de orden en todos sus nodos, es decir, para cada nodo, su clave es mayor que todas las claves del subárbol izquierdo y menor que las claves del subárbol derecho [4]. La propiedad de balanceo garantiza que para cada nodo del árbol, la diferencia de altura entre el subárbol izquierdo y el subárbol derecho es a lo sumo 1 [5]. Gracias a esta propiedad, la complejidad de todas las operaciones (búsqueda, recuperación, inserción y eliminación) en árboles AVL con n nodos, es siempre, proporcional a la altura del árbol, es decir, utilizando notación Big-Oh, es O(log n) en el peor de los casos [5,6].

Las operaciones de búsqueda y recuperación se resuelven con procesos análogos a los aplicados para ABB. En el caso de las operaciones de inserción y eliminación, los procesos existentes para los ABB no son suficientes, ya que se debe contemplar una eventual destrucción de la propiedad de balanceo. Esta propiedad se re-establece identificando el nodo que se desbalanceó y luego cambiando la estructura a través de un mecanismo denominado "rotación" [5,6,7,8]. Una rotación consiste en permutar dos nodos (x e y) conectados por la relación "padre-hijo" manteniendo la propiedad de orden [6, pag. 242].

Los conceptos de balanceado y rotación aparecen en la literatura bajo diferentes formas. En el caso del balanceado, algunos autores lo computan a partir de las alturas de los subárboles, una información adicional almacenada en el nodo [5,6,7]; otros, almacenan, además, en cada nodo un "factor de equilibrio" con valores: -1, 0 o 1, indicando si ambos subárboles tienen la misma altura o cuál tiene la diferencia de 1 permitida [8,9]. Con respecto al mecanismo de rotación, los autores difieren sólo en la denominación; algunos determinan el nombre según la ubicación del nodo hijo mencionado (en el caso de los AVLs coincide con el subárbol que tiene mayor altura al momento de perder su nodo padre la condición de balanceado) [5,6,7]; mientras que otros, por el contrario, consideran la ubicación que finalmente tomará el nodo padre luego de la rotación (en el caso de los AVLs, coincide con el subárbol que tiene menor altura) [6,8,9].

Esta diversidad de criterios también se encuentra diseminada por la web. Cuando los alumnos recurren a Internet para esclarecer algún concepto, encuentran tanta información que los confunde, desanima y obstaculiza su proceso de aprendizaje.

Para evitar esta situación, decidimos plasmar este tema con los criterios propios de la cátedra en un OA que estuviera a disposición de los alumnos. La cátedra adopta los criterios definidos por Mark Allen Weiss [10] por considerarlo referente en el tema. Además, es un autor didáctico y su material resulta amigable para los primeros años de carreras de Ciencias de la Computación.

3. Metodología CROA

En la metodología CROA se define a un OA como un tipo de material educativo digital, que se caracteriza: desde el punto de vista pedagógico, por orientarse a un objetivo específico de aprendizaje, y presentar mínimamente: una serie de contenidos con el fin de abordar la temática relacionada con el objetivo, una actividad que

permita al alumno poner en práctica o problematizar el contenido presentado, y una autoevaluación que posibilite conocer al alumno si ha podido comprender esos contenidos vinculados al objetivo. Desde el punto de vista tecnológico, por contener un conjunto de metadatos estandarizados para su búsqueda y recuperación, y estar integrado, utilizando un modelo de empaquetamiento que respete estándares, y de esta manera permita su diálogo con diferentes entornos tecnológicos [1]. Esta definición enfatiza las características de reutilización, accesibilidad e interoperabilidad.

Esta metodología presenta 5 fases para la creación de un OA. Para cada una de las fases se plantean una serie de preguntas que resultan ser orientadoras tanto para el análisis como para el diseño del OA. Al finalizar cada fase se deben documentar las decisiones tomadas, y esos resultados serán los que fundamenten el OA.

1) Fase de Análisis: se plantean preguntas que dan origen al OA respecto a una necesidad que será satisfecha con el mismo. Qué se necesita que los alumnos aprendan o qué técnica se necesita que ejerciten, son algunas de las preguntas que orientarán el tema que abordará el OA. También deben responderse preguntas respecto a qué nivel educativo está dirigido y cuáles son los conocimientos previos necesarios. Estas preguntas ayudan a contextualizar el tema del OA.

2) Fase de Diseño: esta fase está dividida en tres actividades: Un diseño instruccional, un diseño de la estructura y un diseño de los componentes multimediales. En el diseño instruccional se responden preguntas respecto al objetivo del OA, que información contendrá, y la selección y organización de los contenidos. Además, se deben proponer las actividades y las autoevaluaciones. En el diseño de la estructura se planifica cómo será el recorrido entre los diversos nodos de información que componen el OA. Por último, en el diseño de componentes multimediales se propone de qué manera se van a presentar los elementos multimedia definidos dentro de los contenidos, en las actividades y las autoevaluaciones. Se deben definir los escenarios para la presentación de contenidos y las plantillas que se van a utilizar para el diseño de actividades y autoevaluaciones.

3) Fase de Desarrollo: define cuatro actividades que deben ser realizadas para completar el desarrollo de un OA: la selección de recursos, el desarrollo de los contenidos con las plantillas propuestas, la definición de los metadatos y la integración y empaquetamiento. Dentro de la selección de recursos, se deben elegir las imágenes, los textos, los videos o las animaciones que formarán parte del OA. Luego se debe desarrollar el OA propuesto con alguna herramienta de desarrollo adecuada. Posteriormente, se procederá a la definición de los metadatos para que pueda ser accesible y por último a la integración y empaquetamiento para que pueda ser usado en distintas plataformas.

En todo momento la metodología propone comprobar la coherencia entre la necesidad que originó el OA con los contenidos, las actividades y las autoevaluaciones.

4) Fase de Publicación: se debe definir dónde se va a publicar el OA, en un Entorno Virtual de Enseñanza y Aprendizaje (EVEA), en un repositorio para OAs o en otro sitio.

5) Fase de Evaluación: se propone cómo se va a evaluar el OA para determinar si sirvió o no para satisfacer la necesidad inicial. Puede ser por un sistema de valoración o comentarios de los usuarios.

4. Desarrollo de un OA para la Estructura de los Árboles AVL.

En esta sección se describen los pasos realizados para el desarrollo del OA para los árboles AVL. El primer intento fue la creación de un OA que abordara el tema completo tal y como se desarrolla en la asignatura; que expusiera las características del árbol AVL, su estructura y sus operaciones. De esta manera el OA resultaría demasiado extenso y complejo, contradiciendo, en consecuencia, las características deseables en un OA. Un OA con esa granularidad haría dificultoso su reúso en otros contextos y entorpecería el proceso de aprendizaje del alumno que lo utiliza.

Dado que el OA presentaba varios objetivos claramente distinguibles, decidimos dividirlo para que cada objetivo fuera alcanzado por un OA más pequeño y concreto.

OA1: tiene como objetivo que el alumno pueda comprender el concepto de árboles AVL y diferenciarlo de los ABB.

OA2: tiene como objetivo comprender el mecanismo de rotación en ABB. Cabe señalar que este diseño tuvo un esfuerzo adicional. Dado que se utilizan los mismos mecanismos de rotación, aunque con otra condición de balanceo en otros árboles binarios, por ejemplo, en los árboles Red Black [6,7] y Splay [5,9], debimos desarrollar el tema separándolo de los conceptos propios de los AVL. Esta separación facilitaría el reúso del OA para tales estructuras, ahorrando tiempo en su desarrollo.

OA3: tiene como objetivo entender el proceso de resolución de la operación de inserción en árboles AVL, reconociendo qué tipo de rotación se debe aplicar en cada caso de desbalanceo, según los criterios propios de la estructura AVL. Este OA referencia a los dos OAs anteriores, ya que tanto la definición de AVL como los mecanismos de rotación son requeridos como conocimientos previos.

OA4: es similar al anterior pero respecto a la operación de eliminación.

OA5: tiene como objetivo determinar cuándo aplicar AVLs a problemas específicos.

La Fig. 1 muestra estos cinco OAs y las dependencias que existen entre ellos.

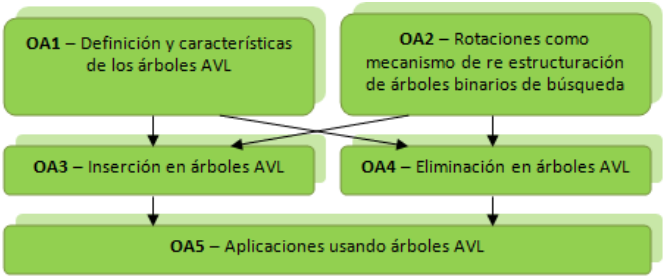


Fig. 1 - Los OAs que constituyen el tema árboles AVL.

Una vez definidos los OAs necesarios para abordar el tema, se los desarrollaron según la metodología CROA, siguiendo las 5 fases mencionadas. La fase de análisis de cada OA da origen a la definición de los objetivos que cada uno deberá alcanzar y fue necesario para dividir claramente el alcance de cada uno.

1) Fase de Análisis

En esta fase se plantean preguntas respecto a qué necesidad será satisfecha con la creación del OA y qué temas abordará. A continuación se muestra un resumen de la información generada en esta etapa para cada uno de los OAs.

OA 1 - Definición y características de los árboles AVL: Los alumnos necesitan aprender la estructura de los árboles AVL, conocer sus propiedades de orden y estructurales y diferenciarlos de los ABB. Por lo tanto, el OA presentará la estructura de datos de los árboles AVL diferenciándolos de los ABB, mostrando sus características y ventajas.

OA 2 - Rotaciones como mecanismo de re-estructuración de ABB: Los alumnos necesitan aprender cómo es el mecanismo de rotación en ABB. Por lo tanto, el OA presentará el mecanismo de rotación sobre estos árboles destacando que se mantienen las propiedades de orden y estructural.

OA 3 - Inserción en árboles AVL: Los alumnos necesitan aprender a insertar un elemento en el árbol AVL manteniendo las propiedades del mismo. Deben reconocer bajo qué condición se pierde la propiedad de balanceo y determinar qué rotación se debe aplicar para restaurar dicha propiedad. Referencia al OA anterior para dar detalle del mecanismo de rotaciones.

OA 4 - Eliminación en árboles AVL: Los alumnos necesitan aprender a eliminar un elemento en el árbol AVL manteniendo las propiedades del mismo. Deben reconocer bajo qué condición se pierde la propiedad de balanceo y determinar qué rotación se debe aplicar para restaurar dicha propiedad. Referencia al OA de Rotaciones para dar detalle de su mecanismo.

OA 5 - Aplicaciones usando árboles AVL: Los alumnos necesitan aprender a aplicar la estructura de árboles AVL para resolver problemas. Por lo tanto, el OA presenta un contexto integrador para los árboles AVL referenciando a los OAs anteriores.

Estos cinco OAs están orientados a alumnos de nivel superior. Los conocimientos previos necesarios son, además de los ABB, las dependencias que se establecen entre ellos. En la Fig. 1 se puede observar cómo las relaciones existentes entre OAs, reflejadas con las flechas que parten de un OA y llegan a otro, nos indican que el OA de partida es requerido y debe estar entre los conocimientos previos.

A continuación por razones de espacio completaremos las fases de la metodología sólo para el OA de Inserción. El diseño y desarrollo de los restantes OAs es similar.

2) Fase de Diseño

Detallaremos las tres actividades definidas en la fase de diseño para el OA de Inserción.

Diseño instruccional: Uno de los aspectos clave en el diseño de los OAs es la definición del objetivo y la selección de los contenidos de acuerdo al mismo. El objetivo de aprendizaje específico es resolver la operación de inserción en un árbol AVL reconociendo qué tipo de rotación se debe aplicar en cada caso de desbalanceo. Para la selección de contenidos se debió trabajar en la adaptación del material utilizado en las clases presenciales. Teniendo como base las presentaciones digitales de la materia y las explicaciones dadas por el docente, se debió transformar esa información para que el contenido del OA sea apropiado y suficiente para el autoaprendizaje. Cuando se trabaja con estructuras de datos, es muy importante comprender la dinámica de la misma en el proceso de resolución de operaciones, en este caso, la de inserción. Respecto a esto, utilizamos en el OA dos estrategias, una de observación y otra de experimentación. La primera permite ver los cambios en cada paso del proceso y es realizada por medio de videos cortos que muestran cómo se

realiza la inserción de un elemento particular dentro de un árbol dado. En estos videos se muestra cómo se ubica la posición del nuevo elemento, cómo se lo crea, la forma de desandar el camino e identificar y aplicar la rotación necesaria para el rebalanceo. Para la segunda estrategia se utilizó una aplicación que permite construir libremente un árbol, insertando un elemento a la vez, observando los cambios que se producen a medida que se forma.

Finalmente se diseñaron las demás actividades y las autoevaluaciones de acuerdo a los objetivos.

Diseño de la estructura: se pensó que la estructura más adecuada para este OA sería una estructura mixta, que permite navegar a partir de un nodo inicial de la forma que se muestra en el mapa de navegación presentado en la Fig. 2.

Diseño de los componentes multimediales: se seleccionaron plantillas adecuadas para la presentación de los contenidos ya mencionados.

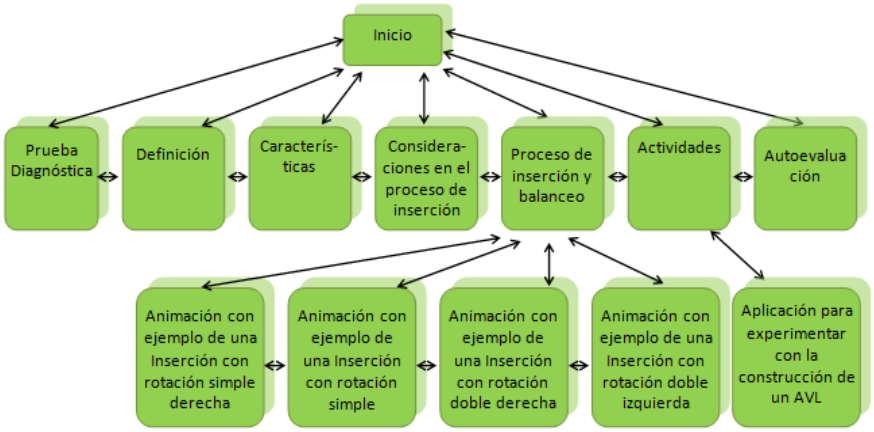


Fig. 2 - Mapa de navegación de los nodos pertenecientes al OA de Inserción en AVL.

3) Fase de Desarrollo

Para el desarrollo e implementación de los OAs se eligieron herramientas de software libre. Para la construcción se utilizó eXeLearning [11]; esta herramienta permite completar los metadatos en distintos modelos de datos, entre ellos LOM [12], el cual es el requerido por la metodología. Para las animaciones se utilizó OGV [13] por ser un formato de video abierto.

La aplicación de construcción fue adaptada a partir de una aplicación desarrollada por la Universidad de San Francisco, respetando sus licencias [14]. La adaptación fue necesaria para respetar los criterios adoptados por la cátedra respecto al tema.

Algunas actividades y evaluaciones fueron implementadas con Ardora [15].

La Fig. 3 muestra una pantalla del OA desarrollado desde la herramienta eXeLearning.

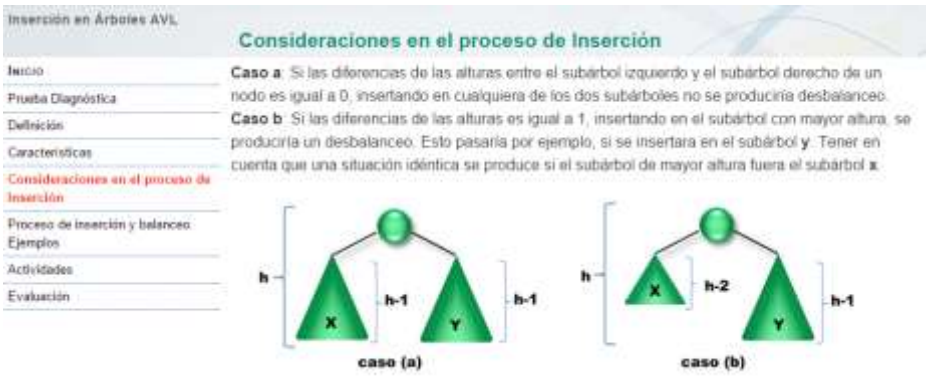


Fig. 3 - Vista del OA desde el entorno eXeLearning

4) Fase de Publicación

La publicación de los OAs se realizó en Moodle dado que es el entorno educativo virtual que ya estaba utilizando la cátedra como medio de comunicación y como espacio de consulta y desarrollo de actividades interactivas.

5) Fase de Evaluación

Para realizar una evaluación del impacto que tuvo el material en la asignatura, hemos realizado dos encuestas, una orientada a los docentes [16] y otra a los alumnos [17]. El objetivo fue evaluar los diferentes puntos de vista respecto al material e identificar los aspectos positivos y negativos según ambas miradas. Las encuestas fueron anónimas, pero se incluyó un campo opcional para el nombre, el cual la mayoría de los encuestados completó y nos permitió tener mayor confianza respecto a las respuestas.

Evaluación por parte de los docentes auxiliares de la materia

La encuesta dirigida a los docentes tenía por objetivo determinar la repercusión que tuvo en los alumnos este material complementario. Fue respondida por docentes auxiliares que asisten en las clases prácticas de la materia. Se consideraron sólo las respuestas de los que tienen más experiencia en la materia (más de 5 años). Únicamente 8 cumplieron este requisito. Se preguntó su opinión acerca de cómo tomaron los alumnos este material nuevo, si les ayudó en la comprensión del tema y si les ayudó a resolver dudas. Es decir, la visión del docente de la influencia de este material en el aprendizaje del alumno.

Una de las preguntas realizadas se refiere a si hubo un aumento o no de la cantidad de consultas respecto a años anteriores. El 75% respondió que la cantidad de consultas disminuyó y que no se realizaron consultas básicas (sobre las definiciones, los conceptos, etc.).

Otra pregunta se refiere al tipo de material presentado en los OAs. Ya mencionamos la importancia de poder visualizar la dinámica de los procesos que afectan a estas estructuras, el haber agregado los materiales multimedia que permiten realizar estas visualizaciones fue muy bien recibido por los alumnos. La mayoría de los docentes (87%) destacó la inclusión de contenidos multimedia como facilitadores del aprendizaje.

Evaluación por parte de los alumnos que asisten a la materia

La encuesta dirigida a los alumnos tenía por objetivo determinar si el material les fue útil en su proceso de aprendizaje, fue respondida por 44 alumnos. Se preguntó acerca de la cantidad de veces que consultaron el material (con esto se trató de distinguir a los alumnos que accedieron al material, de los que nunca lo hicieron, para valorar las respuestas de distinta forma). Se preguntó los motivos por los cuáles volvieron a consultar el material, y en qué aspectos les fue útil.

Con una de las preguntas se intentó determinar qué parte del material les resultó más útil. La mayoría de alumnos (81%) destacó que los elementos multimedia e interactivos les resultaron muy interesantes. El 75% respondió que pudo despejar dudas que le surgieron al momento de realizar el trabajo práctico, sin recurrir al docente auxiliar en las clases prácticas. También destacaron la utilidad de la aplicación para construir un AVL observando la dinámica paso a paso y tener la posibilidad de ejercitar esa construcción en forma autónoma, despejando cualquier duda que hubiera en ese proceso. Esto coincide con las respuestas dadas por los docentes que manifiestan que recibieron menos consultas respecto al tema y las que recibieron fueron más específicas, de un nivel de detalle mayor. Por último, los alumnos mencionaron la utilidad de contar con OAs para otras estructuras de datos.

Evaluación por parte de los otros docentes en cuanto al reúso del OA

No se ha podido aún realizar una evaluación en cuanto al reúso de los mismos. Se encuentra en proceso el desarrollo los OAs para árboles Red Black, que reusará el OA de rotaciones. Esto reducirá los tiempos y costos del desarrollo del nuevo material.

5. Conclusiones y trabajo futuro

En este trabajo presentamos la experiencia de construir OAs para árboles AVL con el objetivo de ser utilizados en la asignatura Algoritmos y Estructuras de Datos, en el contexto de la convocatoria realizada por la Facultad de Informática, siguiendo la metodología CROA.

En la cátedra teníamos la necesidad de contar con algún material que explique el tema de árboles AVL tal y cómo se dicta en la materia. La gran variedad y diversidad de materiales que se encuentran en Internet respecto al tema generaba confusión a los alumnos. A pesar de ser un tema acotado, aborda varios conceptos, por lo que decidimos dividirlo en varios OAs que fueran más concretos y puedan ser reusados en otros contextos.

Durante la construcción de los OAs se trabajó especialmente en el desarrollo de contenidos interactivos que despertaran el interés de los estudiantes y los motivaran; pretendimos que el material los guiara y los ayudara a consolidar el tema durante el proceso de aprendizaje a través de las actividades y evaluaciones. Que pudiera ser un punto de apoyo para los alumnos que se les dificulta combinar el trabajo con la Facultad, por razones de horario o de tiempo. Podemos resaltar que los OAs permiten a los alumnos realizar su propio proceso de aprendizaje a su ritmo particular y a los docentes plasmar en el OA la información que desean trasmitir a los alumnos, manteniendo los criterios propios y el estilo de enseñanza de la cátedra.

Luego de la publicación de los OAs se realizó una evaluación para conocer el impacto en los alumnos. Para ello se realizaron dos encuestas, con el objetivo de conocer las visiones de los alumnos y los docentes. De ambas se puede resaltar que los alumnos aceptaron con agrado el material (lo consultaron en varias oportunidades), lo que les permitió disipar dudas del tema puntual y de otros subyacentes. Además, manifestaron los beneficios de contar con un recurso que usara los criterios aplicados en la cátedra que les permitiera experimentar por ellos mismos, y verificar su comprensión del tema. Los docentes expresaron que recibieron consultas más específicas y menos sobre conceptos básicos, ya que los alumnos adquirieron esos conocimientos a través de los OAs.

Podemos concluir que fue una experiencia exitosa, tanto para los alumnos como para los docentes. Además cabe destacar que la metodología CROA facilitó el desarrollo de los OAs, guiándonos en su construcción y permitiendo documentar las decisiones tomadas en cada fase. En consecuencia esta experiencia la tomamos como un punto de partida para construir otros OAs para la asignatura. En un trabajo futuro inmediato, se reutilizará uno de ellos para la definición de otro tipo de árboles balanceados, como son los árboles Red Black.

Referencias

1. Sanz, C., Barranquero, F., Moralejo, L.: Metodología para la CReación de Objetos de Aprendizaje (Metodología CROA), http://croa.info.unlp.edu.ar
2. Convocatoria de la Facultad de Informática de la UNLP. http://ead.info.unlp.edu.ar/
3. Adel´son-Vels'kii, G. M., Landis, E. M.: An algorithm for the organization of information Soviet Mathematics Doklady 3, 1259-1263 (1962).
4. Aho, A., Hopcroft, J., Ullman, J.: Data Structures and Algorithms. Addison-Wesley (1983)
5. Weiss, M.A.: Data Structures and Algorithm Analysis (2nd Ed.). Addison-Wesley (1995).
6. Cormen,T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, Second Edition. The MIT Press-McGraw-Hill Book Company (2001).
7. Weiss, M.A.: Data Structures and Algorithm Analysis in Java. Addison-Wesley (2007).
8. Preiss,B.R.: Data Structures and Algorithms with Object-Oriented Design Patterns in Java. John Wiley & Sons (1998).
9. Wiener, R., Pinson, L.J.: Fundamentals of OOP and Data Structures in Java. Cambridge University Press (2001).
10. Mark Allen Weiss, http://users.cis.fiu.edu/~weiss/
11. eXeLearning.net version 2.0, http://exelearning.net/
12. LOM - IEEE LTSC.(2002). Draft Standard for Learning Object Metadata, http://129.115.100.158/txlor/docs/IEEE_LOM_1484_12_1_v1_Final_Draft.pdf
13. OGV, http://www.vorbis.com/
14. Aplicación para construir y visualizar los árboles AVL, https://www.cs.usfca.edu/~galles/visualization/AVLtree.html
15. webArdora.net, http://webardora.net/index_cas.htm
16. Encuesta para los docentes auxiliares de Algoritmos y Estructuras de Datos, https://docs.google.com/forms/d/1eGUe9Zbzr4A3LkqgiLrMxMF_Hbib1YoHGp8L4TKwgjQ/viewform
17. Encuesta para los alumnos de Algoritmos y Estructuras de Datos, https://docs.google.com/forms/d/11qPGaZyx27QulIMBjod2GVxc5KJWFdachDrbIHpUE6k/viewform

Requerimientos de competencias en el mercado laboral actual

Proyecto Perfil.AR

Claudia Russo[1], Mónica Sarobe[1], Marcela Dillon[2], Fernando Lopez Gil[1], Martin Calcaterra[1], Pedro Ochipinti[1], Hugo Ramon[1]

[1] Instituto de Investigación y Transferencia en Tecnología - Universidad del Noroeste de la Provincia de Buenos Aires
Roque Saenz Peña 456 - Junin (CP 6000)
crusso@unnoba.edu.ar, monicasarobe@unnoba.edu.ar, fernandolopezgil@itt.unnoba.edu.ar, martin.calcaterra@itt.unnoba.edu.ar, pocchipinti@unnoba.edu.ar, hugoramon@unnoba.edu.ar

[2]Dirección de Calidad Institucional - Universidad Austral
Mariano Acosta s/n y Ruta 8 – Derqui (CP1629)
mdillon@austral.edu.ar

Abstract: Los avances de la ciencia y la tecnología, el desarrollo de la comunicación y su impacto en la sociedad demandan a las instituciones de educación una nueva mirada en la formación de profesionales integrales. Para ello se requiere el rediseño de los perfiles académico-profesionales, bajo el enfoque de competencias que permitan el desarrollo de destrezas asociadas a la búsqueda, procesamiento y articulación de información; y una gran capacidad de innovación, que les permita estar abiertos a los cambios, y también ser generador de esos cambios.

Keywords: Competencias básicas, Formación por competencias, Perfiles educativos, Calidad, Diseño Curricular.

1. Introducción

La formación de profesionales no está solamente en el conocimiento específico de su ámbito de estudio, sino en la formación de virtudes transmitidas a través de vivencias durante toda la vida. Estos principios el Proyecto Tuning de la Unión Europea, los agrupó en 30 competencias genéricas.

El concepto de competencia surge de la necesidad de valorar no sólo el conjunto de los conocimientos apropiados (saber) y las habilidades y destrezas (saber hacer) desarrolladas por una persona, sino de apreciar su capacidad de emplearlas para responder a situaciones, resolver problemas y desenvolverse en el mundo.

Igualmente, implica una mirada a las condiciones del individuo y disposiciones con las que actúa, es decir, al componente actitudinal y valorativo (saber ser) que incide sobre los resultados de la acción. Por este motivo, pareciera que la mejor manera es que los estudiantes se formen y desarrollen en las competencias a lo largo de toda su carrera, embebidas en las materias, en forma transversal al plan de estudio. El diseño curricular debe hacer énfasis en el principio de aprender a aprender y de obtener experiencias útiles para el aprendizaje permanente, incorporando los valores humanos, como herramientas para aprender y seguir aprendiendo a lo largo de la vida en situaciones cambiantes y complejas.[1]

Finalmente, con el diseño del perfil académico-profesional del egresado se pretende formar sujetos capaces de desempeñarse proactivamente en un mundo globalizado, como miembros útiles de la sociedad.

2. Contexto

En este proyecto proponemos concebir las competencia como …"procesos complejos de desempeño con idoneidad en determinados contextos, integrando diferentes saberes (saber ser, saber hacer, saber conocer y saber convivir), para realizar actividades y/o resolver problemas con sentido de reto, motivación, flexibilidad, creatividad, comprensión y emprendimiento, dentro de una perspectiva de procesamiento metacognitivo, mejoramiento continuo y compromiso ético, con la meta de contribuir al desarrollo personal, la construcción y afianzamiento del tejido social, la búsqueda continua del desarrollo económico-empresarial sostenible, y el cuidado y protección del ambiente y de las especies vivas"…[2]

2.1. Competencias educativas básicas

Entre los exponentes más significativos del planteamiento del curriculum por competencias básicas se pueden señalar el proyecto Definición y Selección de Competencias (DeSeCo) de la Organización para la Cooperación y el Desarrollo Económicos OCDE (2002) [3], y la recomendación del Parlamento Europeo y del Consejo sobre las competencias clave para el aprendizaje permanente (2006) [4]. En el ámbito universitario es reseñable el proyecto Tuning (2003) [5]

Son dos enfoques distintos de plantear las competencias básicas. En la propuesta DeSeCo de la OCDE y en el proyecto Tuning se plantean como competencias básicas las "competencias genéricas", que también se denominan "generales, transversales o metadisciplinares", ya que son comunes a todas las áreas de conocimiento, mientras que en la propuesta del Parlamento Europeo se plantean como competencias básicas tanto las competencias genéricas que son comunes a todas las áreas de conocimiento, como aquellas competencias específicas de área que pueden tener uso interdisciplinar o disciplinar, pero que aunque específicas se consideran igualmente competencias básicas porque son imprescindibles para la vida.

2.2. Competencias básicas genéricas (enfoque generativo y procesual)

La OCDE en su proyecto "Definición y Selección de Competencias (DeSeCo, 2002) estudió la sociedad del conocimiento en doce países e identificó tres grupos de competencias clave que son interdependientes y que de forma progresiva se irán integrando en el proyecto PISA de evaluación. En este proyecto se define la competencia como "la capacidad de responder a demandas complejas y llevar a cabo tareas diversas de forma adecuada. Supone una combinación de habilidades prácticas, conocimientos, motivación, valores éticos, actitudes, emociones y otros componentes sociales y de comportamiento que se movilizan conjuntamente para lograr una acción eficaz". Sus rasgos diferenciales serían los siguientes: constituye un "saber hacer", esto es, un saber que se aplica, es susceptible de adecuarse a una diversidad de contextos y tiene un carácter integrador, abarcando conocimientos, procedimientos y actitudes.

Las universidades están en estos momentos en proceso de convergencia y adaptación de títulos y planes de estudio para la creación de un Espacio Europeo de Educación Superior (EEES) para el año 2010. Una de las líneas de acción es la adopción de un sistema de titulaciones fácilmente reconocibles y comparables. El proyecto Tuning [5], que está sirviendo de referencia para muchas universidades, propone el sistema de competencias como lenguaje común para describir los objetivos de los títulos y planes de estudio, así como referencia para la evaluación de los resultados de los alumnos. En concreto, la

Agencia Nacional de Evaluación de la Calidad y Acreditación (ANECA) ha tomado como base de referencia la propuesta Tuning para la formulación de las competencias transversales de las nuevas titulaciones en las universidades españolas. Diferencian las competencias genéricas y las específicas.
Competencias genéricas son aquellas que son compartidas y que pueden generarse en cualquier titulación. Competencias específicas son aquellas que están asociadas a áreas de conocimiento concretas.

Competentencias instrumentales	*Competencias interpersonales*	*Competencias sistémicas*
1. Capacidad de análisis y síntesis 2. Capacidad de organizar y planificar 3. Conocimientos generales básicos 4. Conocimientos básicos de la profesión 5. Comunicación oral y escrita en la lengua propia 6. Conocimiento de una segunda lengua 7. Habilidades básicas de manejo del ordenador 8. Habilidades de gestión de la información 9. Resolución de problemas 10. Toma de decisiones	11. Capacidad crítica y autocrítica 12. Trabajo en equipo 13. Habilidades interpersonales 14. Capacidad de trabajar en un equipo interdisciplinar 15. Capacidad para comunicarse con expertos de otras áreas 16. Apreciación de la diversidad y de la multiculturalidad 17. Habilidad de trabajar en un contexto internacional 18. Compromiso ético	19. Capacidad de aplicar los conocimientos en la práctica 20. Habilidades de investigación 21. Capacidad de aprender 22. Capacidad de adaptarse a nuevas situaciones 23. Capacidad de generar nuevas ideas (creatividad) 24. Liderazgo 25. Conocimiento de culturas y costumbres de otros países 26. Habilidad para trabajar de forma autónoma 27. Diseño y gestión de proyectos 28. Iniciativa y espíritu emprendedor 29. Preocupación por la calidad 30. Motivación de logro

Gráfico 1 -. Competencias genéricas

En esta misma línea se sitúan numerosos países europeos [6] Estas competencias básicas genéricas se caracterizan por su potencialidad de transferencia y su multifuncionalidad tanto en las distintas áreas disciplinares como en situaciones de la vida diaria, así como por su durabilidad en el tiempo. Estas competencias genéricas no se aprenden de forma aislada, sino integradas en los distintos escenarios de la vida (educativo, profesional-laboral, comunitario y personal). En el escenario de la enseñanza formal, tienen sentido en la medida que se integran en los contenidos de todas las áreas curriculares. Usando la terminología de B. Rey (1996) [7] se les puede llamar "generativas", en el sentido de que median en el proceso de enseñanza-aprendizaje, engendrando una infinidad de conductas adecuadas respecto a una infinidad de situaciones nuevas. Estas competencias genéricas no son directamente evaluables, sino mediante su transferencia en materias, contextos y situaciones concretas.

2.3. Competencias básicas genéricas y específicas (enfoque terminal)

La recomendación europea de formulación de un Marco de Referencia Europeo (2006) [8] define la competencia clave o básica como una "combinación de destrezas, conocimientos, aptitudes y actitudes y la disposición de aprender, además del saber cómo. Las competencias clave representan un paquete multifuncional y transferible de conocimientos, destrezas y actitudes que todos los individuos necesitan para su realización y desarrollo personal, inclusión y empleo. Estas deberían haber sido desarrolladas para el final de la enseñanza o formación obligatoria y deberían actuar como la base para un posterior aprendizaje a lo largo de la vida".

En este marco se definen las ocho competencias clave que se consideran necesarias para el aprendizaje permanente a lo largo de toda la vida, que, en relación con el desarrollo que se hace de las mismas en España y Francia, son las siguientes:

Unión Europea	*España/LOE*	*Francia/Socle commun*[21]
1. Comunicación en lengua materna 2. Comunicación en lenguas extranjeras	1. Competencia en comunicación lingüística	1. Dominio de la lengua francesa 2. Práctica de una lengua viva extranjera
3. Competencia matemática y competencias básicas en Ciencia y Tecnología	2. Competencia matemática 3. Competencia en el conocimiento e interacción con el mundo físico	3. Conocimiento de los principales elementos de la Matemática, y dominio de la cultura científica
4. Competencia digital	4. Tratamiento de la información y competencia digital	5. Dominio de las técnicas habituales de la información y comunicación
5. Aprender a aprender	7. Competencia para aprender a aprender	
6. Competencias sociales y cívicas	5. Competencia social y ciudadana	6. Adquisición de las competencias sociales y cívicas
7. Sentido de la iniciativa y espíritu de empresa	8. Autonomía e iniciativa personal	7. Adquisición de la autonomía y del espíritu emprendedor
8. Conciencia y expresión culturales	6. Competencia cultural y artística	4. Dominio de la cultura humanística

Gráfico 2 - Ocho competencias clave para el aprendizaje permanente

Entre estas competencias básicas hay algunas que tienen carácter más transversal, tales como la competencia para aprender a aprender, comunicación lingüística, tratamiento de la información y competencia digital, competencia social y ciudadana y, autonomía e iniciativa personal. Otras están más directamente relacionadas con áreas o materias concretas del currículo, como la competencia matemática, conocimiento e interacción con el mundo físico, competencia cultural y artística. Hay competencias, como por ejemplo, la comunicación lingüística y la social y ciudadana que en parte son genéricas y en parte específicas. El clasificarlas como específicas no quiere decir que el uso, por ejemplo, de las competencias matemáticas no sea relevante en otras áreas, sino que el aprendizaje básico de las competencias matemáticas corresponde al área de la Matemática.

3. Desarrollo del proyecto Perfil.AR

Perfil.Ar toma como base el proyecto “Tuning” [5] En este se vislumbraban dos problemáticas [8]

1. Por un lado la necesidad de modernizar, reformular y flexibilizar los programas de estudio de cara a las nuevas tendencias, necesidades de la sociedad y realidades cambiantes de un mundo vertiginoso.
2. Por otra parte, vinculado estrechamente con el anterior, la importancia de trascender los límites del claustro en el aprendizaje brindando una formación que permitiera el reconocimiento de lo aprendido más allá de las fronteras institucionales, locales, nacionales y regionales.

En (2004-2007) surge el proyecto Tuning América Latina, en su primera fase buscó iniciar un debate cuya meta fue identificar e intercambiar información y mejorar la colaboración entre las instituciones de educación superior, para el desarrollo de la calidad, efectividad y transparencia de las titulaciones y programas de estudio.
La nueva fase (2011-2013) parte de un terreno ya abonado fruto del desarrollo de la fase anterior y ante una demanda actual de las universidades latinoamericanas y los gobiernos de facilitar la continuación del proceso iniciado. La nueva etapa de Tuning en la región tiene por objetivo general, contribuir a la construcción de un Espacio de Educación Superior en América Latina. [9]

4. Metodología Inicial

Se trabajará con encuestas a diferentes niveles (docentes, alumnos, graduados y empleadores) para determinar una línea base de medición y para ver la importancia que se le da a las diferentes competencias.
Se realizarán conferencias y talleres a docentes, graduados y alumnos, con el fin de mostrar la necesidad de incorporación de las competencias básicas como temas de estudio.
Se trabajará en el desarrollo de nuevas actividades y de material para la formación de los estudiantes.[10]

NOTA: En la actualidad se realizaron las encuestas para relevamiento de información de empleadores y se están modificando las encuestas docentes ya que las mismas resultaron muy extensas. Se está trabajando en el formato de las mismas.

5. Encuesta a Empleadores

En base a las respuestas obtenidas en la encuesta que se realizó a empleadores se pudieron ver los siguientes resultados.

5.1. Análisis de la muestra

Sobre el total de encuestados, se puede ver que los mismos se distribuyen de la siguiente manera respecto del alcance que tienen las empresas a las que pertenecen:

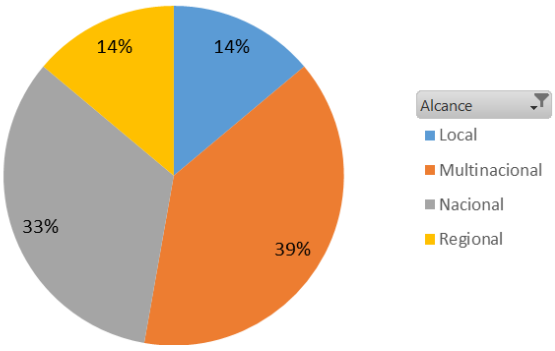


Gráfico 3 –Distribución de los encuestado por alcance de sus empresas

En el caso de considerar la actividad a la que se dedican las empresas, podemos ver que la distribución se encuentra dividida de la siguiente forma:

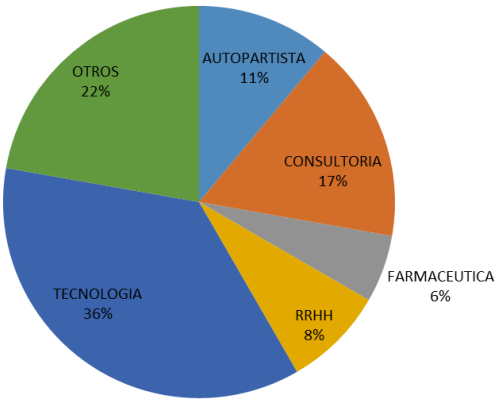


Gráfico 4 – Distribución de los encuestados por actividad

5.1. Resultados de las competencias requeridas

Dentro de la encuesta realizada, se pidió que indicaran para cada una de las competencias del trabajo, cuál era a su criterio, la importancia que ésta tenía. Las respuestas posibles eran irrelevantes, recomendables, necesarias e imprescindibles.
En base a estas respuestas, se realizó un ranking con el puntaje obtenido por cada una de las y se las ordenó de mayor a menor como se puede ver en la tabla 1.

Tabla 1 – Resultado de las competencias requeridas

Orden	Descripción	Competencia	Ranking
1	Compromiso ético	Competencia 25	302
2	Capacidad de trabajo en equipo	Competencia 16	300
3	Capacidad para identificar, plantear y resolver problemas	Competencia 14	298
4	Capacidad de aplicar los conocimientos en la práctica	Competencia 2	291
5	Compromiso con la calidad	Competencia 26	289
6	Habilidades interpersonales	Competencia 17	278
7	Capacidad de abstracción, análisis y síntesis	Competencia 1	278
8	Capacidad de comunicación oral y escrita	Competencia 6	264
9	Habilidades para buscar, procesar y analizar información procedente de fuentes diversas	Competencia 10	261
10	Capacidad para organizar y planificar el tiempo	Competencia 3	253
11	Capacidad de aprender y actualizarse permanentemente	Competencia 9	246
12	Capacidad para tomar decisiones	Competencia 15	242
13	Capacidad para actuar en nuevas situaciones	Competencia 12	236
14	Capacidad creativa	Competencia 13	224
15	Conocimientos sobre el área de estudio y la profesión	Competencia 4	223

16	Habilidad para trabajar en forma autónoma	Competencia 23	223
17	Capacidad crítica y autocrítica	Competencia 11	216
18	Responsabilidad social y compromiso ciudadano	Competencia 5	214
19	Capacidad de motivar y conducir hacia metas comunes	Competencia 18	210
20	Capacidad para formular y gestionar proyectos	Competencia 24	206
21	Capacidad de Negociación	Competencia 27	199
22	Valoración y respeto por la diversidad y multiculturalidad	Competencia 21	193
23	Capacidad de comunicación en un segundo idioma	Competencia 7	183
24	Capacidad de investigación	Competencia 8	178
25	Habilidad para trabajar en contextos internacionales	Competencia 22	177
26	Compromiso con su medio socio-cultural	Competencia 20	152
27	Compromiso con la preservación del medio ambiente	Competencia 19	148

En el gráfico 5 se puede observar la distribución de las respuestas obtenidas para cada una de las competencias ordenadas de acuerdo al mismo ranking.

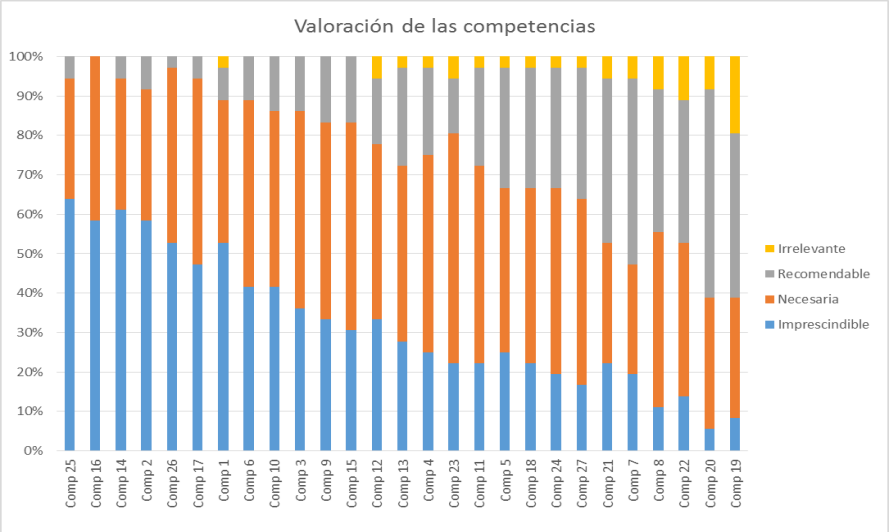


Gráfico 5 – Valoración de las competencias requeridas

1.2. Resultados de las competencias adquiridas

Por otro lado se le preguntó a los encuestados para cada una de las competencias, cuál era a su entender el nivel de desarrollo de la misma, con el que los graduados llegaban a trabajar a sus empresas.

Las respuestas posibles eran valores de uno a cinco, siendo uno nada desarrollada y cinco completamente desarrollada.

En este caso también se realizó un ranking con el puntaje obtenido por cada una de las competencias y se las ordenó de mayor a menor como se puede ver en la tabla 2.

Tabla 2 *– Resultado de las competencias adquiridas*

Orden	Descripción	Competencia	Ranking
1	Capacidad crítica y autocrítica	Competencia 11	127
2	Capacidad de trabajo en equipo	Competencia 16	127
3	Habilidades para buscar, procesar y analizar información procedente de fuentes diversas	Competencia 10	127
4	Capacidad de abstracción, análisis y síntesis	Competencia 1	125
5	Conocimientos sobre el área de estudio y la profesión	Competencia 4	125
6	Capacidad para identificar, plantear y resolver problemas	Competencia 14	122
7	Compromiso ético	Competencia 25	121
8	Habilidades interpersonales	Competencia 17	118
9	Habilidad para trabajar en forma autónoma	Competencia 23	118
10	Capacidad creativa	Competencia 13	116
11	Capacidad de aplicar los conocimientos en la práctica	Competencia 2	116
12	Capacidad para actuar en nuevas situaciones	Competencia 12	115
13	Capacidad de comunicación oral y escrita	Competencia 6	113
14	Compromiso con la calidad	Competencia 26	112
15	Capacidad para tomar decisiones	Competencia 15	109
16	Valoración y respeto por la diversidad y multiculturalidad	Competencia 21	108
17	Capacidad para formular y gestionar proyectos	Competencia 24	104
18	Capacidad de aprender y actualizarse permanentemente	Competencia 9	104
19	Capacidad de investigación	Competencia 8	103
20	Capacidad para organizar y planificar el tiempo	Competencia 3	103
21	Compromiso con su medio socio-cultural	Competencia 20	100
22	Compromiso con la preservación del medio ambiente	Competencia 19	99
23	Responsabilidad social y compromiso ciudadano	Competencia 5	98
24	Capacidad de Negociación	Competencia 27	97
25	Capacidad de motivar y conducir hacia metas comunes	Competencia 18	96
26	Capacidad de comunicación en un segundo idioma	Competencia 7	96
27	Habilidad para trabajar en contextos internacionales	Competencia 22	95

1.3. Análisis de las diferencias

En base a los resultados obtenidos respecto a las competencias adquiridas por los graduados y a las requeridas por la industria, se elaboró la siguiente tabla donde se pueden observar las diferencias en el ranking.

Tabla 3 *– Comparación de los resultados de ambos rankings*

Competencia	Orden Req	Orden Adq	Dif
Compromiso ético	1	7	-6
Capacidad de trabajo en equipo	2	2	0
Capacidad para identificar, plantear y resolver problemas	3	6	-3
Capacidad de aplicar los conocimientos en la práctica	4	11	-7
Compromiso con la calidad	5	14	-9
Habilidades interpersonales	6	8	-2
Capacidad de abstracción, análisis y síntesis	7	4	3
Capacidad de comunicación oral y escrita	8	13	-5

Habilidades para buscar, procesar y analizar información procedente de fuentes diversas	9	3	6
Capacidad para organizar y planificar el tiempo	10	20	-10
Capacidad de aprender y actualizarse permanentemente	11	18	-7
Capacidad para tomar decisiones	12	15	-3
Capacidad para actuar en nuevas situaciones	13	12	1
Capacidad creativa	14	10	4
Conocimientos sobre el área de estudio y la profesión	15	5	10
Habilidad para trabajar en forma autónoma	16	9	7
Capacidad crítica y autocrítica	17	1	16
Responsabilidad social y compromiso ciudadano	18	23	-5
Capacidad de motivar y conducir hacia metas comunes	19	25	-6
Capacidad para formular y gestionar proyectos	20	17	3
Capacidad de Negociación	21	24	-3
Valoración y respeto por la diversidad y multiculturalidad	22	16	6
Capacidad de comunicación en un segundo idioma	23	26	-3
Capacidad de investigación	24	19	5
Habilidad para trabajar en contextos internacionales	25	27	-2
Compromiso con su medio socio-cultural	26	21	5
Compromiso con la preservación del medio ambiente	27	22	5

6. Conclusiones y trabajo futuro

La evaluación de las encuestas realizadas a los empleadores, nos muestra un conjunto de desvíos entre los valores esperados en las competencias básicas de los graduados y los valores encontrados.
También nos hace ver el poco nivel de importancia que se le está dando en el mercado laboral a una serie de competencias, esto podría abrir una nueva línea de investigación.
Por otro lado, será necesario completar el análisis de las encuestas a docentes, graduados y alumnos para tener un panorama completo del estado de situación.
Se pretende a partir de estos resultados trabajar en los siguientes aspectos:

- Determinar las necesidades de formación en competencias de los alumnos de la Escuela, detectar las diferentes necesidades y desarrollar un programa integral de formación para poder desarrollarlas.
- Concientizar a los profesores en la importancia de la formación en estas competencias de un modo directo y transversal a lo largo de todo el ciclo de formación del futuro profesional
- Definir herramientas educativas que permitan desarrollar dichas competencias, tratando de lograr que los alumnos descubran la importancia que estas tendrán en su vida profesional
- Determinar en base a los aportes de las diferentes materias del plan de carrera cómo se está cubriendo la formación de los alumnos en un conjunto de competencias que no son específicas de las carreras.
- Determinar el estado de la formación actual de los estudiantes en las diferentes competencias básicas.
- Determinar la importancia que se le asigna desde el mundo laboral a ciertas competencias en las que se pretende formar a los alumnos de la Escuela.

- Desarrollar actividades de formación que contribuyan al desarrollo de dichas competencias.
- Desarrollar material y casos de estudio, que permitan el desarrollo de las competencias básicas, asociados a las diferentes áreas del núcleo curricular de las carreras informáticas (según Res 786/09).
- Crear puentes entre las universidades y otras entidades apropiadas y calificadas para producir convergencia en el área de la disciplina seleccionada

Dentro de la Escuela de Tecnología, se pretende propiciar un marco formal para esta línea de investigación, atendiendo a la formación de recursos humanos, en su rol de investigadores o partícipes activos en el equipo de investigación; fomentando la culminación de sus estudios superiores, promoviendo la redacción, exposición y defensa de Trabajos Finales de Grado y Postgrado.

7. Bibliografía

1. Claudia Russo, Mónica Sarobe, Marcela Dillon y otros (2015) "El desafío de las nuevas tecnologías de la información y las comunicaciones en los contextos educativos" WICC 2015. Universidad Nacional de Salta. Abril 2015.
2. Tobón, S. (2008). La formación basada en competencias en la educación superior: el enfoque complejo. México: Universidad Autónoma de Guadalajara.
3. OCDE (2002): Définitions et sélection des compétences. Fondements théoriques et conceptuels Documents de strategie. DEELSA/ED/CERI/CD(2002)9
4. Recomendación del Parlamento Europeo y del Consejo, de 18 de diciembre de 2006, sobre las competencias clave para el aprendizaje permanente (2006/962/CE)
5. TUNING (2003): Tuning Educatinal Structures in Europe. Final Report, Phase One. University of Deusto/University of Groningen. www.relint.deusto.es
6. Cfr. CARRO, L. (2004) : « Los grados de magisterio en los 25 países de la Unión Europea". En: Título de grado de Magisterio. Voumen 2. Agencia Nacional de Evaluación de la Calidad y Acreditación (ANECA) GARAGORRI, X. (2007): "Propuestas curriculares basadas en competencias en el ámbito europeo".Aula de Innovación Educativa, nº. 161
7. Rey, B. (1996). Les compétences transversales en question . Paris: ESF
8. Pablo Beneitone y otros (2007) "Reflexiones y perspectivas en Educación Superior en América Latina. Informe Final – Proyecto Tuning LA". Universidad de Deusto – Universidad de Groningen. España.
9. José Lino Contreras Véliz y otros (2013) "Educación Superior en América Latina: reflexiones y perspectivas en Informática." Universidad de Deusto. España.
10. Claudia Russo, Mónica Sarobe, Marcela Dillon y otros (2015) "Desarrollo de perfiles profesionales en base a competencias Perfi.AR" TE&ET 2015 Universidad Nacional del Nordeste. Junio 2015.

Pedagogical Practices with Social Impact in the Computer Science School of the National University of La Plata: Interdisciplinary Articulation for the Development of a Digital Communicator for Autistic Children

Claudia Queiruga, Ivana Harari and Jorge Rosso

LINTI (Laboratory of Research in New Information Technologies), Computer Science School, National University of La Plata, 50 and 120 (Second Floor), La Plata (1900), Argentina
{claudiaq, iharari, jrosso}@info.unlp.edu.ar

Abstract. The involvement of public universities in the solution of problems in society and the participation in the processes of social and production development of the country are goals of university education in Argentina. The use of ICTs as instruments that contribute to Independence, quality of life and inclusion of persons with disabilities is a topic of concern for many disciplines worldwide. This paper presents HERMES, a project involving the articulation of two subjects of the Computer Science School of the National University of La Plata – "Diseño Centrado en el Usuario" ("User-Centered Design") and "Laboratorio de Software" ("Software Laboratory") and the civil organization CEDICA, a referent in the province of Buenos Aires regarding the application of AAT (Animal-Assisted Activities) among people with disabilities (physical, cognitive, sensory, psychological and social). HERMES consists on the development of a digital communicator based on pictograms, available for tablets, aimed at children and teenagers with an Autism Spectrum Disorder (ASD) attending CEDICA. This project involves students and teachers of both subjects and CEDICA professionals, families, and children and teenagers with an ASD, in the process of design and development of the communicator. Currently, the first prototype of HERMES is under evaluation. We conclude that this type of initiatives tend to complement the academic training of students, promoting an open and integrating culture within the computer science community, accepting diversity in all contexts and contributing to the solution of the problems in its community.

Keywords: inclusion, autism, mobile applications, disabilities, education, Android, JAVA

1 Introduction

National universities in Argentina represent the maximum level of education in the country; they are characteristic for being public, free of charge, free of access, autonomous and co-governed. The educational system of Argentina has been acknowledged by international organizations such as UNESCO and UNICEF. The National University of La Plata (UNLP) is one of the 53 public universities of Argentina, and one of its main objectives is to contribute from the university community to searching for answers to social problems, particularly to those arising in the most vulnerable sectors of society, and to participate in the processes of social and productive development of the country. In this sense, university education and knowledge are understood as a social and public asset. For this reason, policies must be established at all ranks and roles of the educational entity that realize concrete lines of action.
From the beginning, the Computer Science School of the UNLP has continuously undertaken extension projects with a social impact through joint work with social organizations and public institutions, among them social diners, associations of persons with disabilities, elderly people, secondary and technical schools, hospitals, etc. This outreach to the community strengthens bonds that allow the university to pay attention to, learn about and tend to demands and put computer science at their service, generating a great impact to the inside of the School that goes beyond transference. These experiences not only allow us to discover the scope of

technology and its impact on the quality of life of the members of marginalized sectors of society and of persons with disabilities, but also extend and manifest themselves in multiple ways such as their incidence in the academic formation of the courses of the School.
In this context and with the goal of adding social responsibility in the pedagogical practices of the Computer Science School of the UNLP, we have undertaken the HERMES project. This project consists of the development of a digital communicator for children and teenagers with an Autistic Spectrum Disorder (ASD) who attend CEDICA[1].
For the purpose of this project, two subjects of the last years of the B.S. in Computer Science and the B.S. in Systems had to work in an articulated manner – "Diseño Centrado en el Usuario" ("User-Centered Design") and "Laboratorio de Software" ("Software Laboratory"). In a joint and interdisciplinary manner with CEDICA professionals, its design, development and testing was performed through practical and articulated activities, respecting adequate software development engineering software based on user-centered design, where usability and accessibility matters were dealt with from the initial stages of development (Garrett Jesse, 2011) (Lowdermilk Travis, 2013).
This paper presents the HERMES project: the context in which this proposal took place, the work to articulate both subjects of the Computer Science School courses and the results obtained from this experience analyzed from multiple perspectives: from the academic, from transference and also from the personal viewpoint. The involvement of students and teachers of the school in real projects with social impact and a common goal makes it a morally enriching learning experience, allows the training of future professionals that are competent, upright, caring and committed to the reality of their community and its diversity.

2 The Social as a Cross Axis in Strategic Lines of the Computer Science School of the UNLP

The Computer Science School of the UNLP has been actively working on different extension, research and volunteering projects related to several topics, among them education, digital inclusion, environment and accessibility (Díaz Javier et al, 2012a). These projects were validated both by calls of the Ministry of Education of the Nation and by the UNLP itself and have as their goal to contribute to the solution of concrete problems coming from diverse sectors in society, put forth in most cases by social organizations and public institutions of the region. Among the projects that aim at linking university and community, we can mention, among others, "eBasura: reutilización tecnológica y concientización ambiental"[2], "Informática Inclusiva"[3], "Impulsando la Ley 26653 sobre Accesibilidad Web a escuelas técnicas"[4], "Acercar la programación a la escuela secundaria"[5](Javier et al, 2011) (Díaz Javier et al, 2012b) (Díaz Javier et al, 2014).
This policy carried out in theses years of continuous communication and interaction with the community has allowed for the creation of a strong bond between the School and society and has allowed the School to listen to its demands and transform them into concrete interventions. This process slowly adapted and extended and even evolved within the classroom space, affecting the pedagogical practices of the School and directly impacting the academic training of the students.
In this sense, 2014 saw the development of the HERMES project, a digital communicator aimed at persons with ASD, carried out together with CEDICA professionals and persons involved, specifically the children that would use it, their families, special tutors, therapists and other professionals. Carrying out the project required an interdisciplinary coordination and articulation process between the subjects of the School and CEDICA, for the purpose of its full

[1] CEDICA: Centro de Equitación para Personas con Discapacidad http://www.cedica.org.ar/, https://www.facebook.com/cedica.equitacionparatodos
[2] Official website of eBasura: http://e-basura.linti.unlp.edu.ar/
[3] Official website of Informática Inclusiva: http://brechadigital.linti.unlp.edu.ar/
[4] Official website of Dirección de Accesibilidad: http://accesibilidad.linti.unlp.edu.ar/
[5] Official websites about taking programming to high school: http://robots.linti.unlp.edu.ar/ and http://jets.linti.unlp.edu.ar/

development.

3 The HERMES Project

The HERMES Project arose from the bond between the Computer Science School and CEDICA since 2012, which has resulted in multiple projects of use of technology for treating issues related specifically to disabilities in children and adolescents. Together with CEDICA, the Computer Science School developed the first videogame on Equine-Assisted Activities and Therapies (a special type of Animal-Assisted Activities and Therapies, AAAT) called eQuino (Díaz Javier et al, 2013). Likewise, the Computer Science School, together with CEDICA, APADEA[6] and Asociación Azul[7] participated in multiple events such as the "Maratón de Desarrollo para la accesibilidad" (Marathon of Development for Accessibility) organized by the Accessibility Department of the Computer Scienec School and the "Accesibilidad y Universidad. Ideas para la Acción" (Accessibility and University. Ideas for Action) sessions organized by the Secretary of Institutional Relations of the UNLP. Both events took place on the International Day of Persons with Disabilities, on December 3, 2013 and 2014, respectively.

From this continuous and progressive bond with CEDICA and APADEA came a set of demands in relation to the use of technology to assist and foster the independent lives of persons with disabilities. One of the needs put forth was having a digital communicator that serves as a support tool for the communication between therapists/parents and children/students with an ASD attending CEDICA. Currently existing digital communicators are not always free, they cannot be customized as these users need, they must be installed in a dedicated device, they were not thought for our cultural environment, which may result in them using vocabulary and expressions that we do not use, and are generally unavailable for mobile devices such as tablets and smartphones. This makes most institutions continue to use the traditional communicator in the form of a folder with drawings or paper photographs that children carry with them at all times. These folders are usually organized into categories and, depending on the progress of the child, may contain a large amount of sheets. The greatest problem observed with this paper device is that with time the sheets may go missing, deteriorate, or break; moreover, it is impossible to associate the pictograms with sounds that may encourage speaking, and customizing them is more complex than in a current mobile device.

This analysis resulted in proposing HERMES, a digital communicator for tablets that encourages the communication of children with autism by means of representative pictogram that are translated by the program into auditory and textual expressions when selected. It is a simple piece of software that the children themselves can use to express desires, tastes and emotions they cannot express verbally.

The full realization of the device involved other features, such as configuring files and adapting voice, developing a desktop application for therapists or parents to connect to and access the activity log for the child's tablet.

3.1 HERMES Features

The name HERMES refers to the homonymous god of Greek mythology who was characteristic for his ability to "communicate, announce, say, confront, and make clear with words"; he is the god that announced the messages from the gods, given his skills and eloquence.

HERMES is an Augmentative and Alternative Communication System (AACS) (Tamarit Javier, 1992). Augmentative systems allow us to "increase" the communication capabilities of persons who have impediments to achieve functional verbal communication. In cases in which verbal expression is not possible, these systems will substitute it by means of artificial speech, and are thus called "alternative" means of communication. Augmentative and Alternative

[6] Sitio oficial de APADEA: http://apadea.org.ar/

[7] Sitio oficial de la Asociación Azul: http://www.asociacionazul.org.ar/

Communication (ACC) is not incompatible but complementary with rehabilitation of natural speech. It includes multiple symbol systems that adapt to the needs of persons with very different ages, motor skills, cognitive skills and linguistic skills. These symbols can be graphic (photographs, drawings, pictograms, words or letters), or gestural (mimic, gestures). The first group requires the use of support technological devices or products.
The main goal of the HERMES project is to develop a Pictographic System of Communication that is easy to interpret for children and teenagers with an ASD that attend CEDICA. This system consists of two applications: the digital communicator application and the monitor application.
The digital communicator application is aimed at being used primarily by the child and to a lesser degree by the therapist who is in charge of customizing it to the needs of each child. Tablets are the idea device because they are easy to interact with (tactile interaction), to transport (small, light), and the size of the screen is adequate for display.
The monitor application is a desktop application for exclusive access of the parents, tutors or professional thatwish to receive notifications coming from the digital communicator application. Through the monitor application, users have access to the activity log that includes the expressions activated by the child. Thus, it is possible to detect repetitive behaviors, analyze the activities chosen and the moods associated with time and activities, which would allow parents to have instant knowledge of what the child expresses when in school or in therapy.

3.2 The Communicator Application

The communicator application works in a tablet and allows the child or teenager to communicate with their therapist, teacher or family member through a choice of pictograms. These pictograms are classified according to whether they express emotions, moods, bodily needs, feeding needs, games or other matters the child may want to express. Likewise, the pictograms have texts and audios associated in order to emphasize on orality and in the transmission of the message.
The communicator has a tutor profile that allows the therapist to adapt and adjust the content to the characteristics of each child.
The strategy and form of support chosen for interaction with the communicator will be through direct selection by touch. After the chosen symbol has been transmitted the therapist can help the child make a dependent sweep or exploration, presenting them with multiple possible actions if necessary.
There are galleries with predefined pictographic symbols to which new icons (photographs or JPG/PNG images) can be added by the tutor/therapist user by selecting them from the application configuration menu.
Interviews conducted with CEDICA therapist resulted in a decision to organize pictographic symbols into categories based on the function of the symbol, making it possible to configure the amount and size of the symbols displayed. Likewise, each symbol has an associated audio file that represents its meaning in oral form.
Pictograms have a time attribute associated to them (morning, noon, afternoon or night) that allows for dynamic generation of the communicator screen depending on the time of use. Thus, the application allows pictograms that will logically be used in a specific time of day to be easily found (e.g., the pictogram “Go to the park” will be kept in the morning or afternoon time frames, so the application will not propose it at night).
The tutor profile allows for the configuration of different pictogram galleries for each user, which makes it possible to switch the set for pictograms by selecting the gallery associated with each child that will use the communicator. In order to facilitate the construction of pictogram galleries, the communicator provides a default gallery and the possibility to build new ones from a copy of one of the existing galleries (the default gallery or an existing customized gallery).
In the first version of the communicator, the galleries are composed of four pictogram categories:

- Acciones (Actions)

- Animales (Animals)
- Emociones (Emotions)
- Alimentos (Foods)

The HERMES communicator is a native Android application developed in JAVA compatible with versions 2.2 onward. Android is an operating system for mobile devices based on the Linux kernel, a free of charge, free of use and multiplatform operating system core. Android is distributed under the free software license Apache version 2 or ASL 2.0 (Apache Software License).
Fig. 1 shows the initial screen of the application with the pictograms in category "Acciones" (Actions) of the default gallery.
The communicator is organized into 4 sections as seen in Fig. 1: "Acciones" (Actions), "Animales" (Animals), "Emociones" (Emotions) and "Alimentos" (Foods), presented in tabs which allow access by touch. The user can also switch between sections by sliding the screen towards the section they desire. This type of interaction is known as "swiping" in Android terminology. Each section has two buttons: "yes" and "no" that allow the therapist to continue the dialog once a pictogram is chose, e.g. if the child chooses "Pasear" (walk) the therapist may ask if they want to do it with a horse.

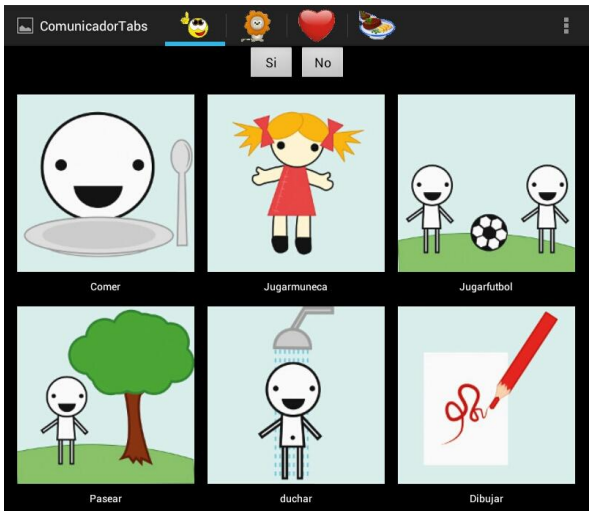


Fig. 1. List of images in the Acciones category

The available pictograms clearly represent the concept they wish to communicate, they are differentiable from each other and easy to understand. The attributes of a pictogram are as follows:

- Image (PNG, JPG, GIF, TIF)
- Related audio (.mp3, optional)
- Descriptive text
- Time (What time frame it belongs to)
- Category it belongs to
- Gallery it belongs to

Once the child "touches" a pictogram, it is marked as selected and the associated audio is played, and a notification is sent to the monitor application to record the interaction.

3.3 Configuring the Communicator Application

The communicator application provides adaptation and configuration features that allow for configuring the tool according to the needs of the child. This feature is available to the therapist or professional.
The configuration mode can be accessed from the communicator application itself through a

menu that allows therapists to create and edit pictogram galleries that are suitable for each child, also allowing for the definition of general configuration parameters such as the IP of the monitor and the values of the time intervals that will be associated to the pictograms.
Fig. 2 shows the gallery management screen from which it is possible to view existing galleries, edit them, delete them and create a new one.

Fig. 2. Gallery management screen

As mentioned in section 3.2, new galleries are created from existing ones, thus facilitating the formation of pictograms seizing similarities among the galleries of different children. Fig. 3 shows the dialogue box for the creation of a new gallery in which the user is asked to name it and state from which gallery it will be generated (list of existing galleries).

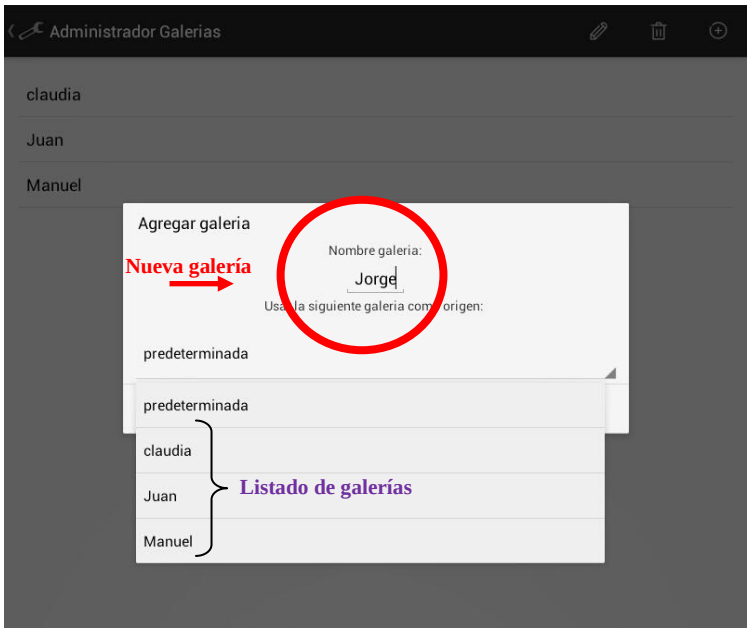


Fig. 3. Dialogue box for the creation of a new gallery

By selecting a gallery in the gallery management screen it is possible to edit it: add pictograms, edit them (change the descriptive text, the time associated, the image and the category it belongs to) and delete them from that gallery.
Fig.4 shows the edition of a pictogram from the gallery being edited.

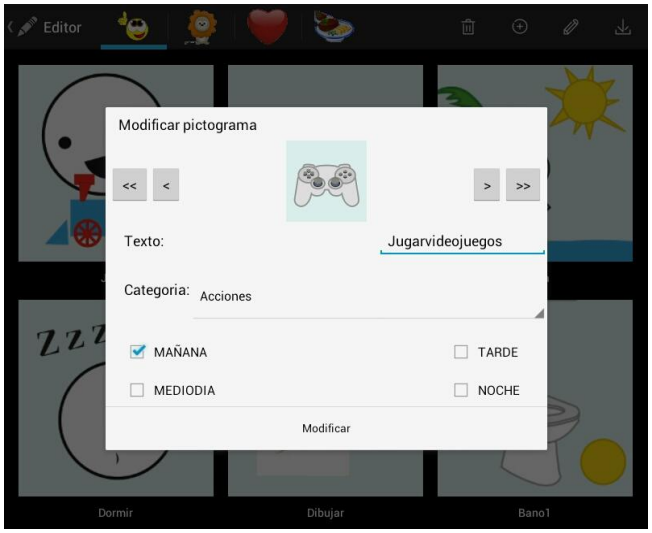


Fig. 4. Pictogram Edition Screen

Regarding general configurations, the therapist can: define the location of the monitor application in the network (IP address and port) as shown in Fig. 5, and, as shown in Fig. 6, establish the default gallery (the gallery that the application will show when it starts) and define the time interval parameters (e.g., the time tomorrow will be from 7 AM to 12 PM).

Fig. 5. Network configuration for location of the monitor

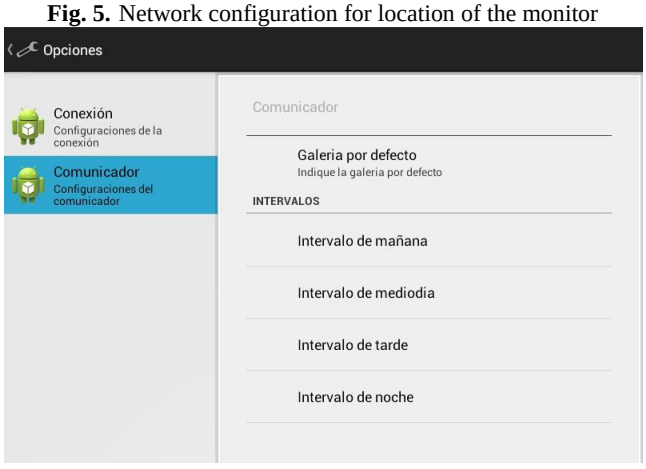


Fig. 6. Establishment of default gallery and definition of time intervals

3.4 The Monitor Application

The monitor application is meant to be used exclusively by therapists, parents or tutors. The monitor receives notifications of interactions between the child and the communicator.
The notification mechanism is unidirectional, i.e., from the communicator to the monitor, thus it behaves like an inbox where notifications arrive and are queued for later examination by

means of filters and statistics.
Each notification sent includes the following information:

- Name of the gallery active at the time of sending the notification
- Category to which the chosen pictogram belongs
- Pictogram descriptive text
- Date
- Time

The monitor is a desktop JAVA application. Fig. 7 shows the screen of the monitor with the notifications of the gallery called María filtered by the category Sentimientos (Feelings).

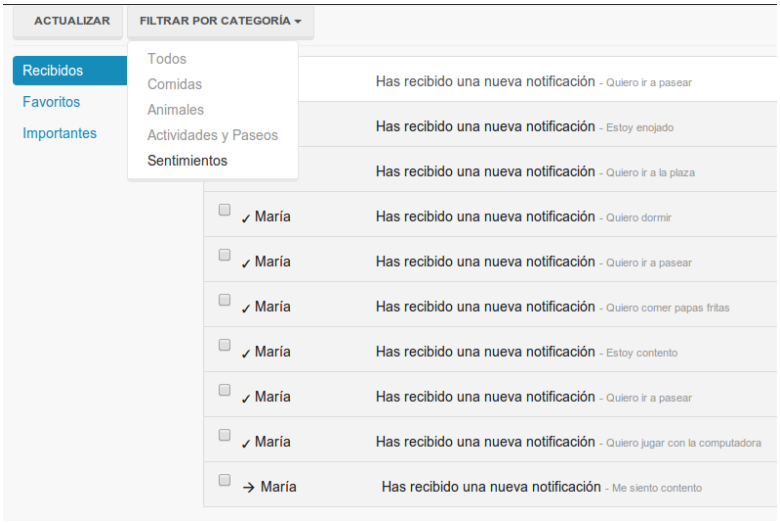


Fig. 7. Screen of the Monitor Application

Monitor notifications can be viewed in multiple ways and there are functionalities to: define labels, associate labels to notifications by date, by category, and by label and order them by multiple criteria. This information can be relevant to the therapist and allow them to detect patterns in behavior or taste and then apply these in the form of adjustments to therapy. It is possible to elaborate statistics from the information gathered by the monitor that can also be valuable to the professional.

4 Interdisciplinary Articulation for the Development of HERMES

The HERMES project was developed by means of an appropriate coordination and articulation between teachers and students of the "Diseño Centrado en el Usuario" and "Laboratorio de Software" subjects together with CEDICA professionals and the families of the children attending this institution. This pedagogical innovation involved the systematized tasks in Table 1.

Activity	Actions	In charge
Preliminary meetings in CEDICA	-Identification of the problem -Identification of the implications and whether the discipline could work on these issues.	Teachers of both courses and CEDICA professionals.
Interdisciplinary meetings	-Analysis of the proposal -Organization of the practical applications	Teachers of both courses

	-Division of the work for each subject	
Requirement analysis	Surveys, interview, analysis of the information in CEDICA	Students of “Diseño Centrado en el Usuario”
Design	Sketches, tests for their adequacy with teachers, therapists and children attending CEDICA.	Students of “Diseño Centrado en el Usuario” and a Digital Artist (a student of the Fine Arts School)
Prototype development	-Study of the technology to be used -Analysis of sketches -Understanding of the case -Development of the communicator and the monitor	Teachers and students of “Laboratorio de Software”
Testing the applications	-Study of the adequacy of the pictograms and the interaction with them -Survey regarding changes, criticism and improvement suggestions	Teachers and students of “Laboratorio de Software” and CEDICA professionals.

Table 1. Synthesis of the activities

5 Results of the Experience

The experience was very positive for students and teachers of both subjects, who had the possibility of participating in an innovative technological development project, with real users and with the noble goal of contributing to an improvement in the quality of life of persons with disabilities.
As regards the prototype obtained, the HERMES project is currently being evaluated by therapists and the families of the children attending CEDICA, with satisfactory results.
The evaluations conducted resulted in some proposals for the improvement of the communicator, which included:

- Creating categories dynamically following the progress of each child.
- Incorporating pictograms related to the equine therapy activities themselves as performed in CEDICA, in order to use the communicator during class.
- Contextualizing the communicator so the child can use it at home, at CEDICA, at school, etc. The pictograms shown in each situation will change depending on where the child is.

6 Conclusions

The goal of the development of the HERMES project is to achieve a commitment of Computer Science School teachers and students to different projects that have social impact. These actions are ranked positively by the community because they contribute answers to real problems and are likewise relevant to the training of our students, as they constitute concrete actions with real applications.
Not only does the public university have the obligation to train its students in academic topics, but their professional formation must include aspects related to social responsibility that will

help them become responsible citizens and generate in them competences that are heavily demanded in multiple sectors.
Lines of action in the Computer Science School, both in extension and in the pedagogical and academic area, such as the one that is the object of this paper, tend to complement academic training of the students, promoting an open and integrating culture within the computer science community, accepting diversity in all contexts. Computer scientists must be constituted as subjects capable of adjusting to the demands of society, taking consideration of the quality, social responsibility and ethics in their creations.
The competences specific to the computer science discipline must be taught in combination with the formation of a responsible being, aware of their role as developer of computing tools and applications, which must be within reach and in the service of the community as a whole and attending to local and regional demands.

References

Díaz Javier, Banchoff Tzancoff Claudia, Harari Viviana, Harari Ivana and Ambrosi (2011): "Accessibility, digital divide and environment: strategies in the Informatics Faculty, National University of La Plata", E-society 2011. Portugal.

Díaz Javier, Banchoff Tzancoff Claudia, Harari Viviana, Ambrosi Viviana, Harari Ivana and Raimundo Marcelo (2012a): "Improving University Education in Computer Science through Community Extension and Volunteering Activities". Innovations 2012, World Innovations in Engineering Education and Research.

Díaz Javier, Harari Ivana, Ambrosi Viviana, Banchoff Tzancoff Claudia and Harari Viviana (2012b): "Articulating two social based informatics projects", International Conference on Information Technologies 2012 (ICIT 2012) Saratov, Russia. June 2012.

Díaz Javier, Queiruga Claudia, Arce Tomás (2013): "e-Quino: an Interactive Videogame to Complement Equine Therapy". Proceedings of the 8th Immersive Education Summit (iED 2013, Boston MA. Boston: Immersive Education Initiative. ISSN 2325-4041.

Díaz Javier, Banchoff Tzancoff Claudia, Queiruga Claudia, Martín Eliana Sofía (2014): "Experiencias de la Facultad de Informática en la Enseñanza de Programación en Escuelas con Software Libre", Proceedings of "Congreso Iberoamericano de Ciencia, Tecnología, Innovación y Educación". Organized by Organization of Ibero-American States (OEI). ISBN 978-84-7666-210-6. Artículo 1426. Buenos Aires, Argentina, 12 - 14 November, 2014.

Garrett Jesse James (2011): "The Elements of User Experience: User-Centered Design for the Web and Beyond, Second Edition". Ed: New Riders

Lowdermilk Travis (2013): "User-Centered Design". Ed: O'Reilly Media

Tamarit Javier (1992): "¿Qué son los Sistemas Alternativos de Comunicación?". Sistemas alternativos de Comunicación, Universidad Nacional de Educación a Distancia, March, 1992.

Entorno para Experimentación de Vulnerabilidades en la Enseñanza de Buenas Prácticas de Programación

Uri Yael y Benjamín Barán

Universidad Nacional de Asunción

Resumen Este trabajo propone un entorno sencillo y de fácil utilización para un instructor que permite a los alumnos experimentar con vulnerabilidades de seguridad en un contexto controlado, consolidando sus conocimientos teóricos de cursos anteriores y/o complementando lo aprendido en la carrera, con laboratorios prácticos que complementen el proceso de aprendizaje que muchas veces se limita a presentaciones teóricas por falta de recursos como tiempo o equipos. Se reportan resultados prometedores con alumnos de carreras universitarias en informática y electrónica.

Keywords: educación, seguridad, inyección de código, XSS, CSRF

1. Introducción

Los cursos de seguridad de la información deben cubrir una vasta cantidad de áreas y temas pero el tiempo disponible muchas veces impide ahondar suficientemente en todas las áreas, por lo que muchos autores intentan confeccionar guías generales que puedan ser aplicadas en forma transversal [17]. Es crucial seleccionar buenas técnicas pedagógicas para que los estudiantes puedan adquirir los conocimientos necesarios en un espacio de tiempo razonable [8,17].

1.1. Contexto y Motivación

Al momento de crear sistemas informáticos, la seguridad debe ser un objetivo primario desde el mismo inicio. En gran cantidad de casos, los sistemas que no tuvieron dicho objetivo ya en la etapa de diseño, deben luego pasar por reestructuración y diversos cambios, posiblemente con un importante costo en recursos (*e.g.* tiempo de desarrollo, atrasos en fechas límite, dinero) [14].

Los errores de código que pueden considerarse *bugs* clásicos, cuya detección puede ser automatizada y validada en sucesivas verificaciones, no son el único motivo por el que existen vulnerabilidades. Otras fuentes de vulnerabilidades –que pueden derivar en huecos de seguridad– son los errores conceptuales en protocolos e interacciones, susceptibilidad a la falta de autenticación, autorización o el chequeo de integridad.

En cierto sentido, las técnicas de desarrollo de software generalmente enseñadas en los cursos regulares son inadecuadas para crear programas seguros debido a que se centran en la correctitud. Una definición de correctitud es: "*cumplimiento de las especificaciones que determinan cómo los usuarios pueden interactuar con el sistema y cómo debe comportarse el sistema cuando es utilizado en forma correcta*". Mientras la correctitud se encarga de las funcionalidades del sistema, la seguridad se encarga de la falta de funcionalidades. Es decir, se intenta validar que sin importar las acciones llevadas a cabo, no se podrán efectuar ciertas acciones. Las comprobaciones automatizadas son métodos para validar la existencia de una funcionalidad, pero no existe un método que asegure la inexistencia de una funcionalidad.

En opinión de Matt Bishop y Jacob West [16], existen diversos mitos por los cuales muchas instituciones educativas no aumentan los esfuerzos por incorporar en la práctica, la enseñanza de codificación segura, entre ellos: (a) no hay espacio en la malla curricular. No se necesitan más cursos, sino revisar el contenido de los ya existentes; (b) si los alumnos aprenden a escribir programas seguros, cuando salgan al mercado laboral ¿aceptarán las compañías el aumento del costo y tiempo que toma crear un programa?, ¿los clientes pagarían precios más altos?, ¿realmente se requerirá a los alumnos que pongan en práctica lo que aprendieron?; y (c) la creencia de que ya lo estamos haciendo. Salvo ciertas excepciones, en los centros de formación, la evolución de los programas educativos y de las técnicas de enseñanza no es tan rápida como los avances en el campo de la seguridad de la información lo requiere.

1.2. Aspectos Educativos y Pedagógicos

Las personas aprenden a través de distintos procesos. La estrategia de aprendizaje o de enseñanza debe ser seleccionada teniendo en cuenta tanto los contenidos como el público destinatario. En [4] Bishop y Orvis presentan un enfoque para realzar buenas prácticas e implementar un estilo de programación adecuado a lo largo de una malla curricular. Según Bishop y Orvis, la técnica es también utilizada por escuelas de abogados para enseñar escritura legal. El concepto de buenas prácticas debe ser entendido como un conjunto de sugerencias relacionadas a cómo realizar procesos (*e.g.* ciclo de vida del *software*), normas de cómo escribir código fuente, entre otros [1,2,3].

Los autores del presente trabajo, en clases de la cátedra Redes de Computadoras dictadas en la Universidad Católica *"Nuestra Señora de la Asunción"*, tuvieron la oportunidad de introducir conceptos generales de seguridad y hacer énfasis en aquellos que se relacionan a programación y protocolos. En diversas ocasiones, y aunque existen excepciones, la mayoría de los alumnos captan las ideas a partir de ejemplos concretos pero tienen dificultades al momento de abstraerse y comprender el caso general. Esto se puede deber a la poca o nula cantidad de experiencias prácticas relacionadas a seguridad de la información que se desarrollan en la carrera. Los alumnos encuestados dan tres motivos principales por los que creen que no es mayor el número de experiencias similares en su carrera: falta de tiempo tanto en clase como de los profesores, alta dificultad

de actualización y mantenimiento para los recursos con los que se cuenta, y la complejidad de crear y ejecutar las experiencias.

La inclusión de experiencias prácticas en el aula podría ayudar a los participantes a fijar los conceptos teóricos. Este trabajo está motivado por el hecho de que muchos instructores y profesores se dedican a la enseñanza como una actividad laboral relevante intelectualmente, pero no económicamente. Es decir, el tiempo dedicado a la preparación de los cursos en los que participan en general es reducido y la adaptación, configuración y realización de experiencias prácticas excede al tiempo disponible; todo esto sin considerar otros aspectos como los requerimientos de equipos y configuración.

1.3. Objetivos del Trabajo

El presente trabajo tiene como objetivo brindar una herramienta práctica que por su simplicidad de instalación, mantenimiento y gestión así como por los escasos recursos que requiere, podría disminuir notoriamente el costo en tiempo y en materiales, al realizar actividades prácticas de seguridad en clase.

Conocidos los beneficios reportados al realizar en clase experiencias como laboratorios prácticos, y revisando las dificultades encontradas por los instructores en las distintas técnicas de enseñanza [4,10,14,15,16,17], la herramienta aquí presentada se diferencia de otras en la simplicidad de preparación y administración del entorno, ya que según los datos recabados, la mayoría está de acuerdo en que estas experiencias son beneficiosas para los estudiantes, pero que los instructores no cuentan con recursos suficientes (*e.g.* tiempo, componentes) para llevarlas a cabo.

En lo que sigue del documento, la Sección 2 presenta el entorno de experimentación con sus características y escenarios, finalizando con la Sección 3 donde se presentan las conclusiones y posibles trabajos futuros.

2. Entorno de Experimentación

El lenguaje de programación utilizado en la construcción de la herramienta es PHP debido a su uso extendido y popularizado. Otros lenguajes que pudieron haber sido seleccionados son Ruby o Python pero los autores priorizaron el hecho de que PHP es anterior a dichos lenguajes por lo que es más probable que los instructores, profesores o guías estén familiarizados con él. Como motor de base de datos fue elegido PostgreSQL y con poco esfuerzo podrían realizarse las adaptaciones necesarias para utilizar MySQL, Oracle u otro motor, dada la simplicidad de las funciones y demás objetos de base de datos.

En la Figura 1 pueden observarse las tablas y relaciones que conforman el Diagrama de Entidad Relación. La base de datos tiene dos funciones principales: almacenar en forma de bitácora las acciones relevantes realizadas por los participantes, y almacenar los datos utilizados en los distintos escenarios. Las tablas `base_personas` y `base_usuarios` almacenan los datos originales que son insertados en `datos_personas` y `datos_usuarios` por cada nuevo participante

que se registra en la plataforma de forma que los participantes tengan cada uno un ambiente aislado. Los datos base son también utilizados cuando se acciona un mecanismo de restauración para un participante dado. La tabla `cupones` es utilizada en el octavo escenario, y las demás tablas mantienen información sobre los participantes y las acciones que realizan.

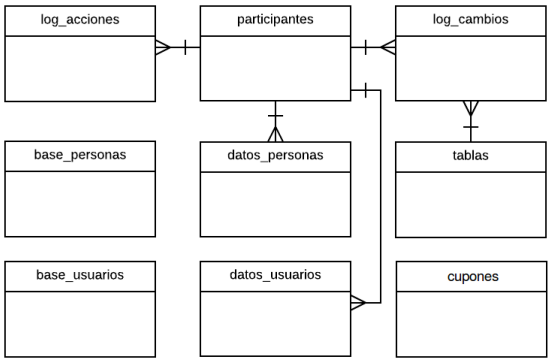


Figura 1. Diagrama entidad relación simplificado.

El entorno desarrollado se conforma principalmente de escenarios que son de nivel básico, y no son suficientemente complejos como para satisfacer las necesidades de una materia que se especialice en seguridad pero permite a los participantes llevar a la práctica las nociones mínimas y adentrarse en el área de seguridad. Entre las listas de vulnerabilidades más críticas y relevantes se destacan la de OWASP (*Open Web Application Security Project*) [11], y la de CWE (*Common Weakness Enumeration*) y SANS (*SysAdmin Audit, Networking and Security Institute*) [9]. Cada escenario plantea un desafío distinto basado en las vulnerabilidades abarcadas por las listas antes mencionadas de manera a abarcar situaciones actuales y de alto impacto. En particular, el quinto escenario plantea una situación real no cubierta por dichas listas pero sí por algunas guías [6,12].

En el diseño del entorno se tuvo en cuenta que el instructor puede agregar nuevos escenarios o modificar a los escenarios existentes. Para ello se necesitan algunos conocimientos de programación. Los escenarios podrán ser modificados, ampliados o extendidos en la medida que las listas de OWASP y CWE/SANS sean actualizadas. Una forma alternativa de utilizar la herramienta propuesta es que los participantes puedan crear o modificar escenarios y que los mismos sean probados por sus pares.

2.1. Escenarios

A continuación se describen los ocho escenarios incorporados actualmente, cuya relación con vulnerabilidades frecuentes se resumen en el Cuadro 1.

Cuadro 1. Resumen de relación entre escenarios y vulnerabilidades

Escenario	CWE/SANS	OWASP
1	CWE–89	A1, A2
2	CWE–79, CWE–134	A3
3	CWE–79, CWE–352, CWE–601	A8, A10
4	CWE–79, CWE–862, CWE–829	A6, A7
5		
6	CWE–89, CWE–250	A1, A3, A5
7	CWE–79, CWE–311	A2, A3
8		A5, A9

Escenario 1. Demuestra una de las consecuencias de un error típico de programación relacionado al limpiado de datos. La vulnerabilidad generada por el error de programación permite que se inyecte código SQL (*Structured Query Language*) y ocupa el primer lugar en ambas listas: CWE/SANS con 93,8 puntos bajo el identificador CWE–89, y OWASP con identificador A1. El escenario también se relaciona con el segundo puesto de la lista OWASP sobre déficit en autenticación y manejo de sesiones. Específicamente, el escenario se centra en explotar la vulnerabilidad en el método de autenticación. El objetivo planteado es ingresar sin conocer ni adivinar usuario y contraseña, como puede verse en la Figura 2. Se espera que el participante, aunque requiera de una pequeña investigación en Internet (e.g. `http://www.redeszone.net/seguridad-informatica/inyeccion-sql-manual-basico` o `http://www.securityartwork.es/2013/11/21/evasion-de-autenticacion-con-inyeccion-sql`), realice una inyección de código que cambie la sentencia lógica evaluada por el motor de base de datos en la aceptación del usuario permitiendo así su acceso al sistema.

Escenario 2. La inyección de código se encuentra en cuarto puesto del CWE/SANS con identificador CWE–79, cuya raíz es la inapropiada neutralización de datos de entrada durante la generación del contenido. La vulnerabilidad denominada CWE–134 trata acerca de la falta de control sobre los formatos de cadena. Según OWASP, el tercer riesgo más crítico es el de XSS (*Cross-Site Scripting*) causado por la falta de limpiado de datos ingresados por el usuario y el correspondiente escape de ciertos caracteres como los símbolos mayor y menor. El participante debe lograr inyectar código JavaScrip para que se despliegue una alerta a los usuarios que carguen la página. Cada persona listada posee un botón que permite ingresar a la edición de los campos correspondientes a sus datos (ver Figura 3). El ejercicio plantea sólo que se despliegue un cartel emergente pero el

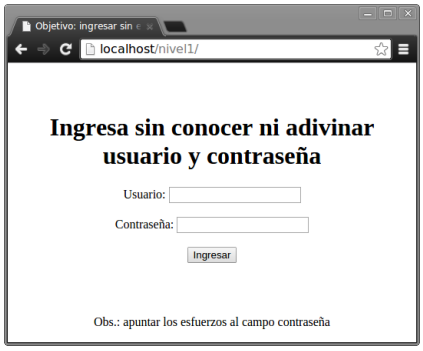


Figura 2. Pantalla del escenario 1.

ataque tiene gran potencial ya que podría descargar datos al cliente y acceder a información existente en el DOM (*Document Object Model*), entre otros métodos que forman parte de los ataques. En concreto, el participante realiza inyección de código a través de uno de los campos y verifica si tuvo o no éxito reingresando al formulario.

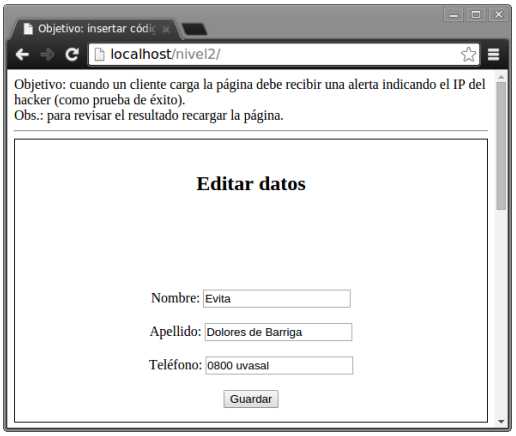


Figura 3. Pantalla del escenario 2.

Escenario 3. Es similar al anterior y provee la misma interfaz al participante, pero difiere en el objetivo ya que el código JavaScript inyectado debe redirigir al usuario en forma automática a otro sitio completamente diferente. La peligrosidad de este ataque aumenta si el sitio vulnerable utiliza marcos (*frames*) ya que en tal caso el usuario podría no notar el hecho de que parte del sitio que observa fue reemplazado por contenido malicioso que podría imitar al original. Este escenario está relacionado al error con identificador CWE–79 que trata sobre la

inapropiada neutralización de datos de entrada, con CWE–352 que se encuentra en el puesto once y trata sobre CSRF (*Cross-Site Request Forgery*), y con CWE–601 que se relaciona con las redirecciones a sitios no confiables. Revisando la lista de OWASP, la vulnerabilidad se relaciona al octavo puesto sobre CSRF y al décimo puesto sobre no validación de redirecciones y reenvíos. Debe destacarse que actualmente algunos entornos como Django y algunos navegadores web implementan cabeceras que controlan el origen del contenido y no permiten que un marco *iframe* sea cargado con contenido de un dominio distinto a los autorizados o al del sitio principal. Esta vez, el participante realiza inyección de código JavaScript y verifica si tuvo o no éxito cargando nuevamente la página que lo debería redirigir al sitio externo inyectado por el atacante.

Escenario 4. Se centra en el método utilizado por un formulario al enviar los valores y en la mala práctica de utilizar campos ocultos para definir el comportamiento de una funcionalidad. El escenario está relacionado con el error CWE–79 ya referido anteriormente, y con los errores CWE–862 (autorización faltante) y CWE–829 (inclusión de funcionalidades desde ámbitos de control no confiables). En lo referente al listado de OWASP, el escenario se relaciona al sexto puesto (exposición de información sensible) y al séptimo puesto (falta de control de nivel de acceso). Como objetivo del escenario se solicita al participante que obtenga la contraseña de un usuario aprovechando una vulnerabilidad de diseño. Este escenario demuestra la peligrosidad de que el usuario tenga acceso a modificar el funcionamiento del sistema por falta de validación, exposición de código y en general, por mal diseño. En este caso, el participante debe observar el código HTML (*HyperText Markup Language*) y cambiarlo de forma que en vez de obtener el correo electrónico recupere la contraseña. Se hace notar aquí la función del instructor que podría necesitar dar pistas a los participantes sobre los posibles campos que podrá tener la tabla en cuestión.

Escenario 5. Enfoca un error humano que puede dar lugar a diversas vulnerabilidades. El objetivo es demostrar la gravedad del error en forma general. Para ello, se muestra al participante un texto que habla sobre la importancia de tener copias de respaldo antes de realizar modificaciones importantes a ciertos archivos. El error humano surge debido a que algunas personas dejan esas copias de respaldo en ubicaciones alcanzables por un usuario, y luego se olvidan de borrar los archivos temporales. Para realizar una copia de respaldo rápida, una forma muy utilizada es copiar el archivo y agregar un sufijo que indique que el mismo es una copia vieja o de respaldo (*e.g.* utilizando extensiones como .old, .bkp). Un servidor web, a menos que sea configurado en forma distinta, descarga al cliente los archivos cuyas extensiones no sean bloqueadas o indiquen que el archivo debe ser interpretado (*e.g.* .php, .aspx). Otra forma de exposición de código se da cuando el programador deja comentarios que incluyen datos sensibles como contraseñas, nombres de usuario u otra información privada. El error de procedimiento no está incluido en las listas referidas debido a que estas listas se centran en las aplicaciones y no en la configuración de servidores, pero figura en guías como [6,12]. En resumen, el participante podría modificar la dirección

de un recurso como `conexion.php` intentando acceder a una copia de respaldo, por ejemplo `conexion.php.bkp` o `conexion.old`.

Escenario 6. Se enfoca en el alcance que podría tener la inyección de código SQL solicitando al participante que borre de la base de datos todos los datos de una tabla. El escenario está asociado en forma directa a la vulnerabilidad de mayor riesgo según CWE/SANS: CWE–89 (relacionada a la falta de neutralización de código SQL), y se ve potenciada por CWE–250 (relacionada a la ejecución con permisos innecesarios). También se relaciona al primer puesto de la lista OWASP debido a la posibilidad de inyectar código SQL que resulte en el borrado de datos, al tercer puesto debido a la falta de limpieza de datos de entrada que posibilita un ataque de tipo XSS y al quinto puesto relacionado a una mala configuración de seguridad ya que la operación de borrado no se ve obstruida por permisos. Básicamente, el participante debe aprovechar un campo del formulario para inyectar código SQL con instrucción de borrado de datos, por ejemplo, utilizando la sentencia `delete` y verificando el éxito de su ataque al no encontrar datos en la lista de usuarios existentes que muestra la página.

Escenario 7. Tiene como meta el robo de información de las *cookies* a través de inyección de código JavaScrip. El objetivo del escenario es demostrar cómo un atacante podría robar y alterar la información que contienen dichos archivos. Este ataque podría causar graves daños como alteración de información o permitir que el atacante gane acceso a una aplicación a través de la cuenta de la víctima aún sin conocer nombre de usuario ni contraseña. Este ataque posee características de XSS encontrándose en el cuarto puesto de la lista de CWE/SANS bajo el identificador CWE–79 y en el tercer puesto de la lista OWASP, y al segundo puesto de ésta última relacionado al déficit en autenticación y manejo de sesiones. El esfuerzo del atacante para modificar los datos es menor si las *cookies* son almacenadas en texto plano en vez de encriptado, considerando que el octavo puesto de la lista CWE/SANS lo ocupa la vulnerabilidad CWE–311 relacionada a la falta de encriptación de datos sensibles. El participante debe inyectar código JavaScript en un formulario para acceder a la *cookie* ya creada por la herramienta donde se almacena la información de sesión.

Escenario 8. El octavo escenario se centra en un área del diseño de protocolos seguros. Demuestra en forma práctica lo que podría ocurrir cuando se requiere que un dato sea aleatorio, pero debido a la implementación se asignan variables con una aleatoriedad pobre, razón por la cual no es aconsejable que personas no expertas utilicen generadores pseudo-aleatorios de diseño propio [5,13]. A modo de ejemplo, HTTP es un protocolo sin estado, por lo que en cada transacción el navegador envía al servidor el valor de una variable de sesión para recuperar la información. Si el atacante adivina dicho valor puede ganar acceso a la sesión sin necesidad de interceptar paquetes que intercambian entre cliente y servidor, ni de realizar un ataque directo a la víctima. En el octavo escenario se presenta una pantalla desde donde solicitar cupones para analizar la aleatoriedad de los códigos. El objetivo es que el participante deduzca la lógica detrás de la generación de códigos y encuentre el cupón por valor de un millón de Dólares. En ocasiones se hace uso de componentes de terceros con vulnerabilidades conocidas o que

requieren cierta configuración para funcionar en forma segura. Estas dos últimas posibilidades ocupan el quinto y el noveno puesto en la lista de OWASP bajo los títulos de "Mala configuración de seguridad" y "Utilización de componentes con vulnerabilidades conocidas", respectivamente. En concreto, el participante puede observar la relación entre cupón y premio para luego intentar deducir el código del cupón especificado en el ejercicio, para lo cual hace uso de una entrada del formulario donde puede verificar el premio que corresponde a cada código que el participante desea consultar. El participante puede probar tantos códigos de cupones como desee, pero claramente le será muy difícil lograr el objetivo por búsqueda exhaustiva.

3. Conclusiones

Los cursos de seguridad de la información deben cubrir una vasta cantidad de áreas y temas; el tiempo disponible impide ahondar suficientemente en todas las áreas, por lo que muchos autores intentan confeccionar guías generales que puedan ser aplicadas en forma transversal, y que sean mayormente válidas para cada caso [17]. El tiempo y los recursos son los principales motivos por los que muchas instituciones educativas no aumentan la cantidad de prácticas, pero existen formas de incluirlas a lo largo de la malla curricular [4,16]. Es crucial seleccionar buenas técnicas pedagógicas para que los estudiantes puedan adquirir los conocimientos necesarios en un espacio de tiempo razonable. Así, este trabajo está motivado por el hecho de que, en general, el tiempo del que disponen muchos instructores y profesores para preparar los cursos es menor al que requiere la adaptación, configuración y realización de experiencias prácticas. El presente trabajo tiene como objetivo brindar una herramienta práctica que por su simplicidad de instalación, mantenimiento y gestión así como por los escasos recursos que requiere, podría disminuir notoriamente el costo en tiempo y en materiales de realizar actividades prácticas en clase. Dichas actividades reportan valiosos beneficios sorteando dificultades encontradas por los instructores en las distintas técnicas de enseñanza [4,10,14,15,16,17].

La herramienta presentada se diferencia de otras en la simplicidad de preparación y administración del entorno, el cual se conforma principalmente de escenarios y de componentes destinados a la administración y recuperación de datos. Los escenarios permiten a los participantes llevar a la práctica nociones básicas y adentrarse en el área de seguridad. Cada escenario plantea un desafío distinto basado en las vulnerabilidades frecuentes mencionadas en las listas de OWASP y CWE/SANS de manera a abarcar situaciones actuales y de alto impacto, aunque uno de los escenarios propuestos plantea también situación real no cubierta por dichas listas, pero sí por algunas guías [6,12].

En el diseño del entorno se consideró que el instructor puede agregar nuevos escenarios o modificar los existentes. Los escenarios deberán ser actualizados, ampliados o extendidos en la medida que las listas de OWASP y CWE/SANS sean actualizadas. Una forma alternativa de utilizar la herramienta propuesta es que el objetivo de los participantes sea crear o modificar escenarios y que los mis-

mos sean probados por sus pares. Los autores –en su calidad de profesores de la cátedra Redes de Computadoras– realizaron experiencias que resultaron ser muy alentadoras tanto en las carreras de Ingeniería Electrónica como de Ingeniería Informática de la Universidad Católica *Nuestra Señora de la Asunción*.

Como trabajos futuros los autores siguen trabajando en: (a) la inclusión de más escenarios aumentando la dificultad o abarcando vulnerabilidades tidavía no incluidas en los existentes, por ejemplo, la carga de archivos al servidor; (b) adaptar la herramienta para simplificar la creación de escenarios destinados a ataques de denegación de servicio; y (c) agregar un componente que permita al participante, una vez que descubra la contraseña, acceder a una consola donde pueda ejecutar código SQL en la base de datos, entre otras alternativas.

Referencias

1. Abran, A., Bourque, P.: SWEBOK: Guide to the software engineering Body of Knowledge. IEEE Computer Society. 2014
2. Apple Inc: Secure Coding Guide. 2014
3. Belk, Coles, Goldschmidt, Howard, Randolph, Saario, Sondhi, Tarandach, Vähä–Sipilä, Yonchev: Fundamental Practices for Secure Software Development. 2011
4. Bishop, M., Orvis, B. J.: A clinic to teach good programming practices, Proceedings of the 10th Colloquium for Information Systems Security Education. 168–1174 (2006)
5. Ferguson, N., Schneier, B., Kohno, T.: Cryptography Engineering: Design Principles and Practical Applications. ISBN 9780470474242. Wiley Publishing. 2010
6. Heckathorn, M.: Network Monitoring for Web-Based Threats. 2011
7. Kim, P.: The Hacker Playbook: Practical Guide to Penetration Testing. ISBN 9781494932633. Createspace Independent Pub. 2014
8. Kölling, M.: The problem of teaching object-oriented programming, Part 1: Languages. SIGS Publications. Journal of Object-oriented programming. Vol. 11, Num. 8, 8–15 (1999)
9. Martin, B., Brown, M., Paller, A., Kirby, D., Christey, S.: 2011 CWE/SANS top 25 most dangerous software errors. Common Weakness Enumeration. 2011
10. NESG – Network Engineering & Security Group: NETA: A NETwork Attacks Framework. Architecture and Usage. University of Granada. 2013
11. OWASP Foundation: OWASP Top 10. `https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project`. 2013
12. OWASP Foundation: OWASP Testing Guide v4. `https://www.owasp.org/index.php/OWASP_Testing_Project`. 2014
13. Paar, C., Pelzl, J.: Understanding cryptography : a textbook for students and practitioners. ISBN 978-3-642-04101-3. 2010
14. Schindler, F.: Coping with Security in Programming. Acta Polytechnica Hungarica. Vol. 3, Num. 2, 65–72 (2006)
15. Talib, M. A., Khelifi, A., Jololian, L.: Secure Software Engineering: A New Teaching Perspective Based on the SWEBOK. Interdisciplinary Journal of Information, Knowledge, and Management. Vol. 5, 83–99 (2010)
16. West, J., Bishop, M.: Security Education for the new Generation, Proc. RSA Conference. 2014
17. Yurcik, W., Doss, D.: Different approaches in the teaching of information systems security, Proceedings of the Information Systems Education Conference. 2001

Formación de docentes y alumnos en el desarrollo de actividades de programación colaborativa en línea

Alejandro Héctor González[1], Beatriz Depetris[2], Cristina Madoz[1], Daniel Aguil Mallea[2]

[1]Instituto de Investigación en Informática III- LIDI Facultad de Informática- Universidad Nacional de la Plata - UNLP
[2] Universidad Nacional de Tierra del Fuego - UNTDF
{agonzalez,cmadoz}@lidi.info.unlp.edu.ar
{depetrisb,daguilmallea}@untdf.edu.com

Abstract. Este artículo presenta el trabajo desarrollado entre un grupo de alumnos y docentes de primer año de las Carreras de Ingeniería en Computación de la Universidad Nacional de la Plata y de la carrera de Licenciatura en Sistemas de la Universidad Nacional de Tierra del Fuego.
Se desarrolla la propuesta de actividades de programación colaborativa en línea, donde en forma conjunta se propone la formación de docentes y estudiantes en las estrategias de trabajo colaborativo orientadas al desarrollo de algoritmos y programas. Se trabaja en un marco de educación en línea en forma colaborativa con grupos conformados por participantes de ambas universidades. Cada grupo tiene asignado un tutor encargado de la supervisión de las tareas. Se muestran los resultados obtenidos en la implementación de la primera experiencia a través de encuestas y entrevistas en el desarrollo del curso.

Keywords: educación en línea, trabajo colaborativo, programación, algoritmos

1 Introducción

Este artículo presenta las estrategias de formación de profesores y estudiantes aplicadas en una experiencia de actividades de programación colaborativa en línea.

La generación de habilidades y destrezas de trabajo en equipo, con integrantes que se encuentran físicamente distantes, es una tarea que cada vez está más presente en el desarrollo profesional de los profesores y de los estudiantes. Generar nuevas competencias en el ámbito universitario, incorporando herramientas de apoyo para mejorar los procesos de enseñanza y de aprendizaje, permitirá un mejor desempeño en la sociedad del conocimiento en que estamos insertos.

La propuesta de actividades que se presenta, tuvo por objetivo generar un espacio educativo donde los alumnos pudieron desarrollar la solución a un problema de manera colaborativa, utilizando herramientas digitales en un ambiente en línea. Participaron de la misma dos asignaturas de dos Universidades Nacionales de

Argentina: la Universidad Nacional de La Plata (UNLP), a través del Instituto de Investigación y Desarrollo en Informática III-LIDI, y la Universidad Nacional de Tierra del Fuego (UNTF), a través del Instituto de Desarrollo Económico y de Innovación.

En ediciones de años anteriores, en ambas asignaturas se advirtió una escasa participación grupal en el aula, un tema, por otra parte, que es recurrente en varias materias de las carreras. Este inconveniente no sólo es manifestado por los estudiantes, sino también por los profesores. Esta falta de interacción entre alumnos y entre alumnos y docentes necesita ser revisada, pues es sabido que los ahora estudiantes se enfrentarán, en el ámbito profesional, con situaciones de desarrollo de software en equipo con diferentes tecnologías. Los estudiantes deben tener la posibilidad de experimentar un trabajo real donde deben contactarse con sus compañeros de equipo (otros alumnos y profesores) que físicamente están en otro contexto y no se conocen cara a cara.

2 Marco teórico

El aprendizaje colaborativo se basa en la resolución de problemas reales, que deben enfrentarse en forma grupal e individual donde interviene el aprendizaje colaborativo y auto-regulado o independiente [1].

Para Chan y Aretio el trabajo colaborativo presenta las siguientes características:

Colaboración: el estudiante aprende a colaborar con los demás en pos de alcanzar un objetivo común. Esto incluye el diálogo entre pares y la solución consensuada de conflictos.

Solidaridad: es un valor que consiste en mostrarse unido a otras personas o grupos, compartiendo sus intereses y necesidades.

Respeto: significa aprender a convivir con la diversidad, respetando las opiniones, gustos, costumbres e interpretación de los problemas.

Participación social: en cuanto a tener conciencia de pertenencia a un grupo, comprometiendo su participación libre y activa en trabajar para el bien común.

Responsabilidad: en cuanto a los compromisos y obligaciones para con los miembros del grupo.

Trabajo en equipo: Implica compromiso y la necesidad de que exista liderazgo, armonía, responsabilidad, creatividad, voluntad, organización y colaboración entre todos y cada uno de los miembros del grupo.

Roberts señala como beneficios psicológicos del aprendizaje colaborativo la capacidad para aumentar la autoestima de los estudiantes y la de desarrollar en ellos actitudes positivas hacia los docentes [2].

¿Y qué ocurre con la mediación de la tecnología digital y el trabajo colaborativo?

En la actualidad se cuenta con diferentes herramientas que pueden facilitar el trabajo colaborativo en línea, como documentos compartidos, pizarras electrónicas, webconferences, posters comunitarios, redacción de libros en forma grupal, blogs, wikis, y otras aplicaciones [3].

Al trabajar en un entorno colaborativo virtual, el tutor tiene la posibilidad de supervisar el desempeño de los estudiantes en la resolución de los problemas y de realizar las intervenciones y orientaciones que sean pertinentes en pos del adecuado desarrollo de las soluciones que se plantean. [4]

En la actualidad varias actividades docentes y de estudiantes se desarrollan en línea, pero ¿qué es hacer educación en línea?

Tomando como referentes a García Aretio, y en particular a Schwartzman, Tarasow y Trech, se debe reflexionar sobre si las cosas pueden (ahora) ser en línea o presencial, si podemos enseñar esto o lo otro, si nuestra enseñanza puede ser aceptada (o no), tenemos entonces muchas más cosas que procesar y decidir que antes. [5]

Este proceso interpela y pone en diálogo la estabilidad del contexto del aula (de la vida en las aulas, o de los contenidos) como una opción que no podemos garantizar y el entorno cambiante y el desafío de enseñar en el siglo XXI. [5]

El primer planteo se da en el ámbito de la "distancia" y el ¿por qué la distancia representa un problema? Entendiendo la distancia como distancia geográfica. Hay que pensar esta pregunta en el contexto donde la enseñanza universitaria habitualmente está centrada en la transmisión de información y el proceso de aprendizaje en recepción/asimilación de la información, como modelo transmisivo de conocimiento. La distancia es un problema porque aquello que el docente explica, enseña, ejemplifica no puede ser captado simultáneamente por los alumnos, como sucede en una situación presencial. Entonces, la tecnología es utilizada como un puente que permite que los contenidos fluyan con facilidad, superando la barrera de la distancia, y lleguen a sus receptores; pero esto pareciera que no alcanza. La situación presencial durante muchos siglos sostuvo el modelo transmisivo en sus más variadas formas, pero ya no resulta en todos los casos.

La palabra "distancia" debe ser revisada e integrada en un término que la pueda incluir. La propuesta educativa debe pensarse y articularse de forma que no se encuentre centrada en un modelo meramente de transmisión y "físicamente distante", debe fundamentarse en el diálogo y la construcción del conocimiento donde la utilización de la tecnología digital tendría otras finalidades. Se puede pensar en propuestas donde se incorpore lo sincrónico y asincrónico y pensar en la "educación en línea" como una denominación diferente de este proceso.

¿Cómo lograr este cambio en instituciones con años de enseñanza transmisiva, y cómo formamos a los actores del proceso educativo?

El entorno educativo provisto de tecnología digital debe integrar y comunicar a los profesores, estudiantes y recursos para generar interacciones y desarrollos colaborativos. Diseñar educación en línea es un nuevo desafío para todos en la institución. Se debe dar tiempo a que se desarrolle el proceso de construcción que supone aprender reconociendo que, al tratarse de una práctica social y humana, no todo podrá anticiparse, aunque se conozca el grupo de estudiantes, el contenido a enseñar y la estrategia a desarrollar. [5]

Para Schwartzman un momento importante consiste en la visualización arquitectónica de nuestro territorio digital. Ni colaborativo, ni expositivo, ni conductista: la riqueza del diseño en entornos en línea está dada por la plasticidad de

los mismos. Materiales, roles docentes, eje en actividades. Recorrido, acceso y las intervenciones docentes. ¿Cómo lograrlo? [5]

2.1 Experiencias

Se describen a continuación algunas experiencias universitarias previas que se consideran de interés:

Laboratorios Virtuales en la UNRN.

En el trabajo de utilización de laboratorios virtuales de la Universidad Nacional de Rio Negro de Argentina (UNRN) se presenta una experiencia en el marco de una propuesta de enseñanza y aprendizaje colaborativa. Se utilizan como ambiente de desarrollo, para las prácticas de laboratorio de un curso introductorio de programación, el entorno Moodle y el laboratorio virtual de programación "VPL". [6]

Facultad de Minas de la Universidad Nacional de Colombia.

Se puede mencionar el caso del ambiente instruccional "Sábato", desarrollado por la Facultad de Minas de la Universidad Nacional de Colombia, que fusiona las estrategias educativas del Aprendizaje Basado en Problemas (PBL) y el Aprendizaje Colaborativo Apoyado en Computador (CSCL). Se aplica en el área de desarrollo de proyectos de informática entre otros casos [7].

Ingeniería en Computación en la Universidad Nacional de La Plata.

También se pueden mencionar las diferentes experiencias que se desarrollan desde el año 2013 en la cátedra de Programación 2 de la Carrera de Ingeniería en Computación de la Universidad Nacional de la Plata, donde los estudiantes deben resolver una actividad en línea en forma colaborativa y compartir las soluciones alcanzadas con sus compañeros para la corrección cruzada y lograr posteriormente el acuerdo de cuales soluciones son las que más se aproximan al desarrollo esperado [8].

En todos los casos mencionados intervienen los docentes como tutores del proceso de enseñanza y aprendizaje y se requiere que los mismos tengan las habilidades necesarias para poder favorecer el trabajo colaborativo. Esto último es una tarea que en general debe ser aprendida por los profesores y que requiere no solo de los supuestos teóricos sino también de la práctica real para comprender el complejo proceso.

3 Propuesta integral de formación

3.1 Organización de los tutores

Uno de los propósitos para la formación de profesores en relación a la incorporación de las tecnologías digitales es promover la equidad, la inclusión y el desarrollo personal y académico de los estudiantes. A partir de esta formación, los profesores comienzan a tomar decisiones en relación a las estrategias de enseñanza

que utilizarán en relación con las Tecnologías de la Información y Comunicación (TIC) [9] [10].

El rol del docente en el desarrollo de las actividades en línea es fundamental para garantizar la calidad de la enseñanza. Ryan describe cuatro roles básicos a desempeñar por el profesorado: pedagógico, social, de dirección y técnico [11].

El *rol pedagógico* le permite al profesor contribuir a la creación de conocimiento especializado, a identificar junto con los estudiantes los puntos críticos, a responder preguntas, etc.

A través del *rol social* potencia la creación de un clima propicio para la colaboración entre estudiantes. Esta función social se corresponde con la figura del tutor. La figura del tutor es la que interviene sistemáticamente en el proceso de aprendizaje de los alumnos y en el marco de los desarrollos del entorno virtual. [12] [10]

El *rol de dirección* permite brindar una propuesta pedagógica que interpele a los estudiantes para que sean partícipes de la propuesta de enseñanza, para que se comprometan con la tarea que van a desarrollar, generando más autonomía en su trabajo. Los estudiantes deben saber lo que están haciendo, en el momento en que lo están haciendo y saber para qué lo realizan.

El *rol técnico* atiende a las cuestiones puntuales referidas a la tecnología digital, sus usos y recomendaciones.

3.2 Contexto de la experiencia

El desarrollo de la propuesta se realizó en diferentes etapas que permitieron organizar la actividad de los estudiantes y profesores. Se propuso armar una experiencia piloto con pocos estudiantes para poder observar el desempeño de los profesores y la interacción en general entre todos los actores. La experiencia se desarrolló en el segundo semestre de 2014, entre las cátedras de Programación 2 perteneciente a la carrera de Ingeniería en Computación de la UNLP, y la de Algorítmica y Programación II de la carrera de Licenciatura en Sistemas de la UNDTF. Ambas asignaturas son cuatrimestrales.

Para la experiencia, fueron seleccionados 6 alummos de cada una de las materias mencionadas (de un total de 40 que cursaban la cátedra Programación II y de los 8 que lo hacián en Algorítimica y Programacion II.

El enunciado del proyecto se formuló en la aproximación del concepto de aprendizaje situado (activo, constructivo y real) [13] . La situación estaba inscripta en un problema particular (reserva de pasajes de trenes). De esta manera, se facilitó una participación activa de los alumnos, animando la búsqueda de datos reales y consistentes con la situación y ofreciendo, además, la posibilidad de ensayar distintos conjuntos de datos.

Se describen a continuación las 6 etapas propuestas en la experiencia. [13].

1. Etapa de ajustes y puesta en común del tema.

En esta etapa se acordaron los contenidos y el tema a tratar. Se compartieron los planes de estudios y los contenidos de las diferentes asignaturas candidatas.

Esta etapa se desarrolló entre Noviembre de 2013 y Febrero de 2014. Se analizaron y adaptaron los cronogramas de trabajo durante el semestre para garantizar que los contenidos teóricos y prácticos estuvieran desarrollados. Se encontraron coincidencias en varios temas y se decidió trabajar sobre una actividad integradora final. El tema elegido fue "tipos abstractos de datos" (TAD). Se trabajó con el lenguaje Pascal y se abordó el análisis, diseño y codificación en Pascal para la solución de un problema.

2. Armado del sistema tutorial y actividades en línea.

Se eligieron las herramientas que se necesitaban para lograr un adecuado desarrollo de la actividad. También se acordaron la fecha de inicio y fin de la actividad. Para la experiencia piloto se definió un sistema tutorial donde se contaba con un tutor de cada universidad y dos tutores coordinadores que revisaran las actividades en forma general en cada institución. Se generaron 4 grupos de trabajo de estudiantes.

Las herramientas utilizadas fueron los documentos compartidos en Google Docs, y el Proyect Lazurs, que permite compilar en Pascal y codificar en un ambiente de programación. El entorno virtual de enseñanza y aprendizaje para lograr la comunicación fue Webunlp[1].

Para las actividades en línea se definió un documento que indicaba las pautas de trabajo para los alumnos y tutores.

3. Pautas de evaluación de la actividad

En esta etapa se definió como realizar la evaluación de la actividad en forma completa y, si esta evaluación se incorporaba o no a la nota de acreditación de la asignatura. Se estableció el contenido de las preguntas del cuestionario inicial que sirvió para evaluar el nivel de conocimientos de los participantes con referencia a las herramientas informáticas que utilizaban habitualmente. Se realizó también un documento con las pautas de corrección para los tutores. La corrección se devolvió a los estudiantes en un documento donde se detallaban los comentarios y correcciones sobre:

- *Proceso de diseño de la solución*: se revisaron la cantidad de intervenciones de los estudiantes en el documento compartido, cantidad de consultas en clase, calidad de los aportes.
- *Programa ejecutable entregado*: se establecieron los siguientes casos de prueba: sin elementos, con 1 elemento, con varios elementos. También se observó la calidad general de los mensajes por pantalla y otros comentarios generales a la ejecución del programa. En el caso de encontrar inconvenientes se recomendó colocar una imagen de la pantalla y justificar.
- *Con referencia al código* se formularon algunas preguntas generales referidas a los tipos de datos y funcionalidades del código desarrollado. Se solicitó corregir en color ROJO y justificar en color VERDE.

[1] Este entorno fue desarrollado por el Instituto de Investigación en Informática III-LIDI de la Facultad de Informática de la UNLP y está accesible desde: https://webunlp.ead.unlp.edu.ar/

En la UNLP la actividad completa formaba parte de la nota final de promoción junto a otra actividad grupal en línea y los exámenes de teoría. En caso de no aprobar se podía volver a entregar la solución. En el caso de la UNTDF, esta actividad, denominada, "trabajo práctico integrador 2", formaba parte de uno de los requisitos que el alumno debía aprobar para obtener la cursada, conjuntamente con la aprobación de 2 parciales y el trabajo integrador 1.

4. Armado de grupos y comunicación

Para esta etapa de la propuesta se organizaron 3 grupos de 4 alumnos cada grupo: dos alumnos de la UNLP y dos de la UNTDF. Este balance parece adecuado para la contención del grupo, con dos tutores por grupo. La participación de los estudiantes de la UNLP fue voluntaria. Aquellos alumnos que no participaban de esta experiencia con la UNTDF realizaron la misma actividad pero entre compañeros del mismo curso. Los alumnos pudieron elegir su compañero de grupo dentro de su universidad. Los tutores armaron los grupos finales.

Los estudiantes podían consultar a cualquiera de los dos tutores. Para la comunicación se disponía de la mensajería del entorno virtual WebUNLP y se indicó a los estudiantes que podían usar cualquier otra herramienta de comunicación sincrónica o asincrónica que acordaran con el grupo.

5. Puesta en funcionamiento y seguimiento

En esta etapa se definieron las estrategias para poder poner en funcionamiento en forma coordinada el desarrollo de la actividad en línea. Se trabajó el tema con los estudiantes en el aula presencial. Se presentaron la actividad y las pautas de trabajo colaborativo que debían cumplir en línea.

Los tutores siguieron las consultas vía la mensajería del entorno, las preguntas en el aula presencial y supervisaron las decisiones de diseño que fueron realizando los alumnos en el documento compartido.

6. Evaluación

En esta etapa se realizó la evaluación acordada con referencia al desempeño grupal, individual y el programa realizado. Además se sumó a la evaluación de la experiencia, una encuesta en línea de carácter anónimo realizada a los alumnos, y los aportes realizados por los docentes vía e-mail, donde detallaron los logros y las dificultades encontradas a través de sus experiencias.

3.3 Resultados

Participaron de la experiencia 12 alumnos en total, 6 de la UNLP, 6 de la UNTDF y 4 tutores. Los resultados obtenidos del primer cuestionario sobre el acceso y nivel de conocimientos general de los alumnos fue respondido por el 100% y se obtuvo que:

- El 100% indicó tener computadora en la casa distribuido como sigue: PC de escritorio (50%), notebook (60%), netbook (20%), tablet (5%).
- El 30% tiene dos o más computadoras en la casa.

- Respecto al uso de las redes sociales utilizaban Facebook (95%), Google+ (45%), Twitter (30%), y otras redes (5%) como Taringa. El 50% indicó conocer las facilidades que proveía Google Docs para realizar actividades.

Los tres grupos entregaron y aprobaron las actividades

Finalizada la actividad, se realizó una encuesta que versaba sobre las acciones que los alumnos habían realizado para resolverla. Dicha encuesta fue respondida solo por 4 integrantes de los 12. Cabe aclarar que la actividad les demandó a los alumnos más tiempo que el previsto por los docentes. En el caso de los estudiantes de la UNLP se terminaron las teorías dos semanas antes de la entrega. Si bien se establecieron horarios de consulta los alumnos no asistieron a clase y siguieron trabajando en línea.

La solicitud de completar el cuestionario fue posterior a la entrega de resultados y se intuye que esto hizo que no respondieran todos la encuesta. Ante esta situación se decidió realizar una entrevista personal con los involucrados para poder revisar las actividades.

Con referencia a la interacción entre tutores y estudiantes fueron consultados en el aula presencial, por mensajería y a través de los comentarios del google docs. Usaron además, el correo privado y webconference (skype y hangouts) para comunicarse.

Sólo uno de los grupos tuvo problemas para realizar el trabajo; en él uno de los integrantes manifestó que tuvo que realizar todo solo y que no tuvo demasiada colaboración de los alumnos de la otra universidad. También se pudo observar, en el mismo grupo, que al no realizar el documento en forma compartida fue requerida la intervención del docente tutor para su realización. Aquí los dos estudiantes de la UNLP habían desaprobado el examen parcial con lo cual ya no contaba para ellos realizar la entrega del trabajo grupal para tener la nota de promoción. Esta situación no fue comunicada entre los integrantes del grupo. Lo que seguramente generó el enojo de los restantes integrantes, los que igualmente finalizaron la actividad con excelentes resultados, pero con bastante más dedicación de tiempo y esfuerzo.

Desde los aspectos de la organización de los grupos, se señala que uno se auto coordinó, mientras que en los otros dos se percibió la falta de un líder y la necesidad de su existencia para que alguien los dirigiera.

La evaluación final de las actividades fue realizada por un tutor de cada universidad, realizándose la correspondiente devolución a los estudiantes.

4 Conclusiones

Como aspectos positivos se puede observar que los tres grupos modularizaron bien el problema y definieron adecuadamente las estructuras de datos.

Se destaca que la experiencia permitió que los alumnos de Ushuaia, que viven en una región muy alejada de los grandes centros urbanos, conocieran nuevos compañeros y se relacionaran con pares de otra universidad. Estos contactos, no sólo válidos desde los aspectos sociales, les permitieron constatar que muchos de sus saberes y también de sus dificultades eran compartidas por sus pares, a pesar de estar en diferentes casas de estudio.

Algunas dimensiones a revisar tienen que ver con las expectativas de los alumnos en el desarrollo de programas. Se puede observar que los alumnos están acostumbrados a repetir un funcionamiento y están preparados para relacionarse de una determinada manera.

El tema del tiempo fue un factor clave. Como el ejercicio resultó ser largo y se sumaron las fechas de exámenes parciales, que no coinciden entre ambas instituciones, se dificultó y alargó la tarea, en beneficio de algunos y en perjuicio de otros.

Es importante que los estudiantes puedan organizar espacios de comunicación con la supervisión del tutor. Por ejemplo se puede requerir que planifiquen cuantas conferencias web van a realizar, de cuánto tiempo y el objetivo de cada una.

Los docentes hicieron un balance positivo de la experiencia, si bien manifestaron la necesidad de formarse más en técnicas de manejo de grupo y trabajo colaborativo para poder acompañar y aconsejar a sus estudiantes; esta pequeña experiencia de prueba, con un número reducido de grupos, resultó una primera y beneficiosa aproximación hacia este tipo de aprendizaje.

Un aspecto fundamental, para sostener las buenas prácticas pedagógicas y didácticas en la integración de las TICs en la enseñanza, es el accionar del docente como un proceso sostenido en el tiempo. En este tipo de formación se abre un proceso en el cual el docente puede desarrollar diferentes capacidades con el propósito de reconocer nuevas posibilidades, adquirir la confianza y la autonomía para poner en práctica nuevas propuestas, habilidades, e incluso, experimentar nuevos modos de posicionarse frente a la enseñanza y al aprendizaje.

5 Trabajo futuro

Actualmente se trabaja en el análisis de las estrategias de comunicación que utilizan los alumnos y docentes a través del uso de tecnologías digitales. En particular las que utilizan para desarrollar trabajo colaborativo en línea en los primeros años de las carreras de informática. En este aspecto, se prevén repetir experiencias similares, en la disciplina informática, incluyendo otras instituciones de la zona patagónica de nuestro país.

Se indaga para elegir otros temas a trabajar para el desarrollo colaborativo y ampliar las instituciones y las disciplinas a los que pueda serle útil participar de la experiencia en línea.

Este camino invita a crear nuevas experiencias de trabajo colaborativo y a preguntar: ¿Cómo valorar pedagógica y didácticamente la incorporación de las tecnologías digitales en la enseñanza de la programación de computadoras?

¿Seremos capaces de incorporar tecnologías que impliquen trabajo en redes y en colaboración en el proceso educativo?

¿Qué saberes y capacidades es necesario que desarrollen los profesores ante este desafío? ¿Cuáles son los cambios en la manera de enseñar?

6 Bibliografía

1.Chan, C. Assessment: Problem Based Learning Assessment. Assessment Resource Centre, University of Hong Kong.(2008)

2. Roberts, T. S. Computer-Supported Collaborative Learning in Higher Education. London: Idea Group Publishing. (2005)

3. Diaz Barriga F: "La innovación en la enseñanza soportada en TIC. Una mirada al futuro desde las condiciones actuales". VII Foro Latinoamericano de Educación. Documento Básico / Fundación Santillana. (2011)

4. Aretio, L. G., Cirbella, M. R., & Figaredo, D. D. De la educación a distancia a la educación virtual. (2007)

5. Schwartzman G., Tarasow F., Trech M. De la educación a distancia a la educación en línea: aportes a un campo en construcción. 1er ed. Rosario- Homo Sapiens Ediciones. Flacso Argentina. ISBN 978-950-808-867-3 (2014)

6. Lovos E, Gonzalez A. Moodle y VPL como soporte a las actividades de laboratorio de un curso introductorio de programación. Congreso TE&ET 2014, (2014)

7. Jiménez, J., Pavony, M., Álvares, A.: Entorno de integración de PBL y CSCL para la enseñanza de algoritmos y programación en ingeniería. Revista Avances en Sistemas e Informática de la Universidad Nacional de Colombia. Vol. 5 – No. 3., 189-194 (2008)

8. González A., Madoz C. Desarrollo de actividades colaborativas en un curso inicial de programación de computadoras. III Workshop de Innovación en Educación en Informática (WIEI). XX Congreso Argentino de Ciencias de la Computación (Buenos Aires, 2014). San Justo .Argentina. (2014)

9. Sancho Gil, Juana M. "La formación del profesorado en la era de la información: entre lo conveniente, lo deseable y lo posible". En: II Congreso Iberoamericano de EducaRed. "Educación y Nuevas Tecnologías". Buenos Aires, 30 de junio, 1 y 2 de Julio (2005)

10.Martin M., Gonzalez A., Barletta C., Sadaba A. Aulas virtuales, convergencia tecnológica y formación de profesores. Red de Universidades con Carreras en Informática (RedUNCI) . Evento TE & ET (2012).

11. RYAN, S. y otros The virtual university, The Internet and resource-based learning, London, Kogan Page. (2000):

12. Zapata, M. "Estrategias de evaluación de competencias en entornos virtuales de aprendizaje", RED, Revista de Educación a Distancia, Número 1 [en línea]. Disponible en: http://www.um.es/ead/reddusc/1/eval_compet.pdf (2010)

13.González A., Madoz C., Depetris B., Mallea D. Una propuesta de trabajo colaborativo en línea para el desarrollo de algoritmos y programas. En memorias del Congreso de TE&ET Universidad Nacional del Nordeste, Corrientes, Argentina. (2015)

Caracterización de alumnos de una materia mediante datos multivariados obtenidos de un cuestionario de satisfacción

Myriam Herrera[1], María Inés Lund[2], Laura Aballay[2]

[1] Departamento de Informática, FCEFyN – Universidad Nacional de San Juan
[2] Instituto de Informática, FCEFyN – Universidad Nacional de San Juan
{mherrera, mlund, laballay}@iinfo.unsj.edu.ar

Abstract. Se presenta en este trabajo una caracterización de los alumnos de la materia Diseño de Software, aplicando un proceso de análisis de datos multivariados, mediante la técnica de Análisis de Correspondencias Múltiples, obteniendo una clasificación de los alumnos, de acuerdo a los datos obtenidos del cuestionario de satisfacción respondido por ellos. Esta nueva técnica genera información que permite obtener datos para mejorar la calidad del dictado del curso.

Keywords: análisis factorial, clustering, cuestionario de satisfacción, diseño de software, dimensiones de calidad

1 Introducción

Es una práctica común medir la calidad educativa de un curso a través de cuestionarios de satisfacción, siempre fundamentados en la propuesta de Kan "la satisfacción del cliente es la validación final de calidad" [1], se puede inducir que si se mejora la satisfacción de los clientes de un curso, también se estará mejorando la calidad de la educación entregada a través de él. En este caso se considera al alumno como el cliente de una institución educativa que le brinda un servicio (educación) [2].

Los cuestionarios de satisfacción han sido generados siguiendo los lineamientos sugeridos por Bob Hayes [3] para medir la satisfacción de clientes. Este proceso fue inicialmente adaptado para la medición de clientes de software en [4] [5] y para la medición de calidad de cursos de software [6] [7] [8] desde la perspectiva de los alumnos.

Lo relevante de la información extraída y procesada de estos instrumentos es que permite identificar posibles problemas y ayuda a tomar medidas correctivas, en este caso dentro del ámbito de la cátedra, creando espacios para la reflexión e implantando un proceso de mejora continua, que sin dudas dará un aporte al mejoramiento de la calidad educativa.

En la publicación presentada en [9], se realizó un proceso estadístico más detallado, los resultados permitieron no sólo detectar las falencias en la calidad del dictado del curso, sino también establecer prioridades para mejorarla. Este proceso y análisis estadístico posibilitó identificar características o aspectos puntuales que deben

ser evaluados, estudiados y considerados para mejorar la calidad de un curso, algo que no sucede con una interpretación simple a los resultados de encuestas.

En el presente trabajo, que se expone a continuación, se aplica la técnica de Análisis Factorial de Correspondencias Múltiples [10] para obtener una clasificación (clusters), bajo la forma de perfiles de respuestas. El mismo está ordenado de la siguiente manera: en la sección 2 se presenta el caso de estudio, en la 3 se expone la encuesta, las dimensiones de calidad definidas y el estudio de fiabilidad del instrumento de medición, en la 4 un procesamiento estadístico simple realizado a los resultados de la encuesta y en la sección 5 la aplicación del Análisis Factorial, objetivo de la presente. A continuación las conclusiones y por último agradecimientos y referencias.

2 Caso de estudio

La cátedra Diseño de Software se dicta en 4º año de las carreras Licenciatura en Ciencias de la Computación y Licenciatura en Sistemas de Información, de la Facultad de Ciencias Exactas, Físicas y Naturales de la Universidad Nacional de San Juan. Los contenidos de la asignatura se enfocan principalmente al Diseño Orientado a Objetos usando UML. No es objeto de la presente sección ahondar en la cátedra en sí, para más información remitirse a [9], [11], [12].

Al finalizar el cursado los alumnos completan una encuesta de satisfacción, generada según el proceso de medición de satisfacción [3] mencionado en la introducción, y administrada, para este caso, por www.encuestafacil.com [13], a fin de recopilar, en forma anónima, las respuestas de los alumnos y obtenerlas en forma digital.

Para este caso de estudio se trabajó con los datos de las respuestas de los alumnos de las cohortes 2014 y 2015.

3 Encuesta

La encuesta de satisfacción generada para la cátedra Diseño de Software años 2014 y 2015 se muestra en la tabla 1. Permite medir el nivel de satisfacción de los alumnos, en base a niveles de acuerdo, usando formato de respuesta tipo Likert [14]–[16]. Esta escala permite obtener valores cuantitativos de respuestas cualitativas. Los alumnos contestan cada enunciado en base a una escala cuantificada con valores discretos de 1 a 5, siendo 1 muy en desacuerdo o muy insatisfecho y 5 muy de acuerdo o muy satisfecho con el enunciado.

Tabla 1. Enunciados de la Encuesta

Nº	Enunciado
1	La teoría es apropiada para los contenidos de la materia
2	La metodología de las clases teóricas permite el entendimiento de los contenidos
3	Los ejemplos dados en la teoría ayudan a la comprensión de los contenidos
4	La teoría dada permite la resolución de los prácticos correspondientes
5	La práctica está relacionada a la teoría vista en clase

6	La práctica permite madurar los conceptos teóricos
7	La práctica realizada permite comprender el objetivo y uso de los Diagramas UML
8	La metodología desarrollada en la práctica permitió asimilar los contenidos
9	La práctica es útil para aprender a aplicar los Diagramas UML
10	La predisposición de las profesoras para dictar la materia es buena
11	Pude interactuar con las profesoras adecuadamente
12	Las profesoras demuestran tener experiencia en los contenidos de la materia
13	La explicación de los contenidos teóricos es clara
14	Las clases teóricas demuestran preparación y esfuerzo en cuanto a su metodología
15	La explicación de los ejercicios prácticos ayudaron a comprender la tarea a realizar
16	El profesor me guió en la resolución de los ejercicios prácticos
17	El parcial de teoría puede desarrollarse en el tiempo asignado
18	Los temas evaluados en el parcial de teoría son los vistos en clase
19	El parcial de teoría puede desarrollarse con los conocimientos adquiridos
20	El parcial de práctica puede resolverse en el tiempo asignado
21	Los ejercicios evaluados en el parcial de práctica son equivalentes a los vistos en clase
22	El parcial de práctica puede desarrollarse con los conocimientos adquiridos
23	La ejercitación práctica realizada en clase permitió resolver el parcial
24	La evaluación entre pares del parcial de práctica fue útil para reafirmar los conocimientos
25	Al realizar la evaluación entre pares pude aprender otra manera de resolver el ejercicio
26	La evaluación entre pares me sirvió para ampliar mi punto de vista
27	La evaluación entre pares reforzó mi capacidad de crítica
28	La evaluación entre pares me sirvió como método de aprendizaje
29	Los apuntes de cátedra incluyen todos los temas del contenido de la materia
30	Los apuntes de cátedra son claros y comprensibles
31	Los ejemplos de los apuntes de cátedras aclaran la teoría que ejemplifica
32	Los apuntes de cátedra fueron suficientes para evacuar las dudas
33	Los tiempos asignados para teoría y para práctica son adecuados
34	El nivel (contenidos y exigencias) de la materia es apropiado para alumnos de 4º año
35	Los contenidos de la materia son dictados en forma organizada, entre teoría y práctica
36	Los contenidos de la materia son dictados en tiempo y forma
37	Los contenidos de la materia son interesantes
38	Me gustó mucho la materia
39	Estoy conforme con el dictado de la materia

Los enunciados de la encuesta son agrupados en diferentes dimensiones de calidad, según el enfoque de incidentes críticos propuesto por Flanagan [17] y se muestran en la tabla 2.

Tabla 2. Dimensiones de Calidad

Nº Dim	Dimensión de calidad	Enunciados
1	metodología de clases teóricas	1,2,3
2	relación entre teoría y práctica	4,5,6
3	metodología de clases prácticas	7,8,9
4	predisposición de los docentes	10,11,12
5	actitud docente teoría	13,14
6	actitud docente práctica	15,16
7	parcial de teoría	17, 18, 19
8	parcial de práctica	20, 21, 22, 23
9	metodología de evaluación entre pares	24, 25, 26, 27, 28

10	apuntes de cátedra	29, 30, 31, 32
11	planificación y contenidos	33, 34, 35, 36
12	satisfacción general	37, 38, 39

Es importante medir, pero debemos asegurarnos que el instrumento de medición esté bien generado y que los resultados brindados serán fidedignos.

Con este fin fueron procesadas las 24 encuestas para el año 2014 y 18 para el año 2015. Como las variables (enunciados) que se definieron en la encuesta son subjetivas, se realizó el análisis de fiabilidad del instrumento de medición, calculando el coeficiente de Alfa de Cronbach [18] [9]. Los valores de alfa hallados fueron de 0,978 y 0,979 para los años 2014 y 2015 respectivamente. Estos valores altos del alfa de cronbach nos garantiza que el instrumento de medición es fiable, con lo cual no es necesario exponer en este trabajo el análisis de correlación artículo-total [3], [9] realizado.

4 Procesamiento estadístico de los resultados

Para este procesamiento se utilizó el software SPSS 15.0 para Windows © [19]. Se realizó un estudio simple, analizando las tablas de frecuencias obtenidas de cada dimensión y sus porcentajes, de cada uno de los años. Los datos fueron agrupados en intervalos, reuniendo de este modo los no satisfechos (1-2,99), los dudosamente satisfechos (3-3,50), los satisfechos (3,51-3,99) y muy satisfechos (4-5), como se muestra en la siguiente tabla de porcentajes por dimensión. No se considera la satisfacción general (Dim12), ya que es una variable resumen de la encuesta.

Tabla 3. Porcentajes de frecuencias por dimensión año 2014

Intervalos	dim1 %	dim2 %	dim3 %	dim4 %	dim5 %	dim6 %	dim7 %	dim8 %	dim9 %	dim10 %	dim11 %
1-2,99	0,0	4,2	4,2	0	0,0	8,3	0,0	8,3	8,3	4,2	8,3
3-3,50	16,7	16,7	16,7	12,5	12,5	4,2	16,3	16,7	12,5	16,7	16,7
3,51-3,99	0	0	0	0,0	12,5	4,2	0,0	4,2	0	8,3	0
4,00-5,00	83,3	79,2	79,2	87,5	75,0	83,4	83,4	70,8	79,2	70,9	75,0

Analizando un caso particular para interpretar la tabla 3, tomando la dimensión 11: "planificación y contenidos" se puede observar que el 8,3% de los encuestados opinaron estar insatisfechos, el 16,7% puntuó dudosamente satisfechos, es decir no saben o están indecisos, están en el borde del acuerdo y el desacuerdo y el 75% de los encuestados están muy satisfechos con la planificación y contenidos de la cátedra.

La tabla 3 dio origen al gráfico que se muestra en la figura 1. En este gráfico se observa a simple vista los porcentajes de frecuencia por rango de cada dimensión.

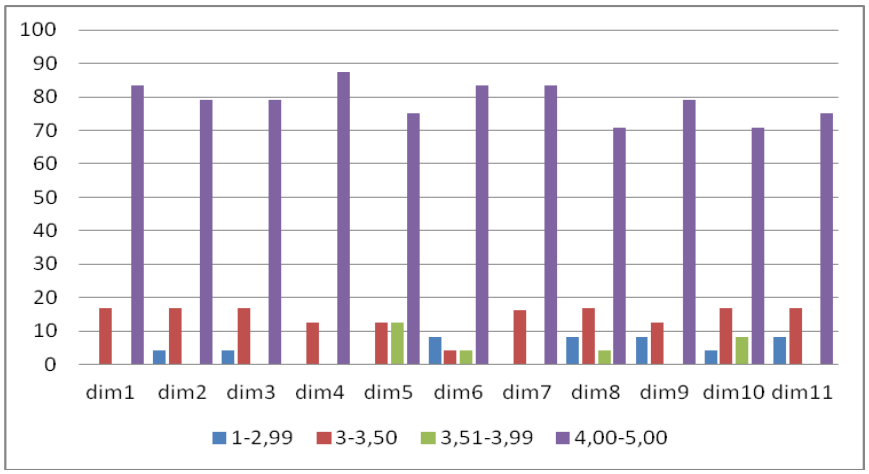


Fig. 1. Porcentajes de frecuencias de intervalos por dimensión año 2014

Para tener una "idea global o general" de los resultados de la encuesta del año 2014 se promediaron los puntajes por intervalo, de todas las dimensiones y se obtuvo el gráfico que se presenta en la figura 2. Este gráfico es un indicador representativo de cómo fueron las puntuaciones promedio por rango. Se podría decir que, en base a las dimensiones consideradas para medir la calidad del curso diseño de software, el 79% están de acuerdo y muy de acuerdo, y un 3% están casi de acuerdo, con lo que se acumula un 82% de satisfacción respecto a un 18% de insatisfacción. Este gráfico exterioriza la calidad del curso, con un proceso estadístico simple basado en las frecuencias.

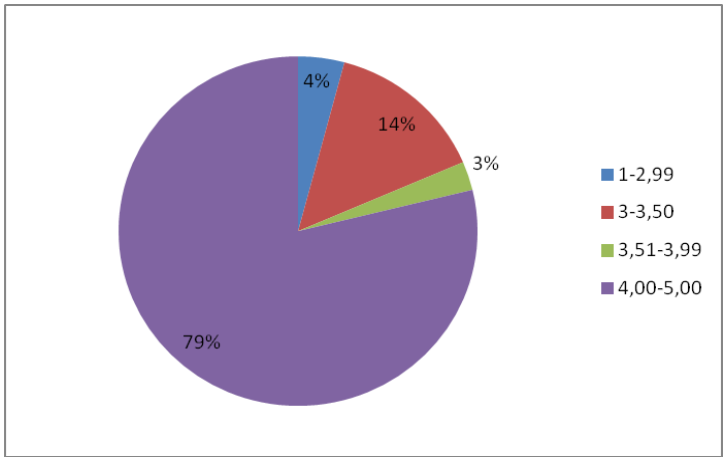


Fig. 2. Porcentajes de frecuencias por intervalo, para el año 2014

De la misma manera que para el año 2014, se realiza el proceso estadístico correspondiente a las encuestas del año 2015. La tabla de porcentajes de frecuencias por dimensión se presenta en la tabla 4.

Tabla 4. Porcentajes de frecuencias por dimensión año 2015

Intervalos	dim1 %	dim2 %	dim3 %	dim4 %	dim5 %	dim6 %	dim7 %	dim8 %	dim9 %	dim10 %	dim11 %
1-2,99	5,6	0	5,6	0	16,7	5,6	5,6	0	5,6	5,6	0
3-3,50	11,1	5,6	5,6	5,6	22,2	16,7	16,7	11,1	16,7	11,1	16,7
3,51-3,99	0	0	0	0	0	0	0	5,6	0	0	0
4,00-5,00	83,4	94,4	88,9	94,4	61,1	77,8	77,8	83,3	77,8	83,4	83,4

En la dimensión 11 de la tabla 4 se puede observar que ninguno de los encuestados opinó estar insatisfechos, el 16,7% no saben o están dudosamente satisfechos y el 83,4% de los encuestados están muy satisfechos con la planificación y contenidos de la cátedra.

Con la tabla 4 se obtuvo el gráfico que se muestra en la figura 3. Acá se representan los porcentajes de frecuencia por rango de cada dimensión y se puede visualizar que en todas las dimensiones prevalece el rango "de acuerdo y muy de acuerdo" (valores entre 4-5).

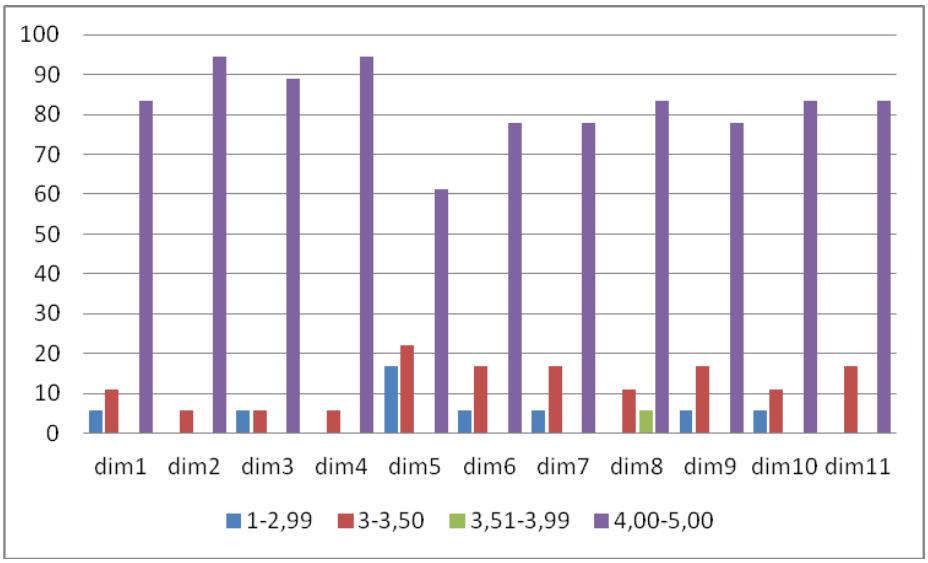


Fig. 3. Porcentajes de frecuencias de intervalos por dimensión año 2015

En la figura 4 se presenta el gráfico correspondiente a los promedios por rango de todas las dimensiones del año 2015. Tal como se puede apreciar, el 82% están de acuerdo y muy de acuerdo, dejando muy poco porcentaje para el resto de los rangos, es decir que la mayoría están en el rango de la satisfacción.

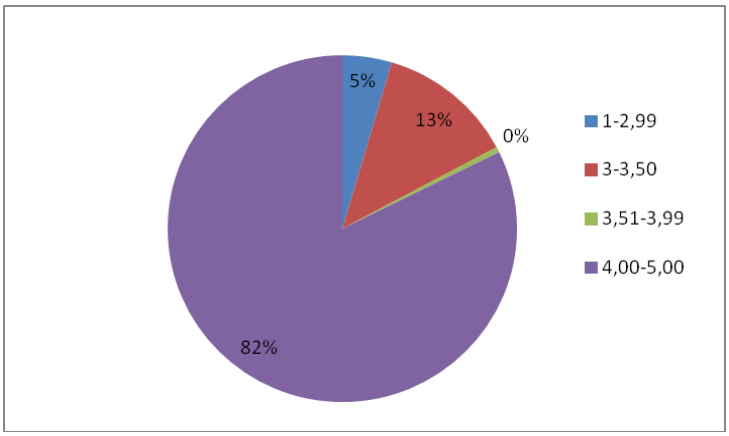


Fig. 4. Porcentajes de frecuencias por intervalo, para el año 2015

5 Aplicación del Análisis de Correspondencias Múltiples a los resultados de las encuestas

Con el fin de comparar los resultados de las dos cohortes se realizó un test de comparación del Nivel de Satisfacción General (Dim 12). Como no se cuentan con variables continuas que verifiquen los supuestos de pruebas T, se realiza prueba no paramétrica donde se comparan las medianas, por tratarse de variables ordinales.

La prueba confirmó la hipótesis nula "no hay diferencias significativas en los dos grupos", con lo cual se consideró un solo conjunto con las respuestas de todos los alumnos (42) para realizar el estudio.

Se realizó una clasificación de las respuestas de las encuestas mediante Análisis de Factores de Correspondencias Múltiples (AFCM) [10] usando el software SpaD-N [20]. Una vez que se obtuvieron los factores que concentran la mayor proporción de inercia se aplicó un Análisis Cluster usando también el software SpaD-N. Este análisis es un método estadístico multivariado de clasificación no supervisada y trata, a partir de una tabla de datos (individuos-variables), situar a los individuos en grupos homogéneos o conglomerados, de manera que los sean considerados similares (comportamiento de las respuestas similares), sean asignados a un mismo cluster o grupo. Este análisis es una técnica de clasificación post hoc pues el número de cluster no es conocido de antemano y los grupos se crean en función de la naturaleza de los datos.

Este análisis considera simultáneamente todas las respuestas de un individuo y clasifica "perfiles de respuestas" de manera tal que, individuos en un mismo grupo muestran características más similares respecto a individuos correspondientes a otros grupos. Esta técnica permite identificar las respuestas más comunes en cada uno de los grupos.

Para este estudio se seleccionaron las variables más correlacionadas a sus dimensiones, considerándolas como las más representativas para medir, y a su vez sólo se consideraron las de aquellas dimensiones más correlacionadas con la dimensión satisfacción general. Estas se muestran en la tabla 5.

Tabla 5. Ítems considerados para los determinar los perfiles

Nº	Enunciado
2	La metodología de las clases teóricas permite el entendimiento de los contenidos
5	La práctica está relacionada a la teoría vista en clase
8	La metodología desarrollada en la práctica permitió asimilar los contenidos
11	Pude interactuar con las profesoras adecuadamente
17	El parcial de teoría puede desarrollarse en el tiempo asignado
22	El parcial de práctica puede desarrollarse con los conocimientos adquiridos
31	Los ejemplos de los apuntes de cátedras aclaran la teoría que ejemplifica
36	Los contenidos de la materia son dictados en tiempo y forma
38	Me gustó mucho la materia

Debido a la naturaleza de estas respuestas, el análisis identificó 3 clusters. Estos se presentan las tablas 6, 7 y 8 en donde se visualizan las modalidades de respuestas compartidas por las encuestas clasificadas para cada grupo. El valor del estadístico del test es quien determina la importancia relativa de una respuesta, se consideran importantes aquellos cuyo valor absoluto es superior a 2 [21].

Se observan también los porcentajes de cada una de estas modalidades de respuesta en el grupo o clase conformada y en el total de la población para poner de manifiesto el por qué los grupos obtenidos se han rotulado como: “Altamente Satisfechos”, “Medianamente Satisfechos” e “Insatisfechos”.

Tabla 6. Modalidad de respuesta para el grupo “altamente satisfecho”

Valor del Test	Clase / Mod (%)	Mod / clase (%)	Global (%)	Modalidad	Variable
6,09	92,86	100	33,33	Muy Satisfecho	Enunciado 22
4.62	**90,91**	**76,92**	**26,19**	**Muy Satisfecho**	**Enunciado 5**
4,25	83,33	76,92	28,57	Muy Satisfecho	Enunciado 38
4,18	90	69,23	23,81	Muy Satisfecho	Enunciado 36
3,08	52,17	92,31	54,76	Muy Satisfecho	Enunciado 11
2,49	75	46,15	19,09	Muy Satisfecho	Enunciado 31
-2,67	14,81	30,77	64,29	Satisfecho	Enunciado 36
-2,89	**12**	**23,08**	**59,52**	**Satisfecho**	**Enunciado 5**

El 30,95% de los 42 alumnos están comprendidos en el cluster o grupo “Altamente Satisfecho”.

Esta tabla se interpreta de la siguiente manera: respecto de la variable “La práctica está relacionada a la teoría vista en clase”, el 26,19% de todos los alumnos del curso opinaron que estaban muy satisfechos, el 76,92% de los alumnos que pertenecen a este grupo opinaron que estaban muy satisfechos, y, de todos los alumnos muy satisfechos en este ítem el 90,91% pertenece a este grupo.

De la misma manera, el 59,52% de todos los alumnos del curso opinaron que están satisfechos, el 23,08% de los alumnos que pertenecen a este grupo opinaron que están satisfechos, y, de todos los alumnos satisfechos en este ítem el 12% pertenece a este grupo.

Tabla 7. Modalidad de respuesta para el grupo "medianamente satisfecho"

Valor del Test	Clase / Mod (%)	Mod / clase (%)	Global (%)	Modalidad	Variable
4,40	94,74	75	45,24	Satisfecho	Enunciado 22
4.06	**81,48**	**91,67**	**64,29**	**Satisfecho**	**Enunciado 36**
4,05	84,00	87,50	59,52	Satisfecho	Enunciado 5
2,63	86,67	54,17	35,71	Satisfecho	Enunciado 11
-3,05	16,67	8,33	28,57	Muy Satisfecho	Enunciado 38
-3,15	**10**	**4,17**	**23,81**	**Muy Satisfecho**	**Enunciado 36**
-4,33	0	0	26,19	Muy Satisfecho	Enunciado 5
-4,47	7,14	4,17	33,33	Muy Satisfecho	Enunciado 17

El 57,14% de los 42 alumnos están comprendidos en el cluster o grupo "Medianamente Satisfecho".

La interpretación se puede realizar siguiendo el mismo criterio que en la tabla 6.

Tabla 8. Modalidad de respuesta para el grupo "insatisfecho"

Valor del Test	Clase / Mod (%)	Mod / clase (%)	Global (%)	Modalidad	Variable
3,13	100	60,00	7,14	Neutral	Enunciado 36
2.98	**57,14**	**80,00**	**16,67**	**Neutral**	**Enunciado 22**
2,71	75,00	60,00	9,52	Neutral	Enunciado 11
2,41	60,00	60,00	11,90	Neutral	Enunciado 5

El 11,90% de los 42 alumnos están comprendidos en el cluster o grupo "Insatisfecho".

La interpretación se puede realizar siguiendo el mismo criterio que en la tabla 6.

6 Conclusiones

Con el procesamiento estadístico simple expuesto en la sección 4 podemos concluir que para el año 2014 (tabla 3, figura 1) las dimensiones más satisfechas son las 4 "Predisposición de los docentes", 6 "Actitud de las docentes de práctica" y 7 "parcial de teoría", y la más insatisfecha la 8 "parcial de práctica" (el 25% de los 24 alumnos puntuaron entre 1-3,50). Para el año 2015 (tabla 4, figura 3) las más satisfechas son las dimensiones 2 "Relación entre teoría y práctica", 4 "Predisposición de los docentes" y 3 "Metodología de clases prácticas" y la más insatisfecha la 5 "actitud docente de teoría" (donde el 38,9% de los 18 alumnos puntuaron entre 1-3,50).

Se debería hacer hincapié en esas dimensiones más insatisfechas y detectar cual es el motivo de insatisfacción e incorporar mejoras. También convendría hacer un estudio analítico-descriptivo de los comentarios y recomendaciones que proponen los

alumnos en la misma encuesta, en espacios específicos para tal fin y que no han sido contemplados en el procesamiento.

Con respecto a los porcentajes de frecuencia por intervalo (figuras 2 y 4) se observa un comportamiento similar, ambos años presentan un 82% de alumnos satisfechos y muy satisfechos y un 18% de alumnos indecisos o insatisfechos, lo cual, a nivel global, el curso es de calidad.

Esta semejanza es corroborada con el test de hipótesis, como se informa en la sección 5. Esto permitió unificar las encuestas e identificar como es el comportamiento o la actitud de todos los alumnos respecto al curso. El clustering realizado con los factores obtenidos por el AFCM identificó 3 grupos (clusters). Un grupo denominado "altamente satisfechos" que abarca el 30,95% del total de alumnos (tabla 6), otro "medianamente satisfechos" que abarca el 57,14% del total de alumnos (tabla 7) y un 11,90% de todos los alumnos pertenecen al grupo de los "insatisfechos" (tabla 8). Este es el grupo de alumnos que hay que identificar para analizar las razones de su insatisfacción.

Se observa que si bien fueron 12 las variables consideradas para realizar este estudio, varias de ellas no aparecen en las tablas mencionadas, esto es porque el valor absoluto del test no es superior a 2, con lo cual no son influyentes para la determinación de las características del grupo.

Respecto del grupo de alumnos "altamente satisfechos" se puede concluir que el 100% de ellos opinaron Muy Satisfactoriamente al enunciado "22", el 100% de ellos puntuaron Satisfactoria y Muy Satisfactoriamente al enunciado "5", el 100% de ellos opinaron satisfactoria y muy satisfactoriamente al enunciado "36", el 92% de los que pertenecen a este grupo opinaron muy satisfactoriamente al enunciado "11" y el 76,92% puntuaron Muy satisfactoriamente al enunciado "38".

En cuanto al grupo de alumnos "medianamente satisfechos", el 95,84% opinaron satisfactoria y muy satisfactoriamente al enunciado "36", el 87,5% opino satisfactoriamente al enunciado "5", y el 75% de los que pertenecen a este grupo puntuaron satisfactoriamente al enunciado "22".

Y, en cuanto al grupo de los "insatisfechos", el 80% de ellos es neutral respecto al enunciado "22", y el 60% opina neutralmente a los enunciados "36, 11 y 5"

Más allá de los resultados estadísticos obtenidos respecto a la calidad del curso Diseño de Software, lo más importante, y que se pretende expresar en este trabajo, es el proceso de medición realizado para obtener esa información.

El AFCM toma las respuestas dadas por los alumnos, a un instrumento de medición elaborado, según un proceso probado de generación de encuestas, cuya fiabilidad ha sido debidamente demostrada. Este análisis permitió caracterizar a los alumnos, en base a sus similitudes de respuestas, en 3 clusters o grupos. En este caso analizando el grado de satisfacción de los alumnos respecto al curso, la técnica nos advierte que hay un cluster, que abarca el 11,90% del total de alumnos, sobre el cual debemos trabajar y tomar acciones correctivas y de mejora en el curso, para disminuir la insatisfacción.

En este trabajo se expuso un proceso de medición de calidad de un curso de diseño de software y también cómo reconocer, a través del AFCM, perfiles de respuestas, identificando las actitudes generales de los alumnos con respecto a ese curso. Medir la satisfacción de un curso permite tomar acciones, lograr transformaciones y cambios

necesarios para mejorar la calidad educativa, no sólo del curso, sino también de la Universidad.

7 Agradecimientos

A la cátedra y alumnos años 2014 y 2015 de la materia "Diseño de Software".

Este trabajo se presenta en el marco de los proyectos: "Modelo de Casos de Uso. Línea base para el proceso de desarrollo de software" y "Determinación y Comparación de Perfiles Sociales y Culturales de estudiantes universitarios a través de Técnicas Estadísticas Multivariadas". Res. Nº18/14-CS.

Referencias

[1] S. H. Kan, *Metrics and Models in Software Quality Engineering*, 2nd ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2002.

[2] D. Tsichritzis, «Reengineering the University», *Commun ACM*, vol. 42, n.º 6, pp. 93–100, jun. 1999.

[3] B. E. Hayes, *Measuring customer satisfaction: survey design, use, and statistical analysis methods*, 2.ª ed. ASQ Quality Press, 1998.

[4] M. I. Lund, S. Zapata, y M. Herrera, «Proposal to Measure Software Customers Satisfaction», en *Argentine Symposium on Software Engineering ASSE'2000 (29 JAIIO)*, Tandil, 2000, pp. 185-197.

[5] M. I. Lund, B. Forcada, M. Herrera, y S. Zapata, «Una experiencia en la obtención de un instrumento fiable para medir satisfacción de clientes de software», en *Proceedings JISBD 2001*, Almagro (Ciudad Real) España, 2001, pp. 437-448.

[6] M. I. Lund, S. F. Ochoa, y S. Zapata, «Un Método para medir la satisfacción de usuarios de Courseware en escenarios de educación a distancia», en *Proceedings JCC2001*, Punta Arenas, Chile, 2001.

[7] S. Zapata, M. I. Lund, M. Herrera, y N. Recabarren, «Elaboración de un cuestionario fiable para medir Satisfacción de Alumnos de Cursos Universitarios de Ingeniería de Software», en *The Seventh International Conference on Engineering and Technology Education – Intertech 2002*, Brasil, 2002.

[8] M. Herrera, S. Zapata, L. Aballay, Berón, F., E. Torres, y S. Ruiz, «Diagnóstico de la Calidad Educativa en las Ciencias Básicas, desde la perspectiva del alumno», en *Proceedings WCCSETE 2006*, São Paulo - Brazil, 2006.

[9] M. I. Lund y M. Herrera, «Curso Diseño de software. Una propuesta para medir la calidad educativa», en *XX CACIC - III Workshop de Innovación en Educación en Informática (WIEI).*, La Matanza, Buenos Aires, 2014, pp. 11-21.

[10] Benzecri, *Correspondence Analysis Handbook*. CRC Press, 1992.

[11] M. I. Lund, L. N. Aballay, M. C. Gomez, y E. G. Ormeño, «Propuesta de un Proceso de Software de Enseñanza-Aprendizaje para la asignatura Diseño de Software», *Rev. Iberoam. Tecnol. En Educ. Educ. En Tecnol.*, vol. 12, n.º especial nº 12, pp. 44-52, abr. 2014.

[12] M. I. Lund, «Diseño de Software basado en Casos de Uso. Aplicación práctica dirigida por un proceso de enseñanza-aprendizaje propuesto», en *Proceedings ASSE-JAIIO 2014. en edicion*, Buenos Aires, Argentina, 2014.

[13] «encuestas online - software encuesta - crea y envia cuestionarios facilmente - Encuesta Facil.» [En línea]. Disponible en: http://www.encuestafacil.com/Default.aspx. [Accedido: 04-ago-2015].

[14] R. Likert, *A technique for the measurement of attitudes*, vol. 22, 140 vols. s.n., 1932.
[15] F. J. Elejabarrieta y L. Iñiguez, «Construccion de escalas de actitud tipo thurst y likert», *Math. Comput.*, vol. 15, n.º 75, p. 299, 1984.
[16] Mendez H, Luz y Peña M, Jose A., *Manual Práctico para el diseño de la escala Likert*, vol. 1. Universidad Autónoma de Nuevo León-UANL, Mexico DF: Trillas, 2006.
[17] J. C. Flanagan, «The Critical Incident technique», *Psychol. Bull.*, vol. 51, n.º 4, p. 33, 1954.
[18] L. J. Cronbach, «Coefficient alpha and the internal structure of tests», *Psychometrika*, vol. 16, n.º 3, pp. 297-334, sep. 1951.
[19] *Software SPSS 15.0 para Windows*. .
[20] «Spad. Get the software safe and easy.», *Software Informer*. [En línea]. Disponible en: http://spad.software.informer.com/. [Accedido: 04-ago-2015].
[21] A. Agresti, *Categorical Data Analysis, 2nd Edition*, Second. Wiley, 2002.

Programación Estructurada en un Curso Introductorio. Una Experiencia Explorando Python.

Edith Lovos[1], Tatiana Gibelli[1], Rodolfo, Bertone[2]

[1] Universidad Nacional de Río Negro, Sede Atlántica
8500 Viedma – Río Negro Argentina
{elovos,tgibelli}@unrn.edu.ar

[2]Instituto de Investigación en Informática III-LIDI.
Facultad de Informática, Universidad Nacional de La Plata
50 y 120, 1900 La Plata, Buenos Aires, Argentina
pbertone@lidi.info.unlp.edu.ar

Abstract. En este artículo se presentan los resultados preliminares alcanzados en relación al uso del lenguaje Python, en el marco de la implementación de una propuesta pedagógica para la enseñanza y aprendizaje de la programación estructurada. La experiencia se desarrolló en el marco de una asignatura correspondiente al primer año de la carrera Licenciatura en Sistemas de la Sede Atlántica de la Universidad Nacional de Río Negro. A partir de la experiencia, se analizan los beneficios y debilidades que presenta el lenguaje propuesto desde la perspectiva docente.

Keywords: Aprendizaje, Programación, Python, Ingresantes

1 Introducción

La asignatura que da lugar a la experiencia que se menciona en esta comunicación se denomina Programación de Computadoras I, en adelante PCI. Esta materia forma parte del plan de estudios de la carrera Licenciatura en Sistemas de la Sede Atlántica de la Universidad Nacional de Río Negro y pertenece al área Algoritmos y Lenguajes de Programación. La misma se dicta en forma presencial en el primer cuatrimestre del primer año con un total de 96 horas.

Cómo objetivos generales, desde PCI, se busca que los alumnos puedan analizar problemas resolubles con computadora, poniendo énfasis en la modelización, la abstracción de funciones y en la descomposición funcional de los mismos, a partir de un paradigma procedural/ imperativo. Se realiza una introducción de las nociones de estructuras de datos, tipos de datos y abstracción de datos.

En relación al enfoque adoptado para la enseñanza, se persigue el aprendizaje del proceso de resolución de problemas (análisis, diseño e implementación) en el marco

de actividades que promuevan en los alumnos el desarrollo de otras habilidades, como el trabajo en equipos colaborativos, el razonamiento crítico, y la capacidad de comunicación; en un espacio que no será siempre el del salón presencial de clases; como por ejemplo el uso del aula virtual de la asignatura.

En este sentido, desde el año 2013, se vienen desarrollando conjuntamente con las actividades de teoría y práctica del curso, actividades de laboratorio, denominadas Actividades Prácticas Entregables (APE) [1], que buscan a través de la resolución de problemas, la integración de los conceptos analizados en el curso. Las consignas de trabajo que se proponen a través de las APE, incluyen la definición del problema, formas de entrega consistente en un cronograma de actividades, fechas de previstas para cada etapa del proceso de resolución y recursos TIC a utilizar para su desarrollo, información sobre pruebas (se definen o se proporcionan los datos con los que serán puestos a prueba los programas y consideraciones especiales).

Desde los inicios de la carrera en el año 2009, el lenguaje de programación utilizado ha sido Pascal. Los resultados alcanzados con el uso de este lenguaje, coinciden con otros estudios a nivel nacional, que señalan por una parte dificultades técnicas como compiladores para plataformas obsoletas, problemas de configuración [2], y también otros relacionados a la empatía que los alumnos tienen con el lenguaje. A partir de este último concepto y teniendo en cuenta que la nuestra es una Universidad novel, que comenzó su actividad académica hace apenas 7 años y dónde la matrícula de ingresantes a la Lic. En Sistemas no es voluminosa, no contemplar esta visión puede como señalan algunos autores [3, 4], restar al objetivo de sostener a los alumnos a lo largo de la cursada. Por otra, desde otras materias de la carrera que le suceden a PCI existen requerimientos para que los alumnos lleguen a las mismas conociendo otros lenguajes, como por ejemplo C.

2 Explorando Python

Python es un lenguaje de programación creado a principios de los años noventa por el investigador holandés Guido Van Rossum. Y en términos de edad, podríamos decir que es joven comparado con otros lenguajes utilizados en el ámbito académico, como Pascal(1970), ó C(1972) . Una característica interesante teniendo en cuenta el auge del software libre, es su popularidad en esta comunidad. Así se puede acceder al mismo a través del sitio oficial https://www.python.org, evitando cualquier tipo de costos asociados a licencias.

Por otra parte, Python es un lenguaje interpretado, multiplataformas, de sintaxis y semántica sencilla y consistente. Y aunque es un lenguaje orientado a objetos, puede ser utilizado para soportar el paradigma procedural y funcional. Promueve la creación de código legible, maneja tipos dinámicos, gestión dinámica de la memoria (recolector de basura) y maneja excepciones. Al ser fuertemente tipado no esta permitido tratar una variable como si fuera de un tipo diferente al que tiene, sin embargo el lenguaje permite la conversión explícita entre todos los tipos de datos. Se

caracteriza también por su modularidad y por incluir una poderosa y extensa biblioteca de clases [5].

En relación a la formas de ejecutar código en Python, esto se puede hacer en una sesión interactiva (escribiendo líneas directamente en el intérprete y obteniendo la respuesta a cada una de ellas; o es posible escribir el código de un programa en un archivo con la extensión .py y ejecutarlo. Ambas formas pueden llevarse a cabo usando el entorno IDLE (Integrated DeveLopment Environment) que viene con los paquetes de Python.

Philip Guo [6], en un análisis sobre los lenguajes de programación más utilizados en la enseñanza en 39 centros de estudios de los Estados Unidos, señala a Python como el mejor posicionado, en los cursos introductorios a la disciplina. A nivel nacional es posible mencionar las experiencias de la FCEFyN - Universidad Nacional de Córdoba [9], la UNNOBA [2], la asignatura "Algoritmos y Programación 1", de la Facultad de Ingeniería de la UBA [7] y de la carrera de ingeniería informática de la Universidad Nacional de Avellaneda [10]. Asimismo algunas de estas experiencias han dado como resultados recursos bibliográficos y herramientas que se presentan como valiosos materiales de apoyo al uso de Python como lenguajes para la enseñanza y aprendizaje de la programación [8,11,12,13,16].

Por lo aquí expuesto, las características y experiencias de uso en docencia, ubican a Python como una opción posible para el lenguaje de programación que permitirá la implementación de los algoritmos en el marco del curso PCI, de una manera que resulte sencilla y amena para los alumnos que recién se inician en la actividad de programar.

3 Desarrollo de la Experiencia

El programa de la asignatura PCI, consta de seis unidades didácticas, cada una con su correspondiente trabajo práctico y tres Actividades Prácticas Entregables (APE) integradoras de los conceptos tratados en la asignatura. Estas APE serán entregadas y evaluadas y las fechas de publicación de las mismas, están establecidas en el cronograma de actividades de la materia. Así mismo, en el programa de la materia se incluye un apartado especial que indica cómo será el desarrollo y evaluación de las mismas.

La asignatura prevé tres exámenes parciales que permiten evaluar los contenidos de las unidades trabajadas y una instancia global de recuperación al final de la cursada. Respecto al dictado, las actividades presenciales de la materia se desarrollan en dos clases semanales de 3 horas reloj cada una. Cada clase tiene una parte teórica y otra práctica. En la primera, se desarrollan los conceptos teóricos previstos en el plan de estudio haciendo uso de ejemplos prácticos que permitan la aplicación de los conceptos analizados. La parte práctica, tienen como objetivo la aplicación de los conceptos trabajados en la teoría, en la resolución de problemas computacionales, a través del diseño algoritmos. En un paso siguiente estas soluciones serán implementadas usando el lenguaje Python.

El énfasis de la asignatura está puesto en la parte práctica, en este sentido, para desarrollar la habilidad de resolver problemas usando algoritmos es fundamental el entrenamiento que se realiza a través de las situaciones que se plantean en los trabajos prácticos, mediante las cuales los alumnos se enfrentan con situaciones en las que tienen que decidir sobre la naturaleza del problema, seleccionar una representación que ayude a resolverlo (modelo) y, monitorear sus propios pensamientos (metacognición) y estrategias de solución. [14]

Durante la primera unidad del programa, resolución de problemas, se trabaja en papel usando pseudocódigo. Luego, a partir de la unidad II, una de las dos clases semanales, se desarrolla en el laboratorio de Sistemas, de manera que los alumnos puedan realizar la parte práctica en máquina y con la asistencia presencial de los docentes de la cátedra.

Para esta experiencia, el lenguaje de programación propuesto para la implementación de las soluciones algorítmicas, fue Python en su versión 3.4 y el alumno estaba habilitado para utilizar en la edición y ejecución de los programas el entorno IDLE (Iteractive DeveLopment Environment), incluido con Python y/o el laboratorio virtual VPL que se encuentra dentro del aula virtual de la asignatura. Para el desarrollo de los ejercicios de la parte práctica de la materia, los alumnos usaron IDLE, en tanto que VPL se utilizó para el desarrollo de las Actividades Prácticas Entregables (APE), posibilitando que los grupos y el tutor compartieran el mismo espacio de edición de la implementación.

4 Resultados

En relación al pasaje del pseudocódigo a la implementación de los algoritmos usando Python, no se observaron mayores dificultades, entendemos que esto es así, debido a que la sintaxis del pseudocódigo se presenta en español, y como puede verse en el ejemplo de la tabla 1, y no difiere de la Python salvo por el idioma y los caracteres especiales (:) que se presentan como fin de las estructuras de control.

Tabla 1. Comparación Pseudocódigo - Python

Pseudocódigo - Algoritmo contarDig	**Python – Programa contarDig.py**

leer(N)
Num = N
Cd = 0
Mientras (Num > 0)
  Cd = Cd + 1
  Num = Num / 10
mostrar(N," tiene ", Cd, “digitos”)
	N = int(input('Ingrese un nro: '))
	Num = N
	Cd = 0
	while (Num > 0):
	  Cd = Cd + 1
	  Num = Num // 10
	print(N,' tiene ', Cd, 'digitos')

En este pseudocódigo, y al igual que en Python, no se usan delimitadores específicos de bloques, como Begin - End de Pascal, los bloques se delimitan por la

indentación, y cada línea termina sin ningún carácter de corte especial (a diferencia del símbolo **;** de Pascal). Sin embargo uno de los primeros problemas a los que se enfrentaron los alumnos al implementar sus algoritmos en Python estuvo asociado a los nombres de las variables, dónde a diferencia de Pascal, estos resultan sensibles al tipo de letra (mayúsculas/minúsculas) y a las conversiones necesarias al momento de realizar una operación de entrada. Otra situación conflictiva tanto para principiantes como para quienes conocen otros lenguajes tipo Pascal, estuvo vinculado a la idea de variable como contenedor de información, ya que en Python las variables son referencias a posiciones de memoria. Sobre este tema, resulta fundamental el desarrollo de actividades prácticas que le permitan al alumno visualizar el concepto de variable e interpretar sus diferentes usos.

Desde la perspectiva del docente, una de las ventajas de utilizar Python está asociada a la facilidad de instalación del aplicativo, a diferencia de lo que sucedía, en experiencias anteriores utilizando Pascal con Lazarus o Geany. Desde el inicio de uso del lenguaje, las consultas de los alumnos se relacionaron con el problema y la expresión de la solución al mismo (sintaxis y semántica), pero en menor medida con cuestiones asociadas a la configuración del lenguaje de programación o del entorno. Así mismo, quienes llegan al aula con las netbooks del Programa Conectar Igualdad, ya tienen instalado Python, solo fue necesario verificar la versión disponible. Sobre esta cuestión, y coincidiendo con el expresado por Marzal[15], es deseable que el lenguaje de programación tenga en el curso un valor instrumental, interfiriendo lo menos posible en la implementación de los algoritmos.

PCI es un curso, que para la mayoría de los alumnos se presenta como su primer contacto con la actividad de programación, por este motivo es común que durante el aprendizaje, los mismos utilicen el método de prueba y error. Así, en el caso de la experiencia usando Python, el entorno IDLE facilitó la edición y ejecución y la detección de errores, gracias a características como el resaltado de errores, el formateo del código y el resaltado de la sintaxis. Aún cuando el programa presente errores, es posible ejecutarlo y obtener algunos resultados parciales, permitiendo que los alumnos puedan descubrir no solo errores sintácticos sino también de semántica.

En Pascal, esta forma de trabajo implica primero eliminar todos los errores sintácticos y luego enfocarse en la semántica.

En cuanto al manejo de las estructuras de datos que se ven en el curso (arreglos, matrices, registros), el tipo lista de Python, resultó de utilidad para trabajar el concepto de secuencia y los algoritmos que permiten manejar estructuras de este tipo. Sobre este punto, en PCI, primero se presentó el concepto de arreglo como estructura de datos homogénea, de acceso directo y luego se utilizaron las listas de Python para implementar los mismos. Aquí se pudo observar que aunque en la práctica se realizaron las implementaciones de las operaciones asociadas a la estructura como por ejemplo buscar, pertenece, existe; luego en el examen parcial usaban directamente las instrucciones propias de Python. Ejemplo, el operador ***in*** para determinar si un elemento está o no dentro de una secuencia.

Un tema que resulto difícil desde la enseñanza, está asociado al concepto de registro, el cual no existe como tal en Python. Aquí se adoptó la línea presentada en el el libro “Introducción a la programación con *Python* 3” [11] y se introdujo la idea de “

Tipos de Datos a Medida" a través del uso de clases, pero sin necesidad de entrar en detalles propios de la programación orientada a objetos.

Históricamente uno de los temas incluidos en el programa de la asignatura PCI, que resulta más difíciles de apropiar y significar para los estudiantes, es el de modularización. Aún cuando logren realizar un diseño modularizado de la solución al problema, luego será necesario utilizar los recursos del lenguaje de programación para implementarlo. En esta experiencia usando Python, el concepto de modularización, se comenzó a trabajar cuando solo se conocían los tipos de datos simples y así los parámetros a utilizar en los módulos, sean estos funciones o procedimientos, pertenecían solo a los tipos simples. Python no dispone de una forma explícita de definir funciones o procedimientos (Procedures y Function de Pascal), sin embargo cuando un subprograma se comporta como una función y retorna un resultado, se utiliza la palabra reservada return como instrucción final del módulo. Así la diferenciación entre funciones y procedimientos no resultó problemática desde el punto de vista de la implementación para el proceso de enseñanza y aprendizaje. Respecto a el pasaje de parámetros con estos tipos, denominados inmutables para Python, se trabajó de forma equivalente al pasaje de parámetros por valor que utiliza Pascal.

Las dificultades en relación al pasaje de parámetros, surgieron al comenzar a trabajar con los tipos compuestos (listas y listas enlazadas), ya que el manejo es similar al pasaje de parámetros por referencia de Pascal y la ausencia de un mecanismo explicito (var de Pascal) para diferenciar entre estos comportamientos, dificultó asimilar el concepto e implicó necesariamente tratar cuestiones propias de Python como son la mutabilidad e inmutabilidad de los objetos del lenguaje.[5,11]

Respecto a si la utilización de Python ha permitido sostener a los alumnos a lo largo de la cursada, el gráfico 1, muestra la distribución de los alumnos para los años 2013, y 2015, dónde se siguió la misma metodología de trabajo, solo que en 2013 se utilizó Pascal y en 2015 Python.

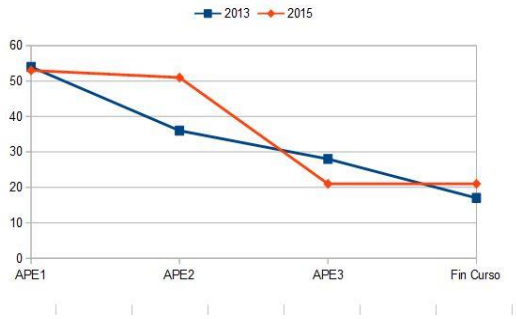


Fig. 1. Distribución de los alumnos durante los cursos 2013 y 2015.

En relación al abandono, aunque en ambos casos los resultados finales son similares, se observa que en la experiencia del año 2015 se ha logrado mantener el nivel de alumnos hasta la mitad del curso. Así mismo, el desarrollo de la primer APE para la cursada 2015, se pudo iniciar luego de transcurrida la semana 5 del curso (en 2013 se inició luego de la semana 6), y entendemos que esta situación se produjo porque resultó más fácil el uso y apropiación del lenguaje por parte de los alumnos.

En este sentido, resulta interesante mencionar que durante la experiencia con Python algunos alumnos utilizaron sus dispositivos móviles (celulares principalmente) para la actividad de programación, posibilitando está actividad fuera del laboratorio físico y en un medio qué les resulta familiar.

4 Conclusiones

Frente a Python, Pascal tiene la ventaja de haber sido diseñado para trabajar las técnicas de la Programación Estructurada. Sin embargo, como el objetivo de PCI no es aprender un lenguaje de programación sino comprender y adquirir destrezas para resolver problemas usando un enfoque procedural; puede decirse entonces que Python ha logrado cumplir ese objetivo. Así, a partir de una sintaxis y semántica sencilla se hace más simple el proceso de implementación de las soluciones algorítmicas. Logrando disminuir la sensación de frustración en los alumnos (principalmente para quienes la programación es una actividad novedosa) en relación a la vinculación con el lenguaje, el entorno de desarrollo, y el proceso de resolución de problemas. De esta forma, el lenguaje de programación se posiciona dentro del curso, como el recurso que permite expresar la solución algorítmica en términos de una sintaxis entendible por un autómata y no como un obstáculo. Todo esto, acompañado por un entorno de desarrollo sencillo (IDLE), con características como el resaltado de sintaxis, la tabulación automática y el resaltado de errores en otras, que facilitan el proceso de implementación.

En relación a los resultados académicos del curso, no puede decirse que Python haya modificado notablemente los mismos, sin embargo y asociado nuevamente a la simplicidad del mismo, permite poner el foco en cuestiones complejas de la resolución de problemas algorítmicos siguiendo el paradigma procedural, como son en esta instancia de la carrera, la modularización, y el manejo de estructuras de datos.

Referencias

1. Lovos, Edith. El uso de estrategias colaborativas mediadas por tecnología. La enseñanza de la programación en el primer año de la Licenciatura en Sistemas de la UNRN. Tesis de Maestría. Facultad de Informática, UNLP. (2014). Disponible:http://hdl.handle.net/10915/38038
2. Osella Massa, G.L., Russo, C. C., Sarobe, M., Pompei, S.: Análisis de nuevos lenguajes para la enseñanza de programación imperativa en los primeros años de las

carreras de Informática de la Universidad Nacional del Noroeste de la Provincia de Buenos Aires. En Actas del XVIII Congreso Argentino de Ciencias de la Computación (2012). http://hdl.handle.net/10915/23860
3. Ramırez, A. O.: Python como primer lenguaje de programación. En: Publicación interna del Tecnológico de Monterrey, Campus Estado de México. (2010)
4. Radenski, A.: Python First: A Lab-Based Digital Introduction to Computer Science. En ItiCSE'06: Proceedings of the 11th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education. pp.197–201 (2006)
5. Peña Rosalia: Paseo por la programación estructurada y modular con Python. En Revista de Investigación en Docencia Universitaria de la Informática. Vol.8 Nro1.SSN 1989-1199 España. (2015)
6. Philip Guo. Python is Now the Most Popular Introductory Teaching Language at Top U.S. Universities. (2014), http://cacm.acm.org/blogs/blog-cacm/176450-python-is-now-the-most-popular-introductory-teaching-language-at-top-us-universities/fulltext
7. Wachenchauzer, R.: Cátedra Algoritmos y Programación 1., https://sites.google.com/site/fiuba7540rw/
8. Wachenchauzer,R., Manterola, M., Curia,M.,Medrano, M., Paez, N. (2010) Algoritmos y Programación I con lenguaje Python. Disponible: https://librosweb.es/libro/algoritmos_python/
9. http://python.org.ar/wiki/Proyectos/UsoDePythonEnLaUniversidad/
10. Wachenchauzer, R. Programación con Python. En Portal Educar. http://www.educ.ar/sitios/educar/recursos/ver?id=121388
11. Marzal A., Gracia Luengo V.I., García,P. (). Introducción *a la programación con Python 3. Publicacions de la Universitat Jaume I. Servei de Comunicació i* Publicacions Campus del Riu Sec. Edifici Rectorat i Serveis Centrals. 12071 Castelló de la Plana. ISBN: 978-84-697-1178-1. (2014)
12. http://www.pythontutor.com/
13. Marzal A., Gracia Luengo, I. : Introducción a la programación con Python y C. Volumen I: Python. Disponible:http://ftp.tku.edu.tw/Linux/Gentoo/distfiles/python.pdf
14. Gary Stager: "En pro de los computadores". Portal Eduteka (2003). Disponible: http://www.eduteka.org/ProComputadores2.php
15. A Marzal, D. Llorens, I. Gracia : Aprender a programar con Python: una experiencia docente. Univ. Jaume I, ftp://metalab.unc.edu/pub/Linux/docs/LuCaS/Presentaciones/200309hispalinux/15/15.pdf
16. Raúl *González Duque. Python para todos.* http://www.ceibal.edu.uy/contenidos/areas_conocimiento/aportes/python_para_todos.

Propuesta didáctica innovadora para el aprendizaje de los sistemas de información administrativa

Laura R. Villarrubia[1], Lía G. Rico[1]

[1] Calle Ítalo Palanca N°10 - Cátedra de Sistemas de Información - Facultad de Ingeniería - UNJu - Jujuy
lauraritavillarrubia@yahoo.com.ar, liagrico@hotmail.com

Resumen. Esta ponencia presenta el desarrollo, resultados y conclusiones del uso de una propuesta didáctica enfocada en la administración estratégica para el planteo de los sistemas de información administrativa. La misma se llevó a cabo en la asignatura Sistemas de Información Administrativa, correspondiente al 5to. año de la Licenciatura en Sistemas, de la Facultad de Ingeniería de la Universidad Nacional de Jujuy. El desarrollo presenta el marco conceptual, la descripción de las dos fases de la propuesta didáctica y los resultados obtenidos. Finalmente, en base a los resultados observados, se valora la propuesta didáctica como válida para el aprendizaje de los sistemas de información administrativa, desde una visión estratégica de los negocios.

Palabras clave: sistemas de información administrativa, administración estratégica, marketing estratégico, procesos de negoción

1 Introducción

Para comprender totalmente los sistemas de información administrativa, es necesario entender las dimensiones más amplias relativas a organización, administración y tecnología de información de los sistemas y su potencial para dar soluciones a retos y problemas del entorno de negocios [1]

Por otro lado es un hecho significativo que las universidades argentinas incluyan en sus áreas de conocimiento contenidos relacionados con Teoría Organizacional, Comportamiento Organizacional, Teoría de la Decisión, Modelos de Negocios, Áreas Funcionales de Negocio, entre otros [2]

Como docentes universitarios no estamos ajenos a esta situación, es por ello que este trabajo presenta una propuesta didáctica que complementa el estudio tradicional de sistemas informáticos, generando en el estudiante la necesidad de pensar y diseñar sistemas de información administrativa alineados con la estrategia global del negocio para constituirse en una verdadera herramienta de soporte a la concreción de la misión de la organización.

2 Marco teórico

2.1 Administración Estratégica

Administración Estratégica es la formulación e implementación de acciones, que mediante el análisis y diagnóstico del ambiente interno y externo de la organización, y enfatizando las ventajas competitivas de la misma, aproveche las oportunidades y sus fortalezas, sortee sus debilidades y se defienda de las amenazas que el ambiente le presenta, siempre en pos de la concreción de sus objetivos y logro de sus metas.

La administración estratégica busca la integración de la alta dirección y de las unidades de línea como: Finanzas, Marketing, Producción, Administración y otras áreas. Las actividades de la administración estratégica se realizan con la participación de todos los integrantes de la organización y de sus niveles jerárquicos. La administración estratégica se refiere a la formulación, implementación y evaluación de la estrategia, en tanto que el planeamiento estratégico se refiere sólo a la formulación estratégica [3]

La administración constituye una técnica para la conducción de organizaciones y el marco teórico se va construyendo asentado en la búsqueda permanente de respuestas para dos interrogantes: ¿Qué es una organización? (problema sistémico) ¿Cuáles son los mecanismos que hacen que funcione bien? (problema psicológico, social y tecnológico) [4]

La estrategia de una organización o empresa, es el plan de acción que los administradores deben desarrollar para posicionarse estratégicamente en el mercado, satisfacer a los clientes, competir con éxito y lograr rentabilidad económica y financiera.

2.2 Marketing Estratégico

La administración de las relaciones con el cliente (CRM) es una estrategia de negocio a nivel de toda la empresa, diseñada para optimizar la rentabilidad y la satisfacción del cliente, enfocándose en grupos de clientes muy bien definidos y precisos [5]

El marketing estratégico busca conocer las necesidades actuales y futuras de los clientes de una empresa, localizar nuevos nichos de mercado, identificar segmentos de mercado potenciales, valorar el potencial e interés de esos mercados, orientar a la empresa en busca de esas oportunidades y diseñar un plan de actuación u hoja de ruta que consiga los objetivos buscados.

En este sentido y considerando que las compañías se mueven en un mercado altamente competitivo se requiere del análisis continuo de las diferentes variables del FODA, no sólo de la empresa considerada sino también de la competencia en el mercado. En este contexto las empresas en función de sus recursos y capacidades deberán formular las correspondientes estrategias de marketing que les permitan adaptarse a dicho entorno y adquirir ventaja a la competencia aportando valores diferenciales.

2.3 Procesos de Negocio

Un proceso de negocio es un conjunto de tareas relacionadas lógicamente, llevadas a cabo para lograr un resultado de negocio definido, el cual puede plasmarse en un producto o un servicio. Los procesos reciben insumos para transformarlos utilizando recursos de la empresa. La transformación comprende una colección de actividades estructurales relacionadas que producen valor para la organización, sus inversores o sus clientes. Los procesos de negocio normalmente atraviesan varias áreas funcionales [6]

Los procesos de negocio abarcan a los procesos de producción, y no deben confundirse. El proceso de producción puede observarse como el proceso que se ocupa de la planificación, administración, control y actividades que son necesarias llevar a cabo para transformar una serie de elementos de entrada en unos elementos de salida. Tiene que ver con aquellas actividades que aumentan la aptitud de los bienes para satisfacer las necesidades del mercado [7]

Hay tres tipos de procesos de negocio: los *procesos estratégicos*, que dan orientación al negocio; los procesos *sustantivos, clave o de generación de valor*, son la parte principal del negocio porque generan algo de valor para el cliente; y los procesos *de apoyo vertical u horizontal*, que dan soporte a los procesos centrales.

Si al analizar una organización, se cambia de un enfoque basado en funciones a una lógica de procesos, se pasa de enfatizar el quién hace qué, al qué se debe hacer para lograr cierto resultado.

La visión funcional descansa en el organigrama de la empresa como modelo fundamental del negocio. En este enfoque el proceso se hace invisible y por lo tanto nadie se responsabiliza por su desempeño de manera integral.

La visión de procesos se orienta al trabajo que debe desarrollar la organización, para que el negocio funcione y entregue un producto o servicio, por el cual un cliente externo está dispuesto a pagar. Es una manera lógica de visualizar y analizar un negocio, porque es la forma en que los clientes lo miran. Los procesos reales cruzan las estructuras organizacionales de manera longitudinal.

2.4 Cadena de Valor

Se denomina cadena de valor a las principales actividades de una empresa comparadas éstas con los eslabones de una cadena; las actividades van añadiendo valor al producto a medida que éste pasa por cada una de ellas. Tomando esta metáfora como una herramienta, se puede desagregar la empresa en actividades, de ese modo, se identifican mejor sus fortalezas y debilidades, especialmente en lo que respecta a fuentes potenciales de ventajas competitivas y costos asociados a cada actividad.

Todas las empresas, cualquiera sea su rubro, cuentan con una cadena de valor, conformada por actividades, que van desde el diseño del producto y la obtención de insumos hasta la distribución del producto y los servicios de post venta.

Esta herramienta clasifica las actividades generadoras de valor de una empresa en dos: *Actividades primarias o de línea*: Son aquellas actividades que están directamente relacionadas con la producción y comercialización del producto, son:

- Logística interior (de entrada): actividades relacionadas con la recepción, almacenaje y distribución de los insumos necesarios para fabricar el producto.
- Operaciones: actividades relacionadas con la transformación de los insumos en el producto final.
- Logística exterior (de salida): actividades relacionadas con el almacenamiento del producto terminado, y la distribución de éste hacia el consumidor.
- Mercadotecnia y ventas: actividades relacionadas con el acto de dar a conocer, promocionar y vender el producto.
- Servicios: actividades relacionadas con la provisión de servicios complementarios al producto tales como la instalación, reparación y mantenimiento del mismo.

Actividades de apoyo o de soporte: Son aquellas actividades que agregan valor al producto pero que no están directamente relacionadas con la producción y comercialización de éste, sino que más bien sirven de apoyo a las actividades primarias.

- Infraestructura de la empresa: actividades que prestan apoyo a toda la empresa, tales como la planeación, las finanzas y la contabilidad.
- Gestión de recursos humanos: actividades relacionadas con la búsqueda, contratación, entrenamiento y desarrollo del personal.
- Desarrollo de la tecnología: actividades relacionadas con la investigación y desarrollo de la tecnología necesaria para apoyar a las demás actividades.
- Aprovisionamiento: actividades relacionadas con el proceso de compras.

La cadena de valor permite identificar las fortalezas y debilidades de una empresa, compararla con empresas competidoras, detectar fuentes potenciales de ventajas competitivas, y comprender mejor el comportamiento de los costos [8]

2.5 Ventaja Competitiva

En términos prácticos, una ventaja competitiva existe cuando una empresa ofrece un producto o servicio que los clientes perciben que es superior a los que ofrecen sus competidores. De ello se sigue que la empresa debe obtener una ventaja que pueda resistir la presión de los competidores. Muchas iniciativas de negocios fracasan porque los emprendedores no entienden la naturaleza e importancia de una ventaja competitiva [5]

2.6 Sistema de Información Administrativa

El S.I.A es el proceso de transformar los datos en información de calidad, que permite a los administradores tomar decisiones, resolver problemas y cumplir con sus funciones/operaciones en forma eficiente y eficaz [9]

Características de los SIA:

- Comprensibilidad: Debe ser elaborado en forma correcta y contener términos y símbolos adecuados para que el receptor lo interprete debidamente.
- Confiabilidad: es necesario que sea preciso, congruente con el hecho, real y comprobable desde la fuente y todo el proceso de transmisión.
- Relevancia: debe ser esencial para el área de responsabilidad de actuación del administrador.
- Integridad: Tiene que contener los hechos que el administrador necesita para tomar decisiones y resolver problemas.
- Concisión: debe omitir materias ajenas al asunto, resumir la información clave y dejar a un lado los detalles y los datos que no tienen relación con el sistema de información administrativa.
- Oportunidad: Tiene que estar disponible en el momento en que sea necesaria para la actuación del administrador, pues anticipada podría ser olvidada o mal usada, y retrasada no tiene valor.
- Calidad: es el grado de precisión con el cual la información retrata la realidad, a un costo razonable, para su procesamiento y distribución.

El plan de SIA ha de formar parte del Plan Estratégico Administrativo.

Existen 2 enfoques para incorporar el SIA en el plan: el Pasivo y Proactivo

a) Enfoque Pasivo: El plan administrativo se elabora primero y después el sistema que satisface las necesidades resultantes.

b) Enfoque Proactivo: Se busca un buen sistema para lograr una ventaja competitiva más amplia, una planeación y mercadotecnia más eficaces del producto y una mejor utilización de los recursos humanos y financieros. Desde este punto de vista, la estrategia del SIA se elabora y se integra con la estrategia global de la empresa.

3 Propuesta didáctica

El objetivo de la propuesta es que el estudiante considere la estrategia del negocio como marco de referencia, para el planteo de los sistemas de información administrativa.

La propuesta incluye dos fases:

1- ***Diseño del caso de estudio***: el problema planteado a los estudiantes según la clasificación de Duch es de nivel 3. Es el nivel de los problemas ABP, requiere capacidad de análisis, síntesis y evaluación. Están relacionados con el mundo real, y no toda la información necesaria para resolverlo está contenida en el propio problema o incluso en los textos del curso. En consecuencia, los estudiantes necesitan hacer investigación, descubrir nuevos materiales y llegar a juicios o decisiones basadas en la información aprendida. El problema puede tener más de una respuesta aceptable [10]

Los diseños que encuentren los grupos de estudiantes es una de muchas soluciones tecnológicas posibles. Como sucede en el mundo real hay más de una respuesta correcta. La solución de diseño de cada grupo estará basada en las suposiciones propias acerca de

las restricciones de la tecnología disponible y limitada solamente por la propia imaginación, creatividad, intuición y sentido común de los estudiantes.

El caso de estudio presenta los datos de identificación de una organización, tipo de organización, ubicación, envergadura, rubro y la descripción del funcionamiento de la misma, obtenida esta de la aplicación de técnicas de relevamiento de información, como entrevistas, observación directa y etnografía. La información que comprende el caso es:

- Cómo se gestó la empresa, hacia dónde se dirige y como es el contexto en el que funciona. Entrevista con el personal estratégico de la organización.
- Sobre los procesos de la empresa. Entrevista con el personal táctico y operativo de la organización.
- Sobre la problemática actual relacionada con el uso de los sistemas de información.
- Sobre el rol que cumple el personal en la organización.
- Sobre el aspecto edilicio.

El caso de estudio propuesto a los estudiantes es una veterinaria denominada "Perro Feliz"

2- ***La guía de actividades*** **para identificar los sistemas de información administrativa en la organización. Cada actividad se ejemplifica con una muestra extraída de los trabajos finales de los estudiantes.**

<u>Actividad 1</u>: Identificar y reconocer necesidades reales del cliente.

El primer paso para iniciar el estudio de los sistemas de información administrativa en la veterinaria es conocer e interpretar con efectividad y precisión las necesidades de los clientes, para ello se desarrollaron los conceptos del marketing estratégico:

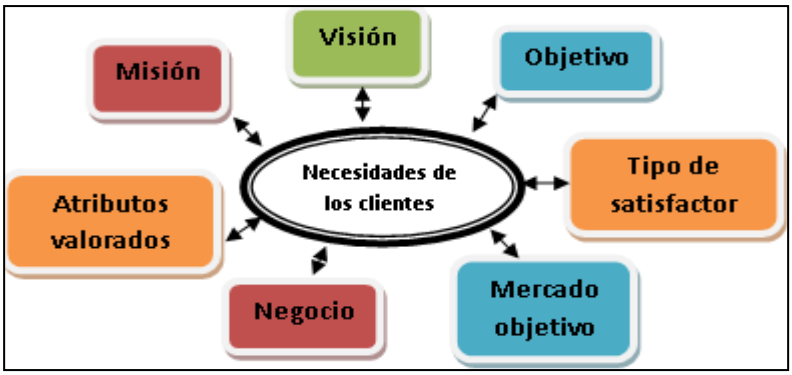


Fig. 1. Marketing Estratégico (elaboración propia)

SATISFACTOR: Servicio de atención canina.

NEGOCIO: Ofrecer a todos los clientes la posibilidad de controlar y mejorar la salud de sus mascotas y la oportunidad de ahorrar tiempo en su cuidado estético en manos de profesionales y con herramientas adecuadas

ATRIBUTOS DEL SATISFACTOR:

Necesidad: Atención médica para mascota rápido y seguro.

Expectativas: Rapidez. Transparencia.

Deseos: Ambiente agradable. Buena atención. Buen trato para con la mascota.

Atributos: Tecnología innovadora aplicada al servicio. Buena calidad en el servicio de atención médica y estética. Bajo costo del servicio.

Actividad 2: **Modelar los procesos de negocio**

El próximo paso, luego de conocer con precisión el servicio que brindará la veterinaria, es determinar el límite y alcance del proceso productivo de la veterinaria de acuerdo al satisfactor identificado y modelar los procesos de la veterinaria a través de técnica SADT (Structured Analysis and Design Technique) y el diagrama de flujo "Anteproyecto del servicio" que permite identificar las actividades que el cliente puede ver o en las que pueda participar [11]

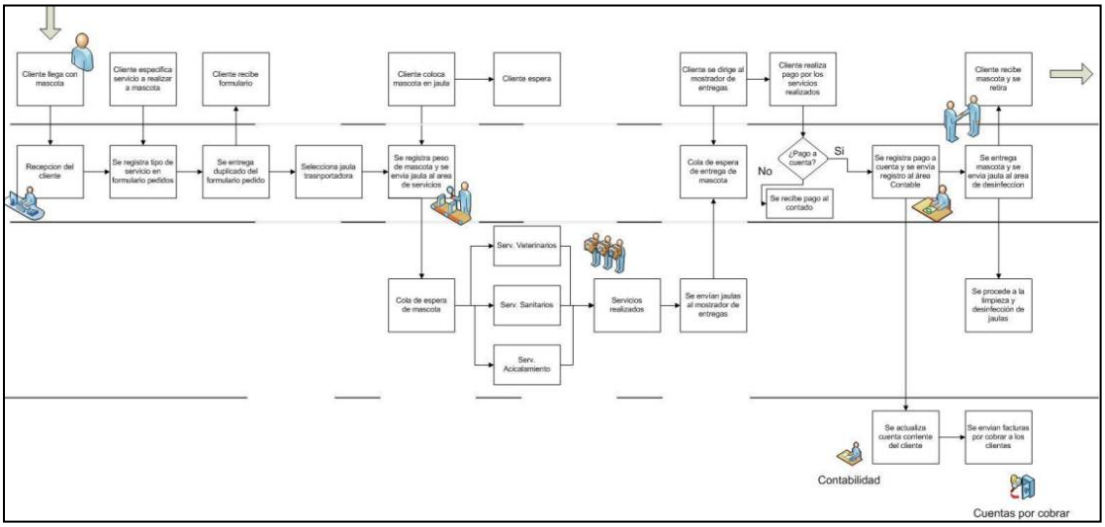


Fig. 2. Anteproyecto de servicio. Estudiantes Villalba, Matías – Lavayén, Bruno

Actividad 3: **Diseñar la cadena de valor**

A partir de tener visualizado el alcance del servicio de la veterinaria e identificado los momentos de contacto con el cliente, se diseña el modelo cadena de valor para determinar las actividades primarias y de apoyo que permitirán crear valor al satisfactor.

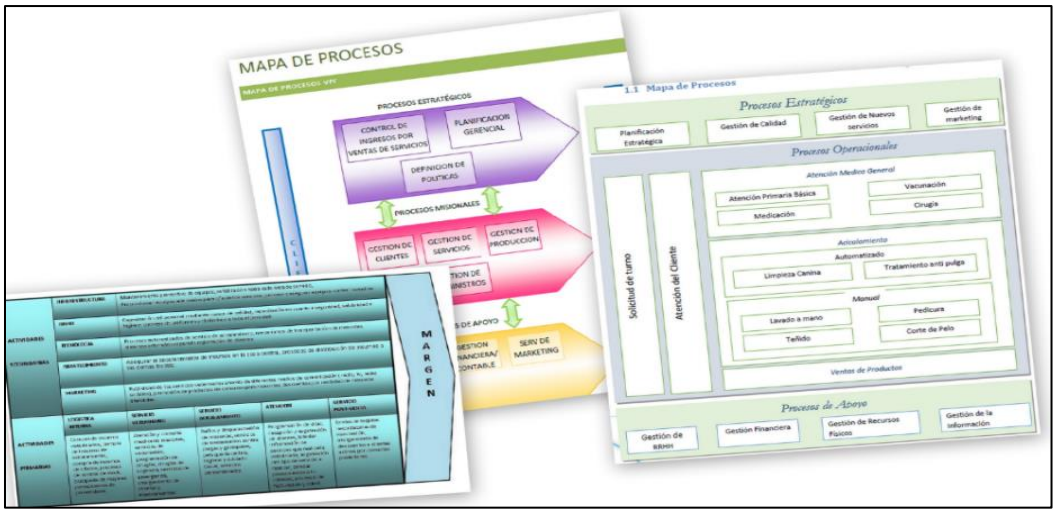


Fig. 3. Cadena de valor. Collage de imágenes de trabajos finales de alumnos.

<u>Actividad 4</u>: Proponer un modelo de creación de valor a partir de la incorporación de los sistemas de información administrativa

Finalmente el estudiante, a partir de la cadena de valor, el satisfactor identificado y los atributos valorados por el cliente, evalúa las necesidades y problemas de la veterinaria en materia de tratamiento de la información, con la finalidad de identificar los sistemas de información administrativa.

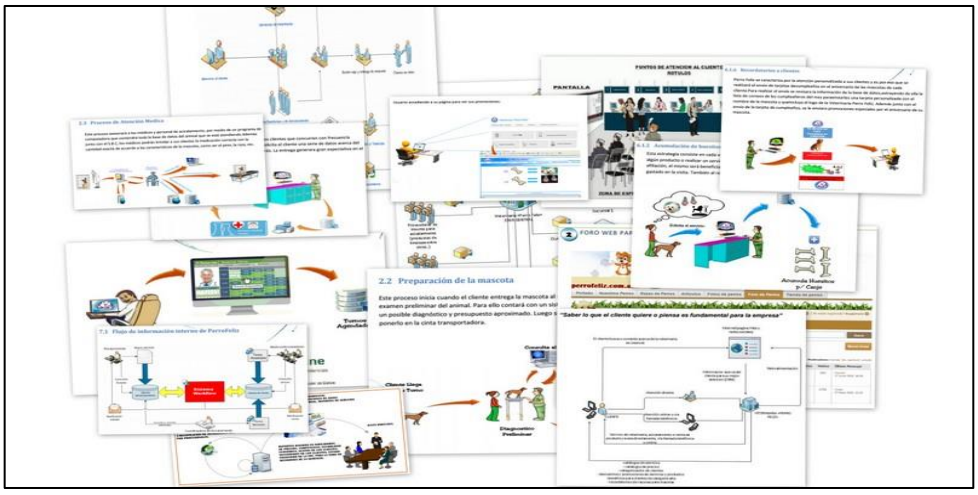


Fig. 4. Sistemas de Información Administrativa. Collage de imágenes de trabajos finales.

4 Resultados obtenidos

Los resultados que se presentan en esta sección, responden a la observación, análisis y evaluación del docente sobre los trabajos desarrollados por los estudiantes. Se pueden visualizar en los mismos, la identificación de diversos modelos de creación de valor desde la perspectiva tecnológica, incorporando a la organización estudiada un amplio abanico de soluciones sistémicas, como consecuencia de captar las dimensiones relativas a organización, administración y tecnología de información.

Los objetivos arribados, en términos de competencias logrados por los estudiantes, a partir de la implementación de la propuesta didáctica, se traduce en:

- Adquisición de una visión integral de la teoría y práctica de la administración y gestión organizacional.
- Adquisición de la habilidad para integrar las tecnologías de información a la administración de operaciones.
- Adquisición de los conocimientos para diseñar los sistemas de información administrativa que se ajusten a la cadena de valor de una empresa.
- Adquisición de la capacidad para lograr que el sistema de información administrativa se transforme en una ventaja competitiva para la empresa.

La propuesta didáctica también es evaluada por los estudiantes, a través de un cuestionario abierto y anónimo y consiste en una serie de preguntas cuyas respuestas son alternativas, siguiendo una escala "Totalmente, Parcialmente y Ninguno" y deja libertad para que quien conteste, se exprese a través de su propia redacción y lo haga según lo que considere oportuno manifestar [12]. El cuestionario fue contestado por todos los estudiantes que cursaron la materia Sistemas de Información Administrativa y los aspectos a evaluar fueron los siguientes:

- Expectativas y motivaciones.
- Aportes recibidos para la formación personal y profesional.
- Relación e integración de contenidos aprendidos en las asignaturas correlativas, ya cursadas y/o aprobadas de la carrera.
- Apoyo por parte del equipo docente, en términos de comunicación y diálogo.
- Seguimiento del cursado y la evaluación propuesta.

De la lectura de los cuestionarios, el estudiante destaca que:

- Conoció otras maneras de tratar la información e identificar otros tipos de sistemas de información.
- Tuvo la posibilidad de diseñar los sistemas de información teniendo en cuenta la necesidad organizacional de la empresa.
- Le permitió tener una visión general de las posibilidades de incorporación de sistemas para el tratamiento de la información.
- Fue escaso el tiempo de aprendizaje para el contenido planteado.

En términos generales se observa el interés creado en el estudiante por cumplir con el objetivo trazado por el docente y el fortalecimiento actitudinal observado durante el proceso de desarrollo del trabajo.

4 Conclusiones y líneas de trabajo

La enseñanza tradicional de los sistemas de información consiste en transmitir a los estudiantes conceptos de dato, información, sistemas, sistemas de información, clasificación de los sistemas de información, herramientas de modelado de sistemas, diseño de sistemas, para aplicar luego estos conceptos a casos de estudio, ya sea, reales o ideales.

Por la experiencia de la cátedra se sabe que al organizar así el contenido y la práctica, se obtienen propuestas de los estudiantes ajustadas y acotadas a solucionar el problema operativo, haciendo un uso adecuado de las herramientas aprendidas. Sin embargo, las propuestas de solución suelen ser pobres en relación a la esencia del negocio en sí, es decir, están lejos de constituirse en una verdadera ventaja competitiva, una herramienta que agregue valor al negocio para el cual se plantean.

Según Riballo Arenas, en la empresa, el objetivo primordial de los procesos informáticos y de las tecnologías de la información que los soportan es apoyar el negocio; mejorar la competitividad de la empresa. Dichos procesos cruzarán vertical y horizontalmente toda la estructura empresarial, han de ser claros y sencillos, y responderán a metas y objetivos concretos (enfatizarán el qué hacen por encima del cómo lo hacen) [7]

Conceptos como cadena de valor, procesos de negocios, valor agregado, marketing estratégico, no hacen a la columna vertebral de los sistemas de información, pero deben ser tenidos en cuenta en el momento de plantear una solución tecnológica para una organización. Si bien, en la práctica se podrá ir depurando la didáctica y mejorando aún más los resultados, se puede tomar la guía de actividades presentada como una muy buena práctica pedagógica.

Bibliografía

1. Gomez Vieites A. & C. Suarez Rey, *Sistemas de Información- Herramientas prácticas para la gestión empresarial*, Alfaomega, México, 2007.
2. Redunci, 2012. "*Áreas de conocimiento en las carreras de informática*". http://redunci.info.unlp.edu.ar/
3. Fuente de información de referencia en gerencia, marketing, finanzas, recursos humanos y economía, en español. http://www.gestiopolis.com/analisis-de-la-administracion-estrategica/ Consultado 01/08/2015.
4. Vicente, Miguel A.; Ayala, Juan C. (Coordinadores); y Beltrami, Ma.V.; Bonatti, Patricia; Bonelli, Mariano A.; Mendonca, Angel H.; Franco, Jorge A.; Fucaracce, Jorge R.; Gatto, Agustín; *Principios fundamentales para la administración de organizaciones*. Prentice Hall-Pearson Educación, 1ª. Ed. Bs. As., 2008.

5. Longenecker, Justin; Moore, Carlos; Petty, William; Palich, Leslie; *Administración de pequeñas empresas - Lanzamiento y crecimiento de iniciativas emprendedoras,* 14a. ed. CENGAGE Learning, 2008.
6. Wikipedia, la enciclopedia libre. https://es.wikipedia.org/wiki/Proceso_de_negocio Consultado 03/08/2015.
7. Riballo Arenas, Miguel Angel, *Implantación de aplicaciones informáticas de gestión*, Ed. RA-MA, Madrid, España, 2008.
8. Sitio web que publica artículos, guías y consejos sobre ideas de negocios, creación de negocios, marketing, ventas, finanzas y otros temas relacionados con los pequeños y medianos negocios. http://www.crecenegocios.com/la-cadena-de-valor-de-porter/ Consultado 03/08/2015.
9. Harldkoontz,.Heinz Weihrich. *Elementos de Administración*. McGraw Hill. 2007.
10. Francisco Gutiérrez Pérez, Daniel Prieto Castillo. *La mediación pedagógica. Apuntes para una educación a distancia alternativa*. Buenos Aires. Ediciones Ciccus. La Crujía. (1999)
11. Brown Andrew. *Gestión de la atención al cliente*. España. Ediciones Díaz de Santos. (1992)
12. Steiman, Jorge. "*Más Didáctica en la educación superior*". 1era edición. Buenos Aires. Unsam Edita. ISBN: 978-84-96571-80-8 (2008)

Simuladores Aplicados en Laboratorios de Energía Nuclear

Pérez Lucero Alejandra Lucía[1,2], Chautemps Norma Adriana[2], Bertone Rodolfo[3], Díaz Laura Cecilia[4]

[1] Maestría en Tecnología Informática Aplicada a la Educación, Universidad Nacional de la Plata
[2] Centro Universitario de Tecnología Nuclear (CUTeN), Universidad Nacional de Córdoba
[3] Facultad de Informática, Universidad Nacional de la Plata
[4] Facultad de Ciencias Exactas, Físicas y Naturales, Universidad Nacional de Córdoba
alejandraperezlucero@gmail.com

Resumen. En este trabajo se presentan avances en el marco del proyecto de tesis "Usos de simuladores virtuales para laboratorios de enseñanza en energía nuclear", que consisten en indagar acerca del grado de aceptación de una herramienta educativa para prácticas de laboratorio sobre física de las radiaciones. Mediante la incorporación de este simulador en los procesos de enseñanza, se persigue el objetivo de reducir los riesgos propios de la manipulación de fuentes radiactivas, antes de haber adquirido la destreza para evitar irradiaciones innecesarias. Se muestran resultados sobre un grupo de estudiantes del Centro Universitario de Tecnología Nuclear (CUTeN), de la Universidad Nacional de Córdoba, que cursan un posgrado en Metodología y Aplicación de los Radionucleídos.

Palabras clave: Aprendizaje, Aceptación, Usabilidad, Simulador, Energía Nuclear.

1 Introducción y Justificación

El interés en el uso del simulador se basa en la necesidad de entrenar a personas que tendrán una gran responsabilidad a su cargo, como es la manipulación de material radiactivo, ya que posibles errores pondrían en riesgo la funcionalidad de costosos instrumentos de medición.

La simulación ofrece una amplia variedad de datos que facilitan la verificación cualitativa y cuantitativa de las leyes científicas [1].

Con el simulador de laboratorio los practicantes se encuentran ante situaciones reales para las cuales la ocurrencia de errores no provoca daños sobre estos instrumentos sofisticados, delicados y escasos por su grado de especificidad. Se desarrollan habilidades y competencias procedimentales aprovechando las ventajas que ofrecen en la comprensión de los sistemas y su funcionamiento. Permite hacer comprobaciones o pruebas que no son factibles en un laboratorio real con lo cual resulta posible investigar conceptos teóricos dado que representan el funcionamiento y las operaciones reales enfocándose en la situación que recrean [2].

Otra motivación para el uso de simuladores, es el gran número de alumnos por curso que necesitan entrenamiento para manejar los instrumentos lo cual demanda tiempo por parte de los docentes. Los alumnos adquieren destrezas para operar las variables del experimento. Por otra parte, desde el punto de vista del docente la utilización de esta herramienta facilita la construcción del conocimiento con alto grado de autonomía y mayor comprensión de situaciones reales como así también el apoyo al aprendizaje de tipo experimental y conceptual en un ambiente realista [3]. Puede convertirse, además, en una valiosa herramienta de evaluación formativa y sumativa [4] al permitir realizar sucesivas pruebas y comparar con los resultados obtenidos. Finalmente se logra adquirir destreza en la protección radiológica dentro del laboratorio y profundizar el manejo de los instrumentos de detección de las radiaciones.

El CUTeN dicta cursos de grado, posgrado y divulgación sobre temas referidos a la energía nuclear. En todos los casos se debe trabajar con más de sesenta alumnos lo que dificulta las prácticas de laboratorio. Una situación favorable para la comprensión de los fenómenos físicos es disponer de varios instrumentos para distribuir en grupos.

Los inconvenientes que se presentan son las precauciones que se deben tomar para manipular material radiactivo, y el elevado número de radioisótopos que se necesita para distribuir en grupos numerosos. Esas fuentes de radiaciones son de alto costo monetario y decaen rápidamente en el tiempo por lo que se deben reponer anualmente. Por otra parte cualquier error que se cometa al operar los instrumentos de medición puede dañarlos o adulterar las mediciones. Para evitar esto se debe poner un instructor en cada grupo de trabajo. El uso de un simulador de laboratorio resuelve las situaciones descriptas dado que cada alumno puede disponer del mismo en su ordenador o dispositivo móvil, y desde allí realizar las prácticas correspondientes.

Aplicando este software el alumno podrá trabajar repitiendo las experiencias que necesite para comprender los fenómenos involucrados. Cada practicante podrá trabajar a su ritmo de aprendizaje evitando la presión de seguir al resto de la clase.

El simulador utilizado es el denominado “RadiationLab” disponible en internet. El software es libre y no es necesario registrarse ni contiene anuncios.

Está disponible en http://getwordwall.com/VisualSimulations. Este entorno web fue creado en 2005 por un programador de la empresa Visual Education Ltd con sede en Oxford, Reino Unido.

La decisión respeto del uso de dicha herramienta se basó en un ejercicio donde se tuvo en cuenta la opinión de los usuarios. A partir de experimentos realizados se tuvieron en cuenta como amigabilidad, facilidad de uso, aspectos metodológicos, resultados obtenidos e interpretaciones que se describen en los apartados siguientes.

2 Aspectos Metodológicos

En relación al desarrollo de la metodología planteada para implementar el uso del simulador se detalla a continuación la secuencia aplicada:

1. Se entrega al usuario un manual de uso de la herramienta el cual se diseñó teniendo en cuenta aspectos teóricos necesarios para la comprensión de los fenómenos físicos.
2. Se induce al trabajo de laboratorio con instrucciones sobre el manejo del instrumento y de los diferentes elementos disponibles para la realización de la práctica.
3. Se procesan los resultados obtenidos verificando los conceptos teóricos abordados y validando los mismos por comparación con lecturas obtenidas con instrumentos reales.
4. Se entrega la encuesta que consistió en nueve preguntas referidas a la accesibilidad, aspecto gráfico, calidad de las animaciones, relevancia del texto, originalidad.

Con respecto al uso del software, el usuario tiene la posibilidad de acceder al mismo desde los ordenadores disponibles en el aula donde se desarrolla el laboratorio dado que se baja desde la página indicada en párrafos anteriores, y se instala en cada computadora. La ventaja de este simulador de uso libre es que se puede utilizar en cualquier momento y retomar la medición que fuera interrumpida, dado que almacena la información en una planilla de cálculo dispuesta en el mismo software.

A continuación se muestra un modelo de la experiencia piloto realizada con este simulador, específicamente el trabajo práctico de laboratorio número 3, cuyo objetivo es observar la variación en la velocidad con la que se cuenta el número de interacciones entre las radiaciones con el detector, al interponer blindajes entre éste y la fuente radiactiva.

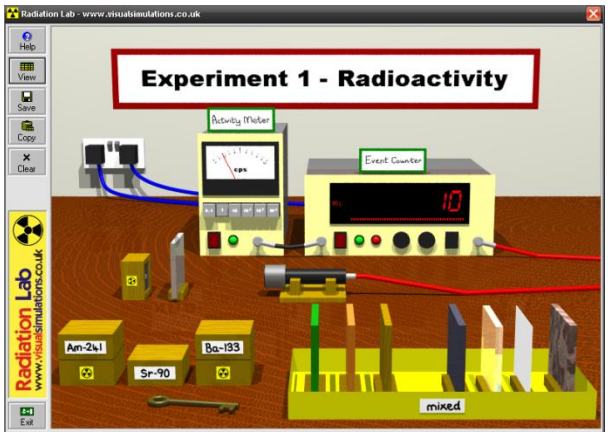


Ilustración 1: RadiationLab, el simulador

Fuente	Tipo Rad.	Material de Blindaje	Espesor de Blindaje (mm)	Distancia (cm)	Duracion de conteo (sec)	Tensión de alimentacion	Cuentas
Sr-90	β	nada	0	6	10	950	73346
Sr-90	*β*	*papel*	*0.2*	*6*	*10*	*950*	*68327*
Sr-90	β	aluminio	5.0	6	10	950	13
Am-242	α	nada	0	6	10	950	73577
Am-242	*α*	*papel*	*0.2*	*6*	*10*	*950*	*296*
Am-242	α	aluminio	5.0	6	10	950	10
Am-242	α	nada	0	12	10	950	11
Am-242	*α*	*papel*	*0.2*	*12*	*10*	*950*	*14*
Am-242	α	aluminio	5.0	12	10	950	13
Sr-90	β	nada	0	12	10	950	40900
Sr-90	*β*	*papel*	*0.2*	*12*	*10*	*950*	*34485*
Sr-90	β	aluminio	5.0	12	10	950	10

Ilustración 2: Tabla de resultados correspondientes al práctico de laboratorio número 3

En la tabla se observan los resultados obtenidos con el simulador para una experiencia determinada que consistió en interponer diferentes tipos de materiales para blindar radiaciones de distinta naturaleza y comportamiento. La intervención del alumno consistió en observar los resultados obtenidos para extraer las conclusiones correspondientes, que resultaron coincidentes con las esperadas.

A partir de esa experiencia, en una primera etapa se procedió a indagar en los alumnos y en los expertos del dominio alrededor de estos aspectos, cuestiones relativas a la facilidad de uso de la herramienta educativa.

En esta instancia se tuvieron en cuenta características relativas a la calidad de dicha herramienta seleccionando algunos aspectos significativos característicos de determinados atributos del software. Para ello se implementó un test de usabilidad típico (IEEE 1983) [5]:

- Facilidad de uso del producto
- Facilidad de aprender el producto
- Facilidad de hacer una determinada tarea
- Facilidad de instalar el producto
- Facilidad de encontrar información en el manual
- Facilidad de comprender la información
- Utilidad de los ejemplos de ayuda.

En una primera etapa se indaga a los expertos en cuanto al grado de aceptación y posteriormente se realiza una encuesta entre los alumnos del curso de Metodología y Aplicación de los Radionucleídos que organiza el CUTeN.

A continuación se muestra el resultado de la entrevista en profundidad que se hizo a los expertos sobre los puntos señalados:

- Facilidad de uso del producto: es fácil de usar, intuitivo en cuanto al manejo de los recursos disponibles tanto los instrumentos como los complementos: blindajes, fuentes radiactivas, tablas de resultados.
- Facilidad de aprender el producto: para aprender a usarlo es un proceso sencillo, lo aprenden en la misma clase. Se le dan indicaciones sobre el uso de la herramienta y lo que se espera obtener de ella.
- Facilidad de hacer una determinada tarea: la tarea se realiza en forma sencilla una vez que se tienen los conocimientos de lo que se busca, de lo que se puede hacer y lo que no. Por ejemplo no se fijan algunas variables para observar otras, y eso debe respetarse para alcanzar los resultados previstos.
- Facilidad de instalar el producto: se puede instalar fácilmente desde la página indicada. Eso permite que el alumno pueda repetir la experiencia o implementar otras. No necesita conexión a internet.
- Facilidad de encontrar información en el manual: el manual es muy didáctico pero muy básico, entonces le dimos información extra sobre la física de las radiaciones para elevarle el nivel de profundidad.
- Facilidad de comprender la información: la información tiene una explicación teórica y práctica con ejemplos lo que facilita el aprendizaje.
- Utilidad de los ejemplos de ayuda: los ejemplos son prácticos y fáciles de comprender.

Para los alumnos se elaboró una encuesta y se aplicó en el curso señalado. El mismo tuvo lugar entre abril y octubre de 2014 con una dedicación al laboratorio de cuatro horas para cada uno y un total de cinco experiencias diferentes.
Los resultados de calidad son los mostrados a continuación:

En la Tabla 1 que contiene la encuesta que se realizó se tuvo en cuenta la siguiente consideración para la evaluación de parte del usuario.

- Aporta instrucciones para el acceso y control de la información: al alumno se le hace entrega del manual del software, el cual le permite obtener información sobre los atributos de la herramienta e indicaciones de uso.

- Tipo de acceso al contenido: se pretende evaluar la facilidad con la que encuentra la manera de realizar el práctico de laboratorio utilizando los recursos disponibles: instrumento de detección, blindajes, fuentes emisoras de radiactividad, selector de voltaje, tiempo, distancia, y almacenamiento de los resultados.

- Calidad y relevancia de gráficos e imágenes: se indaga sobre los gráficos e imágenes que utiliza el software y que se espera faciliten la comprensión de los fenómenos físicos involucrados.

- Calidad de animaciones: evaluar este punto es de interés en cuanto al estímulo que puede generar en el usuario.

- Calidad y relevancia del sonido: el sonido es importante en este tipo de experiencias prácticas pues advierte sobre la presencia de radiaciones como medida de protección radiológica.

- Calidad y relevancia del texto: hace referencia al contenido teórico que explica el comportamiento físico de las radiaciones, como así también el uso del instrumento.

- Elementos innecesarios: se les presenta aquí la posibilidad realizar aportes o sugerencia en cuanto al aspecto visual y auditivo de la herramienta.

- Presenta originalidad y uso de tecnología avanzada: se indaga sobre éste aspecto relevante considerando la posibilidad de incorporar mejoras que complementen el uso del simulador.

Facultad de Ciencias Exactas Físicas y Naturales Curso de Metodología y Aplicación de los Radionucleidos Encuesta	
Aporta instrucciones para el acceso y control de la información:	Si: ____ No: ____
Tipo de acceso al Contenido:	Bueno ____ Regular ____ Malo ____
Calidad y relevancia de gráficos e imágenes:	Bueno ____ Regular ____ Malo ____
Calidad de las animaciones:	Bueno ____ Regular ____ Malo ____
Calidad y relevancia del sonido:	Bueno ____ Regular ____ Malo ____
Calidad y relevancia del texto:	Bueno ____ Regular ____ Malo _____
Sincronización imagen-sonido-texto:	Bueno ____ Regular ____ Malo _____
Elementos innecesarios: Si _____ No _____	
Si su respuesta es SI indique cuales: _____	
Presenta originalidad y uso de tecnología avanzada:	Si: _____ No: _____

Tabla 1: Resultados de la encuesta

3 Resultados

Luego de realizadas las prácticas de laboratorio, se dialoga con los expertos sobre el "RadiationLab", su uso, aplicación y accesibilidad para el alumno. La principal ventaja que encuentra el docente es la posibilidad de disponer de un software libre stand alone, es decir que no necesita una conexión a internet, facilitando de esta manera el acceso en cualquier tiempo y espacio. Está desarrollado para uso individual de allí la sugerencia por parte de los consultados, de analizar los resultados obtenidos

por parte de cada alumno y extraer conclusiones por comparación de las tabla de valore sobre cada experiencia. En cuanto a los resultados obtenidos en la aplicación de la herramienta virtual en un caso práctico, como es el Curso de Metodología y Aplicación de los Radionucleídos que se dicta a profesionales de la salud e investigación que trabajan con radiaciones ionizantes, los mismos fueron satisfactorios y se describe a continuación los resultados de la encuesta.

1. Se evalúa la aceptación de la herramienta a partir de los siguientes puntos:
    - Tipo de acceso al contenido
    - Calidad y relevancia de gráficos e imágenes
    - Calidad de las animaciones
    - Calidad y relevancia del sonido
    - Calidad y relevancia del texto

    Los resultados en este punto fueron los siguientes:

	BUENO	REGULAR	MALO
Tipo de acceso al Contenido	19	0	0
Calidad y relevancia de gráficos e imágenes	17	2	0
Calidad de las animaciones	16	4	0
Calidad y relevancia del sonido	17	1	1
Calidad y relevancia del texto	18	1	0

Tabla 2: resultados de la aceptación

También realizamos la gráfica comparativa sobre las respuestas a la encuesta en la ilustración 3.

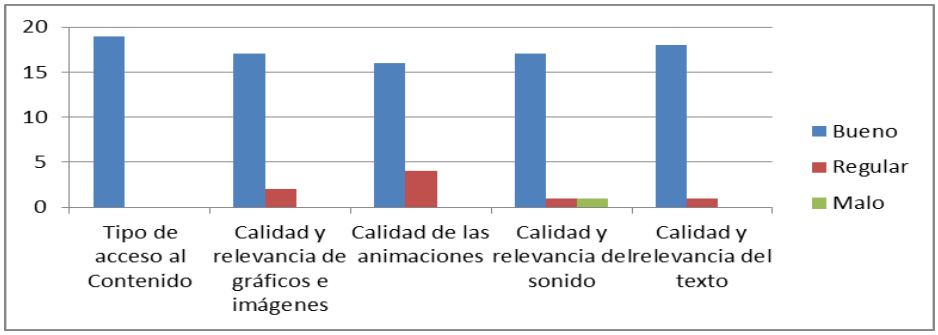


Ilustración 3: gráfica comparativa sobre las respuestas a la encuesta

Como se observa en la encuesta es alto el grado de aceptación a la herramienta siendo este un importante avance para continuar con la implementación de la misma. Es un estímulo para los usuarios que el primer contacto con la herramienta se realice en forma satisfactoria.

2. Se indaga sobre el acceso y control de la información de 19 alumnos. Los resultados son los siguientes:

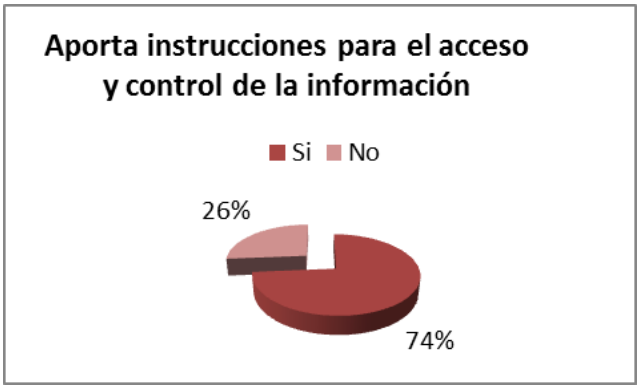


Ilustración 4: Gráfica sobre acceso y control de la información

Si bien se revela favorable el porcentaje de satisfacción en cuanto al aporte de instrucciones en el simulador, cabe mencionar que no fue necesario acudir en su totalidad a la información provista dado que la clase fue presencial y el docente instruyó sobre los contenidos necesarios para avanzar en la realización del práctico.

3. Se consulta sobre elementos innecesarios encontrados como forma de evaluar la herramienta de los 19 alumnos y los resultados fueron los siguientes:

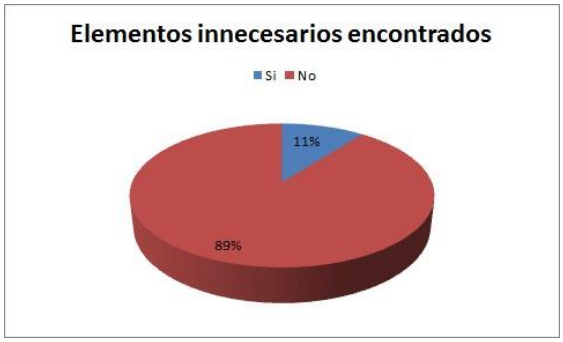


Ilustración 5: Gráfica sobre elementos innecesarios

Se observa un buen diseño de la herramienta en cuanto a que dispone solo de lo necesario para su funcionamiento y comprensión de los temas teóricos, como se refleja en la encuesta.

4. En cuanto a la originalidad del software y el uso de tecnología de avanzada de los 19 alumnos, las respuestas fueron:

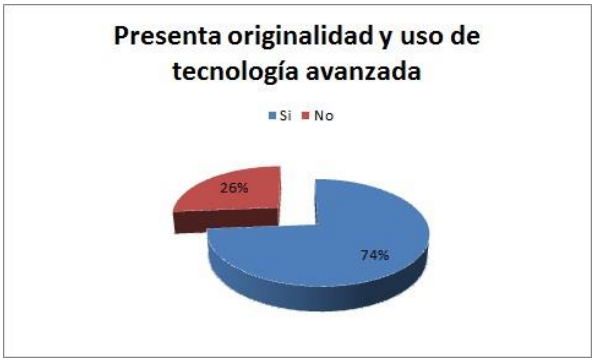


Ilustración 6: Gráfico sobre originalidad

Los usuarios manifestaron como original a la herramienta y de avanzada tecnología lo que resulta alentador para avanzar en la aplicación de la misma como parte de la formación en temas referidos al uso y aplicación de las radiaciones ionizantes.

4 Conclusiones

En las encuestas se puede observar que los expertos y los estudiantes consideran bueno el acceso al contenido, al igual que la calidad de los gráficos, animaciones y sonido. Los resultados en cuanto a la originalidad, acceso a la información y elementos innecesarios encontrados permite concluir que se continuará con el uso de la herramienta educativa en sucesivos cursos donde sean necesarios aplicar conocimientos prácticos sobre efecto de la interacción de la radiación con la materia.

La experiencia con el simulador ha permitido ver el comportamiento de los alumnos con respecto en la práctica de teoría – laboratorios, se observó la aceptabilidad en alto grado del mismo lo que estimula a continuar con el uso de la herramienta. Dado que el simulador es un software libre que se instala en cada computadora, permitió el acceso en todo momento para trabajar y repetir la experiencia cuántas veces fuera necesario. Estas consideraciones serán tenidas en cuenta en futuras prácticas de laboratorio. Las mejoras impulsadas por los docentes se refieren a la importancia de trabajar con los alumnos analizando los resultados obtenidos y verificando los fenómenos físicos esperados a partir de lo estudiado en la teoría.

De esta manera se considera que la herramienta es útil con un instructor que estimule la investigación y exploración de la herramienta, sus potencialidades incluso más allá de lo que ofrece el manual de instrucciones. Tal es el caso de la variación de la tensión aplicada al instrumento que es una variable que no se trabaja desde el manual propio del simulador. En etapas sucesivas se irán incorporando nuevas experiencias que actualmente se desarrollan con instrumentos reales y que a vista de los expertos podrán aplicarse con esta herramienta educativa.

5. Referencias

1. Sierra, J. (2000) "Informática y Enseñanza de las Ciencias". En Perales F.J. y Cañal P. (Eds.) Didáctica de las Ciencias Experimentales (339-359).Alcoy: Marfil.
2. Aldrich, C. (2005) Learning by doing: a comprehensive guide to simulations, computer games, and pedagogy in e- Learning and other educational experiences. San Francisco: Pfeiffer.
3. Guaralnick, D., y Levy, C. (2009). Putting the Education into Educational Simulations: Pedagogical Structures, Guidance and Feedback. International Journal of Advanced Corporate Learning, 2(1), 10-15.
4. Bradley, P. (2006). The history of simulation in medical education and possible future directions. Medical Education, 40(3), 254-262.
5. http://www.redalyc.org/articulo.oa?id=133114988005: Modelo de Evaluación de Calidad de Software Basado en Lógica Difusa, Aplicada a Métricas de Usabilidad de Acuerdo con la Norma ISO/IEC 9126. Gustavo Alberto Ruiz, Alejandro Peña, Carlos Arturo Castro, Angela Alaguna, Luz Mery Areiza y Rafael David Rincón, 2006.

Producción de Objetos de Aprendizaje para la enseñanza Universitaria. Convocatoria a los docentes de la Facultad de Informática de la UNLP

Armando De Giusti[1], Alejandra Zangara[1], Cecilia Sanz[1], Lucrecia Moralejo[1], Fernanda Barranquero[2], Marcelo Naiouf [1]

degiusti@lidi.info.unlp.edu.ar, Alejandra.zangara@gmail.com; csanz@lidi.info.unlp.edu.ar, lucreciamoralejo@gmail.com, mfernandabarranquero@hotmail.com, mnaiouf@lidi.info.unlp.edu.ar

[1] Instituto de Investigación en Informática LIDI (III LIDI). [2] Dirección Pedagógica Facultad de Informática, Universidad Nacional de La Plata
La Plata, Argentina

Resumen: La producción de Objetos de Aprendizaje (OA) para la enseñanza en la Universidad es un tema que despierta tanto interés como controversia. Resultan dilemáticos desde la propia definición de Objeto de Aprendizaje, pasando por las metodologías propias para su diseño y desarrollo y el tema de la reutilización, incluyendo el valor del docente como reutilizador y a la vez resignificador del contenido y formato del OA original. Lo cierto, en un campo de tantas tensiones, es que la decisión institucional de abordar este tipo de proyectos y la capacitación docente, pasan a ser factores clave. Se necesita, entonces, comenzar por una definición prolija y exhaustiva del concepto de Objeto de Aprendizaje, tanto como objeto didáctico cuanto tecnológico. Luego, es indispensable pensar en las competencias que hacen que un docente pueda diseñar, desarrollar, utilizar y reutilizar OAs. Estos temas deben comenzar, necesariamente, por una perspectiva institucional. Las instituciones deben estar a la cabeza de estos proyectos de innovación, brindando a los docentes las condiciones y estímulo necesario para iniciar este camino. Esta idea dio inicio a la "Convocatoria para la Producción de Objetos de Aprendizaje para innovar la enseñanza en la Facultad" que se realizó en la Facultad de Informática de la UNLP en Agosto de 2014, y que fue encarada en conjunto desde la Dirección de Educación a Distancia y Tecnologías aplicadas en Educación, la Dirección Pedagógica y la Maestría en Tecnología Informática aplicada en Educación de la Facultad. Este trabajo da cuenta de los pasos de la convocatoria, la capacitación y los resultados del primer año.

Keywords: Objetos de Aprendizaje, Producción de Objetos de Aprendizaje, Innovación, Reutilización de Objetos de Aprendizaje.

1 Introducción

Hace ya unos cuantos años que los entornos virtuales de enseñanza y aprendizaje (EVEA) forman parte de numerosos escenarios educativos, especialmente en el ámbito universitario. Los docentes y los alumnos se han familiarizado con su utilización, y se han visto involucrados en el trabajo con materiales educativos digitales, que se convirtieron en un pilar fundamental, a la hora de abordar propuestas mediadas por Tecnologías de la Información y la Comunicación (TIC). Así, los docentes también se involucraron en la producción de sus propios materiales o recursos educativos digitales (presentaciones, sitios web, videos, imágenes), lo que abrió el debate en vinculación con su reutilización, en diferentes contextos educativos y tecnológicos.

Los Objetos de Aprendizaje (OA) proponen un camino para el diseño de materiales educativos digitales reutilizables.

Si bien algunos autores denotan su origen al área de las Ciencias de la Computación, y lo vinculan con la Programación Orientada a Objetos por sus características de reutilización, otros indican que el término fue acuñado por Wayne Hodgins quien, en 1992, propuso construir material educativo digital a partir de módulos independientes y reutilizables que se podían acoplar para crear módulos cada vez más complejos, de forma semejante a cómo se crean construcciones con los bloques Lego. Hodgins al observar a su hijo jugando con unos Lego, se dio cuenta de que los bloques de construcción que usaba podrían servir de metáfora explicativa para la construcción de materiales formativos; pensaba en pequeños bloques formativos que permitieran el aprendizaje de una forma sencilla y que pudieran conectarse fácilmente entre sí para crear estructuras o productos más complejos o de mayor alcance [1].

A finales de los años noventa James L´Allier define un OA como "la experiencia de formación independiente más pequeña que contiene un objetivo, actividades de aprendizaje y una evaluación" [2].

Para Polsani (2003) un OA es: "una unidad autónoma e independiente de contenido de aprendizaje que está predispuesta al reuso en múltiples contextos de enseñanza" [3].

Existen numerosas definiciones sobre el concepto de Objeto de Aprendizaje, que han dado lugar a la generación de diferentes materiales educativos, con distintos niveles de granularidad. Lo cierto es que los objetos de aprendizaje están ligados a un diseño para la reutilización, la interoperabilidad, el acceso (desde diferentes repositorios), la durabilidad, por mencionar algunas de las características en las que los autores tienen mayor coincidencia. También, no hay duda que los objetos de aprendizaje se encuentran dentro de la categoría de materiales educativos, y que por tanto tienen una finalidad educativa. La granularidad se ha vuelto uno de los temas más controversiales en relación a los OA. En consecuencia, se pueden encontrar materiales con los más diversos niveles de granularidad, que para algunos resultan ser OA y para otros no. En este artículo se parte de una definición particular, abordada en el marco de la Maestría en Tecnología Informática Aplicada en Educación de la Facultad de Informática, y que atiende a las principales características en las que acuerdan los autores de referencia en el tema, pero que al mismo tiempo diferencia al OA de otros materiales/recursos educativos. Así un OA es: "un tipo de material educativo digital, que se caracteriza, desde el punto de vista pedagógico, por orientarse a un objetivo específico de aprendizaje, y por presentar: una serie de contenidos con el fin de abordar la temática relacionada con el objetivo, actividades que permitan al alumno poner en práctica o problematizar el contenido presentado, y una autoevaluación que posibilite conocer al alumno, si ha podido comprender esos contenidos vinculados al objetivo. Desde el punto de vista tecnológico, se caracteriza por contener un conjunto de metadatos estandarizados para su búsqueda, y recuperación, y estar integrado, utilizando un modelo de empaquetamiento que respete estándares, y de esta manera, permita su diálogo con diferentes entornos tecnológicos" [4].

De esta manera, se diferencia el concepto de OA de otros tipos de materiales educativos, en este caso se definen sus componentes: un objetivo específico que identifique su propósito educativo, una serie de contenidos, actividades y una autoevaluación que se interrelacionen, de manera tal que permitan la consecución del objetivo que se propone. También se delimita su granularidad al indicar que se orienta a un objetivo específico. Se especifica que deben tener metadatos que sigan un estándar para lograr su almacenamiento, búsqueda y recuperación, y que debe ser empaquetado con algún modelo de empaquetamiento de manera tal que respete la característica de interoperabilidad.

Un objeto de aprendizaje entonces es una unidad autónoma que permite alcanzar un determinado objetivo educativo y que combinada con otros Objetos de aprendizaje puede dar lugar a la construcción de estructuras más complejas tales como lecciones, módulos o cursos.

A partir de esta definición, se ha abordado también la creación de una metodología de diseño de OA, llamada CROA [5], que guía al docente (o al equipo de desarrollo), en el diseño y creación de un OA, respetando la definición dada. Esta metodología se ha desarrollado luego de un profundo estudio de varias metodologías orientadas a la creación de OA [6]. CROA realiza un aporte dando guías tanto desde el punto de vista del diseño instruccional como del diseño tecnológico de un OA. Combina ambas miradas y busca orientar el trabajo de un docente que puede no tener conocimientos específicos en el área de programación.

Este artículo presenta la experiencia desarrollada en el marco de la Facultad de Informática de la Universidad Nacional de La Plata, orientada a la propuesta para incentivar la producción de OA en el marco de esta institución y a la creación de un repositorio específico. Para ello en la sección 2 se presenta el proyecto con sus diferentes etapas, en la sección 3 se presentan algunos resultados preliminares, y en la sección 4, se detallan algunas conclusiones y trabajos futuros.

2 Proyecto para la producción de Objetos de Aprendizaje en la Facultad de Informática

En consonancia con el proyecto de la Facultad, de marcar camino en la interacción de los mundos educativo y tecnológico -en este caso en el diseño de objetos de aprendizaje-, se diseñó y llevó adelante un proyecto de trabajo que hizo foco en este tema. Este proyecto se lanzó en conjunto desde las áreas de gestión y formación que atienden estos temas. Intervinieron la Dirección Pedagógica, la Dirección de Educación a Distancia y Tecnología Informática aplicada en Educación y la Maestría en "Tecnología Informática aplicada en Educación". Como parte del Proyecto se definieron una serie de etapas, tal como se presentan en la figura

siguiente. Estas son: Instalación del tema (chala de sensibilización); Lanzamiento de la Convocatoria; Capacitación y Seguimiento de las necesidades de los docentes; Presentación de documentos entregables, y Publicación de OA en un repositorio institucional. En las sub-secciones siguientes se detallará la planificación de cada una de las etapas mencionadas.

Figura 1:
Etapas del proyecto producción de Objetos de Aprendizaje en la Facultad de Informática

2.1. Instalación del tema – Charla de sensibilización

Esta charla se orientó a presentar las bases del proyecto. Se hizo foco en la definición de OA, las metodologías posibles para su diseño y la resignificación del rol docente que podría facilitar su uso y reutilización. Fue organizada entre las dos Direcciones mencionadas y la Maestría. El título de la charla inicial, desarrollada en Septiembre de 2014, fue: "*Concepto, diseño y producción de Objetos de Aprendizaje*". Se trabajó con los siguientes objetivos:

- Presentar el concepto de Objetos de Aprendizaje (OA) como un concepto dilemático, luego se discutió la definición presentada en la primera sección de este artículo. Se debatió también sobre la utilidad de los OA en la enseñanza de grado y postgrado en diversas áreas de contenido.
- Mostrar diferentes repositorios de OA.
- Presentar una convocatoria de la Facultad, a través de la Dirección de Educación a Distancia, para los docentes de grado y postgrado para el diseño de OA en sus cátedras.

Cabe aclarar que la presentación de cuestiones centrales de la gestión de tecnología en la educación como trabajo en conjunto de las Direcciones que trabajan en estos temas es una metodología corriente en la Facultad. Permite generar, a la vez, la presentación de diversos aspectos por especialistas y la instalación de un diálogo/debate entre docentes que muchas veces se continúa luego de la finalización de la charla presencial. Por eso, éste fue el camino elegido para comenzar a andar en este tema.

2.2. Convocatoria

La información de la convocatoria resultó uno de los temas centrales de este proyecto. El título de la misma fue: "*Producción de objetos de aprendizaje para innovar la enseñanza en la Facultad*". Esto daba una clara idea de su objetivo (la producción de materiales educativos bajo el formato de OA), la idea de posicionamiento e innovación en la enseñanza y los destinatarios (en una primera instancia, sólo docentes de la Facultad).

Desde el inicio de la convocatoria, y para continuar la línea de investigación y conceptualización que se presentó en el apartado inicial, se incluyó el concepto de OA. Además, se informó a los docentes la siguiente intención: "*la Facultad quiere tomar la iniciativa y recopilar los materiales, que respondan al diseño de objetos de aprendizaje, con el objetivo de innovar en las prácticas de enseñanza y generar un repositorio de materiales a nivel institucional*". Eso ya anticipaba una de las principales ideas que conlleva este proyecto: la creación de un repositorio específico de OA institucional.

En la convocatoria también se invitó a los docentes a participar de una capacitación, acorde a sus necesidades específicas. Esto se detallará más acabadamente en la próxima sub-sección.

Respecto de la metodología de diseño de OA, tema de tanta relevancia como controversia, se tomó la decisión de dar a los docentes la libertad en la elección de la metodología. Cabe aclarar que algunos docentes de la Facultad ya venían trabajando en este tema antes de la convocatoria, con metodología propia. La forma de unificación que se decidió seguir fue pedir que los OA respondan a una misma definición y se acordaron además, una serie de preguntas-guía-entregables orientadas a que los docentes realicen una planificación básica del OA, más allá de la metodología utilizada.

En la convocatoria se hizo referencia, además, a la conformación de un jurado para la evaluación de los OA a crear, de manera de "premiar" de alguna manera a aquellos que resultaran mejor calificados por los jurados, y lograr así un incentivo adicional para los docentes. Se decidió que el Jurado estuviera conformado por reconocidos expertos en el tema de Universidades Nacionales y extranjeras. Al mismo tiempo en la convocatoria se realizaron las siguientes aclaraciones:

- El OA debe estar referido a alguno de los contenidos de las materias que se dictan en la Facultad y en la que el autor del OA sea docente.

- No es necesario que el OA haya sido usado en la enseñanza del tema seleccionado ni se haya evaluado su uso, aún.

Uno de los puntos básicos de la convocatoria, atendiendo a la orientación que se intentó proporcionar a los docentes desde el inicio, fue el cronograma de trabajo con los dos documentos "entregables" que se requerían para organizar el circuito de producción. Se hará mención específica de este tema en los apartados siguientes.

2.3. Capacitación

La capacitación es considerada uno de los temas centrales del proyecto, ya que representó un desafío por la complejidad del tema en sí y por la heterogeneidad de docentes que se acercaron, interesados por la temática durante las etapas anteriores. Se trabajó, entonces, por módulos interdependientes de capacitación, acorde a las necesidades de los diferentes docentes. Se propusieron los siguientes módulos:

MÓDULO 1: Concepto y metodología de producción de OAs. Se abordó específicamente la metodología CROA diseñada en el marco de un proyecto de investigación de la Facultad. En el marco de la metodología se discutieron los pasos para el análisis, diseño, implementación, y evaluación del OA. Se presentaron algunos estándares de metadatos y se sugirió trabajar con LOM. Al mismo tiempo, se analizar diferentes modelos de empaquetamiento.

MÓDULO 2: Diseño instruccional de OAs. Desde la Dirección Pedagógica se abordó una capacitación específica para el diseño instruccional alineada con la metodología CROA. Se entregó documentación específica.

MÓDULO 3: Herramientas de desarrollo de OAs y de Edición de Metadatos. Se abordó el uso de diferentes herramientas para crear el OA (Ardora, ExeLearning, y la integración de ambas, Reload), también se presentaron herramientas para la edición de metadatos (ExeLearning Reload), y se realizaron recomendaciones, siguiendo a CROA, para completar dicho metadatos.

Además, en el entorno virtual de enseñanza y aprendizaje WebUNLP, que es utilizado en varias cátedras de la Facultad, se habilitó un curso en el que se incluyeron los materiales y se respondieron consultas (con el sistema de tutorías virtuales). Al mismo tiempo, se brindó información sobre el desarrollo de la convocatoria.
Desde la Dirección de Educación a Distancia y Tecnología Aplicada en Educación se ofreció acompañamiento en el diseño y producción de los OA, mediante tutorías online (vía WebUNLP y eventualmente el mail de la Dirección) y, eventualmente, encuentros presenciales en la Facultad.

2.4. Entregables y su seguimiento

Una vez finalizada la capacitación, la siguiente etapa propuesta es la de "Entregables y seguimiento del trabajo de los docentes". Esta etapa es de fundamental importancia el seguimiento de trabajo de los docentes, en dos sentidos:
- las tutorías (presenciales o a través del entorno virtual) y
- la organización de los documentos de entrega intermedios (que llamamos "entregables") que los docentes deben ir presentando para avanzar en sus trabajos de diseño y producción de los OAs.

El primer entregable solicitado consiste en presentar información general del OA y el brief de diseño. El segundo entregable consiste en la implementación del OA y la actualización pertinente del primer entregable, acorde a los avances que se hayan realizado.
Para el primer entregable la información a explicitar es la siguiente:

Datos generales:
-Título del Objeto de Aprendizaje
-Autor/es (incluye el docente/s autor y el docente que avala la producción)
-Cátedra/materia para la que se realiza el OA

Planificación del OA
-¿Qué es lo que los destinatarios necesitan aprender?
-¿A qué nivel educativo se orienta?
-¿Cuál es el tema que el OA abordará?
-¿Qué conocimientos previos debería tener el alumno para utilizar el OA?
-¿Con qué otros conocimientos se relaciona el OA, que pueden ser adquiridos en forma posterior a trabajar con lo que se está diseñando?
-¿Qué objetivo de aprendizaje específico se propone para el OA?
-Mapa de navegación del OA (puede usarse un gráfico)
-Actividades que se le propondrán al alumno en vinculación con el objetivo propuesto
-Autoevaluación que se le propondrá al alumno en vinculación con el objetivo propuesto

Los responsables de las dos Direcciones que organizaron el proyecto y de la Maestría en TIAE[1] constituyeron una Comisión Evaluadora, para responder a cada entregable de los docentes o grupos de docentes inscriptos. Además, se ofrecieron tutorías presenciales para aclarar/profundizar los comentarios.

2.4. Repositorio

En paralelo con el diseño e implementación de la convocatoria y en relación con la producción de los OA, se comenzó a trabajar conjuntamente con el SEDICI (Repositorio Institucional de la Universidad Nacional de La Plata) en la creación de un repositorio específico de Objetos de Aprendizaje vinculado a la Facultad de Informática.

En la etapa de publicación de los OA, se busca iniciar el proceso de disponibilidad de estos materiales educativos, para que puedan ser utilizados por docentes y alumnos. Esto constituye el fin último del proyecto, que se resultará luego en una convocatoria permanente para continuar nutriendo el repositorio.

3. Resultados preliminares

Se realizaron algunos análisis a nivel de cada etapa ya implementada del proyecto. De momento, se está atravesando la etapa de evaluación de los entregables para alcanzar el momento de su publicación en el repositorio.

---

[1] *Maestría en "Tecnología Informática aplicada en Educación": http://postgrado.info.unlp.edu.ar/Carreras/Magisters/Tecnologia_Informatica_Aplicada_en_Educacion/Tecnologia_Informatica_Aplicada_en_Educacion.html*

A continuación se presentan algunos resultados preliminares del recorrido del proyecto ya implementado:

De la **charla inicial** participaron 18 docentes, de grado y postgrado. En referencia a las áreas de procedencia de los docentes, resultó interesante que sólo el 56% pertenecía a materias de informática, asociadas al desarrollo de software. Hubo varios docentes "no informáticos" y esto resultó interesante, porque daba la idea cabal de que el proyecto comenzaba a funcionar, ya que docentes de diferentes áreas se sentían interesados. En la figura 2 se presenta un análisis, en cuanto al tipo de materia que dictan, de los asistentes a la primera charla de instalación del tema en la Facultad (Septiembre 2014).

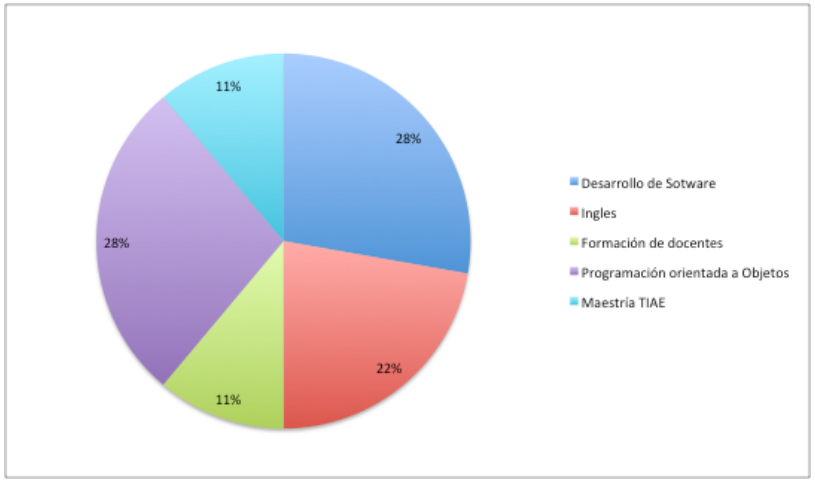


Figura 2:
Asistentes a la charla de: "Concepto, diseño y producción de Objetos de Aprendizaje"

En la **convocatoria** para la inscripción al desarrollo de OA a participar en la premiación, se anotaron 10 docentes o grupos de docentes, con diferentes temáticas a trabajar en los OA que proponían. En la figura 3, se muestra los temas propuestos.

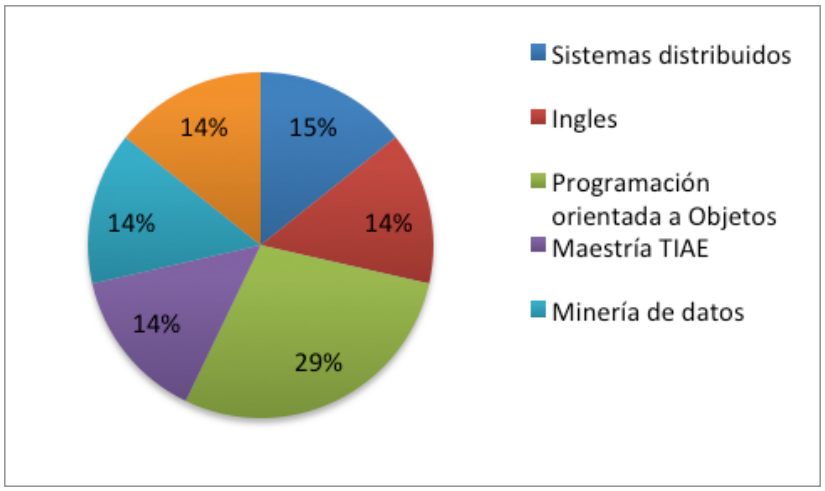


Figura 3:
Inscripción a la convocatoria de OAs y temas

En la **etapa de capacitación**, diferentes docentes hicieron aprovechamiento de los distintos módulos. Los inscriptos en la convocatoria participaron de aquellos módulos que cada uno sentía que más necesitaba. Algunos siguieron todos los módulos, mientras que otros, tomaban uno de los módulos en particular. Incluso, participaron docentes que no se anotaron en la convocatoria, pero que estaban interesados en el tema.

En la **etapa de entregables**, todos los inscriptos presentaron su documentación de planificación del OA, y actualmente, la mayoría ha entregado el OA implementado, salvo algún el caso de un docente que solicitó más tiempo.

Otro aspecto a resaltar como resultado es la importancia de las **tutorías intermedias entre los entregables**, que fueron de gran ayuda para los docentes, tanto para reforzar temas ofrecidos en la capacitación como para trabajar esos temas en el entramado con los objetivos y contenidos propios de cada OA planificado. En las tutorías que se desarrollaron *a posteriori* de la devolución del primer entregable se trabajaron tres temas importantes, que se relacionan con las cuestiones centrales que la literatura presenta, también, en relación con los OAs: Diseño instruccional del OA, Herramientas de desarrollo y Metadatos.

En cuanto al **diseño instruccional**, uno de los comentarios más frecuentes estuvo vinculado con la consistencia, en el tránsito entre el objetivo-contenidos-actividades de práctica y autoevaluación. En algunos casos, la devolución del primer entregable y la posterior tutoría dio como resultado una reorganización completa del OA, atendiendo a cuestiones de consistencia didáctica y de granularidad del OA. Por ejemplo, uno de los equipos de docentes terminó diseñando una cadena de 5 OAs, partiendo de un objetivo inicial-más general- que se descompuso en objetivos específicos de menor nivel (y, consecuentemente, OAs asociados).

En este momento se está trabajando en la evaluación de los OAs que fueron entregados por los docentes como respuesta a la primera convocatoria.

4. Conclusiones y Trabajos Futuros

Si bien existe satisfacción por estos resultados preliminares, el camino recién comienza y se multiplica. Algunos aspectos remarcables para la Facultad son los siguientes:

- Se ha logrado instalar el tema de objetos de aprendizaje como un tipo de material educativo particular entre los docentes de grado y postgrado de la Facultad.
- La convocatoria fue bien recibida por los docentes, considerando su variedad en temas y formación de base.
- Los docentes mostraron fuerza de producción, aún en el trabajo interdisciplinario: gráfico, instruccional, de contenidos, de herramientas.
- Existe interés por parte de los docentes por comenzar a documentar (en forma de *papers* o informes) sus avances tanto en la producción como en el uso de OAs con pares docentes y alumnos.

- La Facultad está en camino de tener su propio repositorio en el área del SEDICI, lo que será una innovación a nivel institucional.

Por otra parte, las próximas acciones en el diseño y producción de OAs en la Facultad serán:

- Generar una nueva convocatoria, que sea permanente, con la idea de que haya producción de OA en forma continuada y los docentes tengan asistencia en este sentido.
- Fortalecer la producción de OAs que se puedan ensamblar para algunas materias.
- Continuar con la implementación de nuevas funcionalidades para el respositorio.
- Avanzar en la evaluación del uso de OAs.
- En este sentido, generar un diseño ad hoc para fortalecer la investigación sobre el tema de la reutilización, lo que sería de gran aporte ya que representa un área de vacancia e interés en estas temáticas.

5. Bibliografía

[1] Hodgins, H. W. (2000). The future of learning objects. The Instructional Use of Learning Objects: Online Version. Recuperado a partir de http:// reusability.org/read/chapters/hodgins.doc

[2] L'Allier, J. (1998). NETg's Precision Skilling: The linking of occupational skills descriptors to training interventions. Recuperado a partir dehttp://www.netg.com/research/pskillpaper.htm

[3] Polsani, P. R. (2003). Use and Abuse of Reusable Learning Objects. Journal of Digital Information, 3(4). Recuperado a partir de journals.tdl.org/jodi/article/viewArticle/89

[4] Sanz C., Moralejo L., Barranquero F. (2014). Materiales del Curso de Doctorado: "Diseño y Producción de Objetos de Aprendizaje".

[5] Sanz C., Moralejo L., Barranquero F. (2014). Metodología CROA. http://croa.info.unlp.edu.ar

[6] Maldonado J., Sanz C., Fernandez Pampillón A.M. (2014). Desarrollo de un marco de análisis para la selección de Metodologías de Diseño de Objetos de Aprendizaje (OA) basado en criterios de calidad para contextos educativos específicos". Tesis de la Maestría de Tecnología Informática Aplicada en Educación. Marzo de 2015. http://sedici.unlp.edu.ar/handle/10915/45063

Una Comparativa del Algoritmo Particle Swarm Optimization con Restricciones entre los Métodos de Factor de Constricción y Neighborhood con Coeficiente de Inercia Dinámico

Miguel Augusto Azar, Fabiola Patricia Paz, Analía Herrera Cognetta

Facultad de Ingeniería - Universidad Nacional de Jujuy
auazar@live.com, fabyppaz@gmail.com, anihco@yahoo.com.ar

Resumen. Los diferentes enfoques de optimización por enjambre de partículas ofrecen variantes que ayudan a mejorar la performance del algoritmo en una serie de aspectos tales como robustez, calidad, confiabilidad de resultados en óptimos globales, entre otros. La hibridación entre algunas de estas variantes no necesariamente garantiza un resultado exitoso en cuanto a los aspectos mencionados. El presente trabajo se concentra en estudiar los problemas de optimización con restricciones mediante la comparación entre el clásico algoritmo PSO con factor de constricción y un híbrido con topología de vecindad y factor de inercia dinámico.

Palabras Clave. Optimización, metaheurísticas, constriction, inteligencia de enjambre, particle swarm optimization, PSO, neighborhood, factor de inercia dinámico, constraints.

1 Introducción

La resolución de problemas de optimización en los cuales se involucra una función objetivo dentro de un espacio multidimensional y sujeto a una serie de restricciones puede tener diferentes soluciones dentro del campo de la investigación operativa tradicional. Tales problemas también pueden resolverse mediante heurísticas en el dominio de la inteligencia artificial y más precisamente en el ámbito de los algoritmos evolutivos bioinspirados. Estos algoritmos, a diferencia de la programación lineal, encuentran resultados que son en todos los casos aproximados dado que controlan un número finito de partículas intervinientes y son sus movimientos aleatorios los que evalúan una posible solución en el espacio de búsqueda n-dimensional.

El algoritmo de optimización por enjambre de partículas (PSO) provee un conjunto de variantes y enfoques que permiten obtener una solución óptima confiable. Estos enfoques posibilitan entre otros objetivos, evitar que las partículas se concentren en óptimos locales, asegurar una amplia exploración del espacio de búsqueda, evitar las búsquedas fuera de la dimensión máxima de la función objetivo y mejorar la convergencia.

El presente artículo estudia una de esas variantes, PSO con factor de constricción versus una hibridación del PSO con vecindad y factor de inercia dinámico. En ambos

casos se emplea una técnica de manejo de restricciones basado en las violaciones de los recursos. El objetivo buscado es analizar si la hibridación de técnicas que mejoran ciertas características del algoritmo logra una mejora global del mismo o se produce una baja de su performance general.

2 El algoritmo PSO con restricciones y sus variantes

El algoritmo Particle Swarm Optimization (PSO) fue propuesto por Kennedy y Eberhart [1] inspirado en la simulación del comportamiento social de bandada de aves. Es un enfoque de búsqueda basado en una población de la cual cada individuo vuela en el espacio de búsqueda con una velocidad que es ajustada de acuerdo a su propia experiencia de vuelo y la experiencia del vuelo de sus compañeros. A diferencia de otros algoritmos evolutivos, en PSO todas las partículas que forman parte de la población se mantienen durante todo el proceso de búsqueda; siendo la velocidad y la posición de las partículas las que se actualizan de acuerdo a la mejor posición encontrada y a las de la otras partículas. Además este es el único algoritmo evolutivo que no implementa la supervivencia del más apto [2]. Para este algoritmo constantemente surgen nuevas versiones respecto del original con el fin de mejorar el rendimiento del proceso de optimización evitando las convergencias prematuras. Entre las variantes más conocidas se pueden mencionar: factor de constricción, vecindad, factor de inercia dinámico, entre otros.

Los problemas de optimización restringida se encuentran en numerosas y diversas aplicaciones: problemas de optimización estructural, diseño de ingeniería, diseño VLSI, de economía, de asignación y ubicación son sólo algunos de los campos científicos en el que se hallan con frecuencia optimización con restricciones [3].

Originalmente, esta heurística no contempla un mecanismo para resolver problemas de optimización restringida, pero existen diversos mecanismos que han sido implementados en algoritmos evolutivos, así como en esta técnica [4]. Uno de los mecanismos más utilizado debido a su sencillez de implementación, es la técnica propuesta por Deb [5] que consiste en reglas basadas en la factibilidad de las partículas durante el ciclo de búsqueda; esto es:

1. Entre 2 soluciones factibles, la solución con un mejor valor de la función objetivo gana.
2. Si una solución es factible y la otra es no factible, la solución factible gana.
3. Si ambas soluciones son no factibles, aquella con el valor más bajo en la suma de la violación de restricción gana.

En este trabajo, se adapta dicho mecanismo para los problemas de producción, en donde las empresas disponen de recursos limitados para su producción. Se modifica el mecanismo de Deb, reemplazando la tercera regla por [6]:

3. Si ambas soluciones son no factibles, se considera aquella con el menor número de restricciones violadas. En el caso de que ambas partículas violen la misma cantidad de restricciones se elige aquella con menor valor en la suma de los porcentajes de violaciones con respecto a los recursos disponibles.

2.1 PSO con factor de inercia dinámico

En el algoritmo clásico de optimización por enjambres se incorpora un parámetro denominado factor de inercia *w*. El objetivo de *w* es equilibrar la búsqueda global y la búsqueda local, encontrando áreas más prometedoras. Las evaluaciones que han sido realizadas con este parámetro, han demostrado un impacto efectivo y muy significativo en los resultados de búsqueda [7]. Esta variable modifica la fórmula de la actualización de la velocidad quedando de la siguiente manera:

$$\overrightarrow{vel_{(i,d)}} = \boldsymbol{w} * \overrightarrow{vel_{(i,d)}} + r_1 * p_1 * \left(\overrightarrow{\boldsymbol{pbest}_{(i,d)}} - \overrightarrow{pos_{(i,d)}} \right) + r_2 * p_2 * \left(\overrightarrow{\boldsymbol{gbest}_{d}} - \overrightarrow{pos_{(i,d)}} \right) \quad (1)$$

Donde $\overrightarrow{vel_{(i,d)}}$ es el valor de la velocidad de la partícula i en la dimensión d, p_1 es el factor de aprendizaje cognitivo, p_2 es el factor de aprendizaje social poblacional, r_1 *y* r_2 son valores aleatorios uniformemente distribuidos en el rango [0, 1], $\overrightarrow{pos_{(i,d)}}$ es la posición actual de la partícula i en la dimensión d, $\boldsymbol{pbest_{(i,d)}}$ es el valor en la dimensión d de la partícula con el mejor valor objetivo encontrado por la partícula i, $\overrightarrow{\boldsymbol{gbest}_{d}}$ es el valor en la dimensión d del individuo del cúmulo (swarm) que encontró el mejor valor de la función objetivo.

El parámetro *w* puede ser constante o linealmente decreciente en el tiempo. En este último caso, se trata de un factor de inercia dinámico y es reducido linealmente utilizando 2 valores límites: *w_inicial y w_final*. Luego, mediante la ecuación (2), el factor de inercia es actualizado en cada ciclo del proceso, hasta el final de las iteraciones (*ciclo_max*).La ecuación de actualización que se utiliza es:

$$\boldsymbol{w} = \frac{(ciclo_max - ciclo) * (w_{inicial} - w_{final})}{ciclo_max} + w_{final} \quad (2)$$

2.2 PSO con factor de constricción

Este parámetro propuesto por Clerc y Kennedy [8], introduce una variable X estática a la ecuación de la velocidad. Este factor se usa para controlar y constreñir las velocidades de las partículas, durante el proceso de búsqueda. La ecuación propuesta es:

$$\overrightarrow{vel_{(i,d)}} = \boldsymbol{X} * \left[\overrightarrow{vel_{(i,d)}} + r_1 * p_1 * \left(\overrightarrow{\boldsymbol{pbest}_{(i,d)}} - \overrightarrow{pos_{(i,d)}} \right) + r_2 * p_2 * \left(\overrightarrow{\boldsymbol{gbest}_{d}} - \overrightarrow{pos_{(i,d)}} \right) \right] \quad (3)$$

Donde al igual que el factor de inercia *w*, el papel más importante que juega este parámetro, es el de asegurar la convergencia del algoritmo y controlar el equilibrio deseado entre el comportamiento de exploración y explotación dentro del espacio de búsqueda. Para asegurar la convergencia, las variables que definen el factor de constricción deben cumplir con los siguientes principios:

$$X = \frac{2k}{\left| 2 - \emptyset - \sqrt{\emptyset(\emptyset - 4)} \right|} \quad (4)$$

$$\emptyset = \emptyset_1 + \emptyset_2$$

$$\emptyset \geq 4$$

$$\emptyset_1 = p_1 * r_1$$

$$\emptyset_2 = p_2 * r_2$$

$k \in [0,1]$

Para valores de $\phi < 4$ el enjambre se acerca lentamente en espiral a la mejor posición encontrada pero sin garantía de convergencia mientras que para valores de $\phi > 4$ la convergencia es rápida y segura [8].

2.3 PSO con neighborhood

A menudo el PSO demuestra una convergencia más rápida en los primeros ciclos, es decir, en la primera fase de la búsqueda, y luego se ralentiza o incluso se detiene cuando el número de generaciones va aumentado. Una vez que el algoritmo se ralentiza, es difícil lograr alcanzar mejores soluciones. Para evitar la terminación o la convergencia en un mínimo local, se introduce una estructura social para determinar la comunicación entre las partículas, la cual se denomina vecindad. De acuerdo a la forma de comunicación entre las partículas que se establezca, se define una topología de vecindad. La idea básica es impulsar esas partículas a una mejor estrategia de comunicación y de esta manera lograr explorar mejor el espacio [9].

En PSO, cada partícula dentro del enjambre pertenece a una vecindad específica de comunicación. Se llevaron a cabo varios estudios con el fin de determinar si la topología de vecindad podría afectar a la convergencia del algoritmo. Estos estudios se basaron en propuestas teóricas e implementaciones de topologías de vecindad comúnmente utilizados por PSO [10]. En esos estudios, algunas topologías de vecindad han obtenido mejores resultados que otros. La forma de agrupamiento de las partículas, puede mejorar el rendimiento del algoritmo. Entre las diferentes propuestas de topologías de vecindad están:

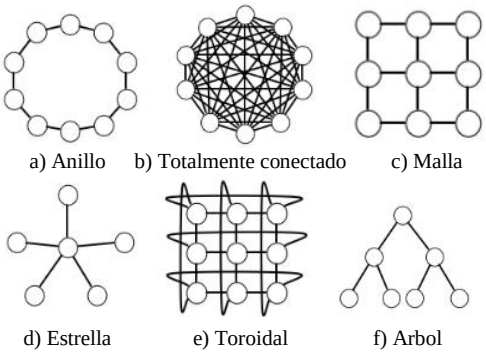


Figura 1: Topologías de vecindades utilizadas en algoritmos de enjambre.

3 Evaluación de los algoritmos y resultados

Con el propósito de lograr una comparación objetiva todas las ejecuciones se configuraron con los mismos valores de parámetros para ambas versiones del PSO, excepto el peso *w* inercia que es dinámico. Los parámetros utilizados fueron los siguientes: tamaño de población (cantidad de partículas) 20; máximo número de

iteraciones 500; factor de aprendizaje cognitivo y social 1.49445; factor de inercia inicial y final de 0.9 y 0.4 respectivamente; factor de constricción de X= 0.7298; cantidad de vecindades 3 de 10 partículas cada una; topología de vecindad estrella. En esta topología implementada todas las partículas de todo el enjambre dirige su vuelo hacia una partícula (la partícula central), y la partícula central dirige su vuelo hacia la mejor partícula de la vecindad. Esta topología debe evitar la convergencia prematura en óptimos locales.

Ambos algoritmos fueron evaluados mediante las funciones de referencia (Benchmark Functions) g04, g08, g09, g15 y g24 debido a que estas son las únicas funciones que contemplan la incorporación de restricciones con disponibilidad. Por otro lado, se evaluó la performance de todos los algoritmos para 500 ciclos dado que en todos los casos se encontró el valor óptimo dentro de ese rango.

Tabla 1: Valores obtenidos del algoritmo PSO con factor de constricción.

FUNCIONES DE PRUEBA - ALGORITMO PSO con FACTOR DE CONSTRICCION-					
G(X)	OPTIMO	MEJOR	PEOR	PROMEDIO	DESV. ESTAND
g04	-30665,539	-30665,0029	-28183,7113	-29927,895	671,561666
g06	-6961,813	**-6961,81388**	-747,977358	-6961,81388	2229,58258
g08	-0,095	**-0,09582504**	-0,02726221	-0,09582504	0,02081586
g09	680,63	680,720704	718,581609	681,336206	8,33319373
g15	961,715	**961,668777**	972,610631	967,356535	3,194949
g24	-5,50801327	**-5,50801327**	-4,03470362	-5,50801327	0,59817649

En las Tablas 1 y 2 se visualizan los resultados estadísticos obtenidos en las 20 ejecuciones para 6 funciones de benchmark con restricciones g04, g06, g08, g09, g15 y g24. Cabe destacar, que el algoritmo PSO con factor de inercia dinámico y vecindad en sus 20 ejecuciones para cada función evaluada encontró soluciones factibles; no así el algoritmo PSO con factor de constricción.

Tabla 2: Valores obtenidos del desempeño del algoritmo PSO con factor de inercia dinámico y vecindad.

FUNCIONES DE PRUEBA - ALGORITMO PSO con FACTOR INERCIA + VECINDAD -					
G(X)	OPTIMO	MEJOR	PEOR	PROMEDIO	DESV. ESTAND
g04	-30665,539	-30665,0029	-30665,00286	-30665,0029	9,24615E-06
g06	-6961,813	**-6961,81388**	-6961,813876	-6961,813876	7,56884E-10
g08	-0,095	**-0,09582504**	-0,095825041	-0,095825041	2,79E-17
g09	680,63	680,65939	685,3846961	681,1705082	1,216223845
g15	961,715	**961,5659**	972,2179663	967,0694139	3,747513459
g24	-5,50801327	**-5,50801327**	-5,508013272	-5,508013272	4,29667E-12

En la Figura 2 puede observarse la gráfica de la convergencia en función de los ciclos para una partícula evaluada en los algoritmos comparados para la función benchmark g04. Se puede observar que ambas versiones para esta función obtienen iguales resultados según Tabla 1 y Tabla 2 y en cuanto a su convergencia también son bastantes similares. En este caso se podría adoptar cualquiera de las versiones, sin embargo, el algoritmo con constricción sería el más adecuado debido a su sencillez en cuanto a programación y por su bajo costo computacional.

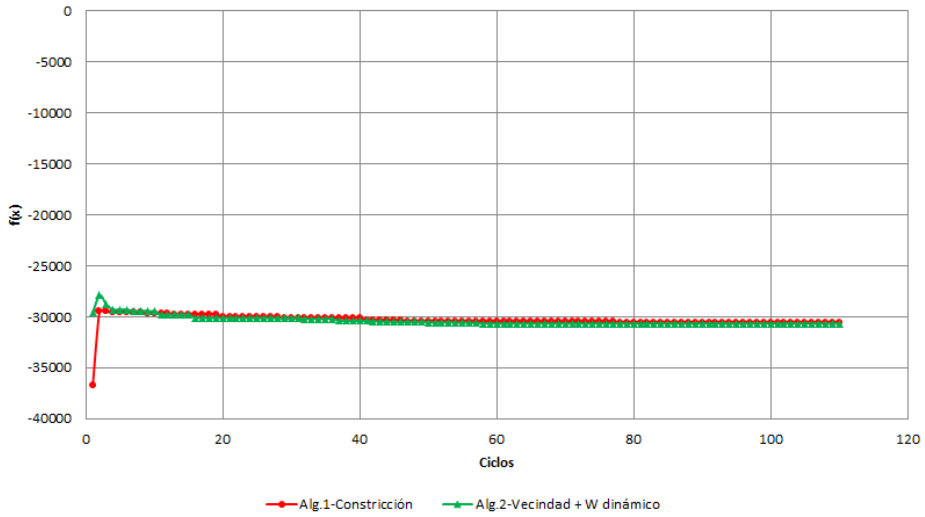


Figura 2: Comparación de algoritmos de constricción y vecindad con factor de inercia dinámico para la función g04 de benchmark.

En la Figura 3 se ilustra la misma comparativa pero evaluada mediante la función benchmark g06. El algoritmo de constricción converge al óptimo más lentamente en comparación con el algoritmo basado en vecindad y coeficiente de inercia dinámico. Es importante resaltar que para esta función, el algoritmo con vecindad se diferencia bastante del de constricción por su robustez como se visualiza en las Tabla 1 y Tabla 2, ya que éste desde el inicio de los ciclos está bastante cerca del óptimo.

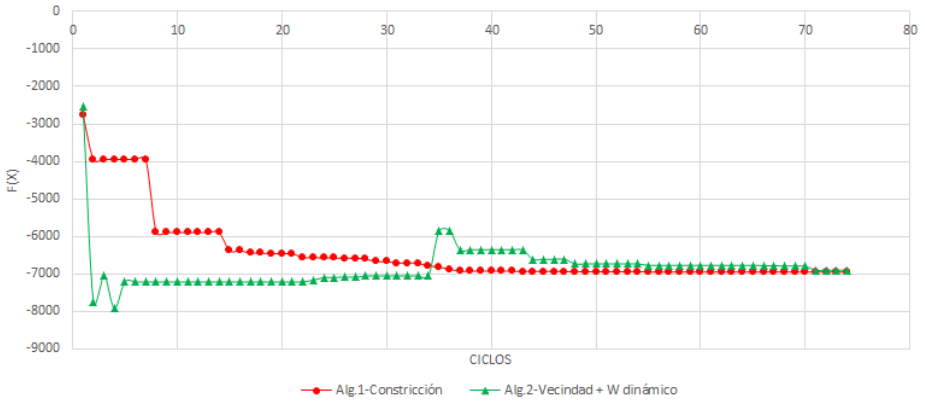


Figura 3: Comparación de algoritmos de constricción y vecindad con factor de inercia dinámico para la función g06 de benchmark.

La función de prueba evaluada, g08, se observa en la Figura 4. En este caso las partículas del algoritmo con vecindad se desplazan muy cerca del óptimo desde los primeros ciclos desde aproximadamente el ciclo 6 en adelante. Es por ello que se justifica un valor de desviación estándar de $2{,}7937*10^{-17}$ según Tabla 2, un resultado

muy superior con respecto al algoritmo de constricción. De todas las evaluaciones esta es la que ofrece la mejor robustez.

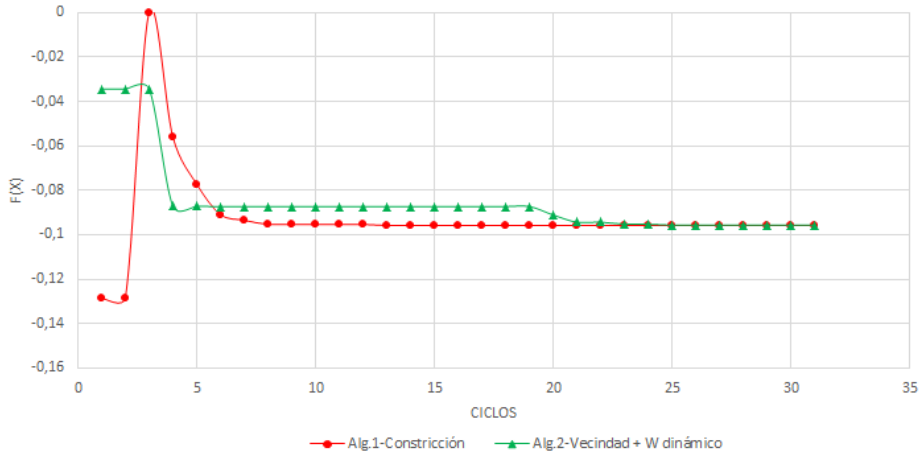


Figura 4: Comparación de algoritmos de constricción y vecindad con factor de inercia dinámico para la función g08 de benchmark.

Los algoritmos comparados mostrados la Figura 5 se evaluaron mediante la función g09 en donde ambos no alcanzan el óptimo conocido, pero se observa que el algoritmo con vecindad tiene un mejor desempeño que el de constricción, debido a que los resultados indican una calidad y robustez superior (ver Tabla 1 y Tabla 2).

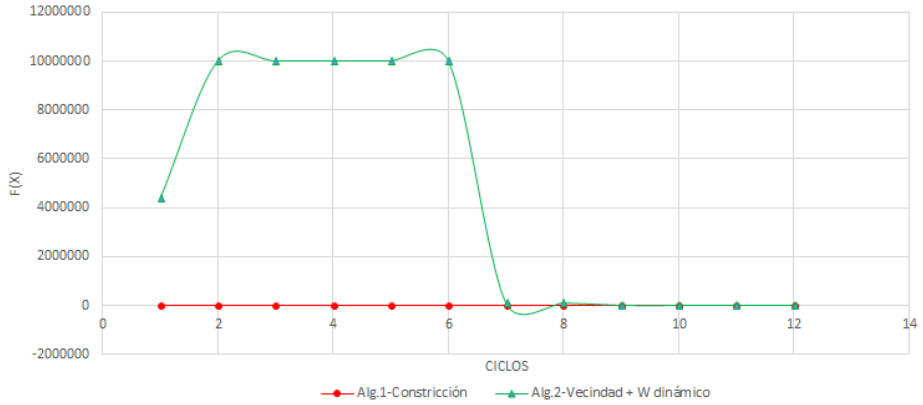


Figura 5: Comparación de algoritmos de constricción y vecindad con factor de inercia dinámico para la función g09 de benchmark.

En la Figura 6 se analizan ambos algoritmos evaluados para la función g15. Se observa encuentran una mejor solución que el óptimo conocido por otros algoritmos de la literatura especializada. En la gráfica al inicio de los ciclos de búsqueda el algoritmo de constricción se encuentra más alejado del óptimo con respecto al algoritmo con vecindad. Sin embargo obtiene una mejor robustez (Tabla 1), pero su

calidad sigue siendo inferior al algoritmo con vecindad y peso de inercia dinámico (Tabla 2). En esta función, el espacio de soluciones es bastante reducido, por lo que se hace más difícil obtener resultados factibles. No obstante, estas dos propuestas hallaron no solo soluciones factibles en las 20 corridas, sino también alcanzaron un muy buen resultado.

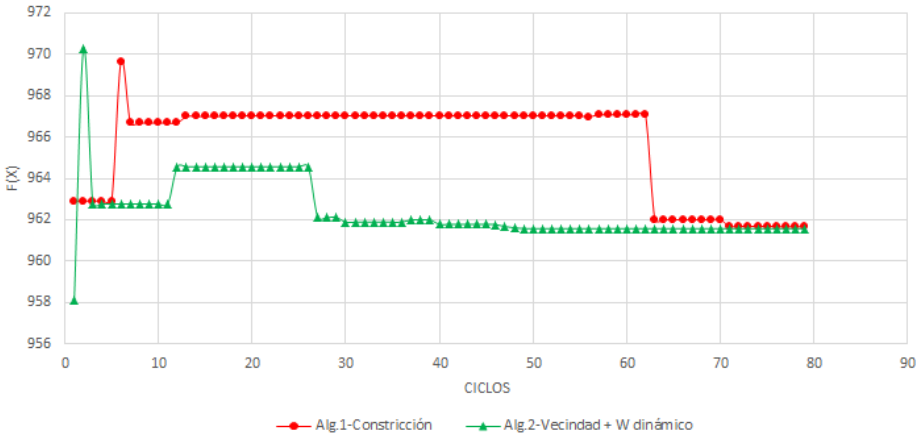


Figura 6: Comparación de algoritmos de constricción y vecindad con factor de inercia dinámico para la función g15 de benchmark.

La última función evaluada y comparada (g24) se encuentra en la Figura 7 en donde ambos algoritmos alcanzan el óptimo. Se observa en la gráfica la convergencia al óptimo y claramente justifica los resultados de la correspondiente desviación estándar observados en la Tabla 1 y Tabla 2, es decir que el algoritmo con vecindad tiene una mejor convergencia al óptimo.

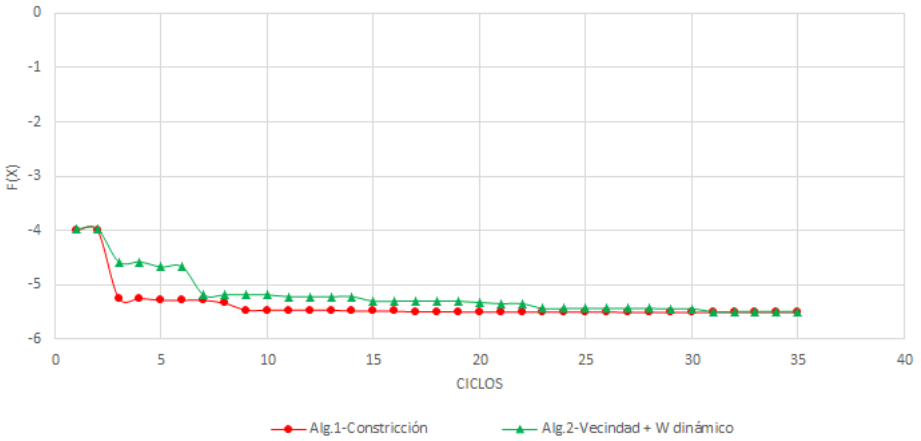


Figura 7: Comparación de algoritmos de constricción y vecindad con factor de inercia dinámico para la función g24 de benchmark.

4 Conclusiones

Los resultados de las evaluaciones presentados en las Tablas 1 y 2, fueron obtenidos con el fin de comparar y seleccionar aquel algoritmo que sea el más adecuado para determinadas funciones. El algoritmo PSO con el parámetro del factor de constricción al igual que el PSO con coeficiente de inercia dinámico y vecindad alcanza el óptimo para las funciones g06, g08, g15 y g15 y se estancan en óptimos locales muy cercanos al óptimo global para las funciones g04 y g09. La calidad de los algoritmos para estas funciones son bastantes similares, pero a su vez se diferencian bastante por su robustez ya que el algoritmo con coeficiente de inercia dinámico y vecindad es superior al de factor de constricción.

Otro aspecto importante que los diferencia en cuanto a las funciones de prueba es la factibilidad de los resultados siendo el algoritmo con factor de inercia y vecindad factible en todas sus ejecuciones.

Las observaciones y el análisis efectuado entre ambos algoritmos permiten afirmar que la hibridación de las técnicas de vecindad con factor de inercia dinámico no siempre incrementa aquellas mejoras que por separado si lo logran.

Es por ello que conviene analizar el posible diseño de hiperheurísticas que permitan la complementación de las mejores características de cada técnica.

5 Trabajos a futuro

Debido a que ambos algoritmos incorporan un mecanismo de manejo de restricciones diferente a los clásicos, es posible plantear modificaciones sobre la técnica citada y observar si se producen mejoras en cuanto a robustez y calidad en las 6 funciones analizadas.

Otro aspecto a considerar es la evaluación y comparación mediante funciones de referencia más actualizadas, esto es, el artículo presente utiliza un conjunto de pruebas de optimización (CEC 2006 Problemas Benchmark [11]) en el cual la dimensión de la función está comprendida entre 2 y 20 lo cual es considerado como un valor de dimensión bajo. Sin embargo, éstas son las funciones que se utilizan habitualmente en las investigaciones referidas al algoritmo PSO para medir su desempeño. En este contexto se pretende en futuras investigaciones estudiar y evaluar el algoritmo con las funciones de referencia restantes además de las funciones de prueba CEC 2010 y 2013[12] [13].

6 Referencias

[1] Kennedy J., Eberhart R. C., Shi Y.: Swarm Intelligence. Morgan Kaufmann, San Francisco, CA (1995)

[2] Eberhart, R. C., Shi, Y. H.: Comparison between genetic algorithms and particle swarm optimization. Annual Conference on Evolutionary Programming, San Diego (1998).

[3] Parsopoulos K.E., Vrahatis M. N.: Particle Swarm Optimization Method for Constrained Optimization Problems (2002).

[4] Coello-Coello C.A.: “Theoretical and numerical constraint-handling techniques used with evolutionary algorithms”: A survey of the state of the art. Computer Methods in Applied Mechanics and Engineering, V.191, No. 11-12:1245–1287 (2002).
[5] Deb K.: “An efficient constraint handling method for genetic algorithms”. Computer Methods in Applied Mechanics and Engineering, v.186 (2/4):311–338 (2000).
[6] Paz F.P., Azar M.A., Herrera Cognetta A., Pérez Otero N.M., Una alternativa para el mecanismo de manejo de restricciones en algoritmos PSO, in Proceedings Second Argentine Conference on Human-Computer Interaction, Telecomunications, Informatics and Scientific Information HCITISI, Córdoba, Argentina, ISBN 978.88.96.471.25.8, Noviembre 21-22 (2013).
[7] Shi, Y. y Eberhart, R. “A modified Particle Swarm Optimizer”. En Proceeding of the 1998 IEEE World Congress on Computational Intelligence, páginas 69–73, Piscataway, NJ (1998).
[8] Clerc, M., The swarm and the queen: towards a deterministic and adaptive particle swarm optimization., Memorias del 1999 Congress on Evolutionary Computation (CEC 99) (1999).
[9] Ajith A, Hongbo Liu, Tae-Gyu Chang. Variable Neighborhood Particle Swarm Optimization Algorithm (2006).
[10] Medina A.J.R., Toscano G.P., Torres R.J.G. A Comparative Study of Neighborhood Topologies for Particle Swarm Optimizers (2009).
[11] Liang J. J., Runarsson T. P., Montes E.M, Clerc M., Suganthan P.N., Coello Coello C.A., Deb. Problem Definitions and Evaluation Criteria for the CEC 2006 Special Session on Constrained Real-Parameter Optimization (2006).
[12] Suganthan, Lǐ, Yáng, Weise. “Benchmark Functions for the CEC'2010 Special Session and Competition on Large-Scale Global Optimization” (2010).
[13] Li, Tang, Omidva, Yang, Qin. “Benchmark Functions for the CEC’2013 Special Session and Competition on Large-Scale Global Optimization” (2013).

Enseñando a programar con RITA en escuelas secundarias

Vanessa Aybar Rosales[1], Claudia Queiruga[1], Isabel Miyuki Kimura[1],
Matías Brown Barnetche[1], Soledad Goméz[2]

[1]LINTI (Laboratorio de Investigación en Nuevas Tecnologías Informáticas), Facultad de Informática, Universidad Nacional de La Plata, calle 50 y 120 (segundo piso)

[2] Facultad de Periodismo y Comunicación Social, Universidad Nacional de La Plata, Diagonal 113 y 63, Nº 291.,. La Plata (1900), Argentina.
{vaybar,claudiaq}@info.unlp.edu.ar {ikimura,mbrown}@linti.unlp.edu.ar
gsoledad25@gmail.com

Abstract. El presente trabajo da cuenta de la complejidad y relevancia de la enseñanza de contenidos de programación en la educación secundaria, como posibilitadora de procesos de aprendizajes que propician la creatividad e innovación con medios digitales. Se socializará la experiencia de trabajo con la herramienta didáctica RITA(Remote Inventor to Teach Algorithms) en escuelas secundarias públicas de la provincia de Buenos Aires, desarrolladas en el marco de proyectos de investigación articulados con actividades de extensión de la Facultad de Informática de la Universidad Nacional de La Plata. RITA es una herramienta libre, desarrollada en el LINTI, que permite programar juegos de robots virtuales mediante "programación con bloques" al estilo LEGO. RITA es un instrumento didáctico que estimula a los estudiantes a participar en el diseño y desarrollo de sus propios juegos y que pone en valor los intereses de los estudiantes, construyendo robots particularmente significativos para ellos.

Keywords: enseñar a programar, escuela, software libre, programación con bloques, JAVA.

1 Introducción

Enseñar a programar en la escuela es hoy una preocupación global, algunos países están a la vanguardia del tema, ejemplo de ello es el Reino Unido que desde 2014 ha incorporado la enseñanza de la programación al sistema de educación escolar a partir de los 5 años; Estonia, una de las economías europeas que más apuesta al desarrollo tecnológico, viene impulsando a través de diferentes iniciativas estatales la enseñanza de la programación en la escuela; varios estados de los EEUU también han implementado políticas activas en respuesta al respaldo de la industria al movimiento "Learn-To-Code", liderado fundamentalmente por las iniciativas globales code.org (http://code.org/) y codeacademy (http://www.codecademy.com). Recientemente el presidente Barack Obama destacó la visión de este movimiento y alentó a los niños y jóvenes a participar en el evento anual "La hora del código" que promueve la educación en "programación" globalmente (https://code.org/educate/hoc) a través de declaraciones emitidas en un video en las que se enfatiza el hecho de formar jóvenes productores de tecnologías y no meros consumidores: "No se compren un nuevo videojuego. Hagan uno. No descarguen la última aplicación. Ayuden a diseñar una. No jueguen con el celular. 'Prográmenlo'" (https://www.youtube.com/watch?v=6XvmhE1J9PY). Israel, Australia, Japón y Finlandia son ejemplos de otros países que cuentan con iniciativas similares en relación de la incorporación de la enseñanza de la informática en las escuelas, en particular sobre contenidos de programación.

Actualmente, el uso de las tecnologías en las escuelas de Argentina no es algo extraño, las computadoras llegaron al aula a través del Programa Nacional Conectar-Igualdad[1], que entregó a la fecha 5.110.545 netbooks a estudiantes de escuelas secundarias públicas como parte del material escolar. Esto está permitiendo superar la primera brecha digital que es la del acceso a las tecnologías digitales. Sin embargo, las computadoras hoy en día son utilizadas por lo general, para navegar y buscar información en Internet y usar aplicaciones estándares como procesadores de texto y planillas de cálculo. Esto pone en evidencia la segunda brecha digital que Tim Berners-Lee (Berners-Lee T., 2013) identifica como aquella que separa a las personas que saben programar de aquellas cuyas habilidades informáticas se limitan a saber cómo trabajar con las aplicaciones estándares. Saber programar es un habilitador para entender e intervenir en un mundo cada vez más tecnológico, permitiendo pensar y actuar creativamente ante problemas inesperados (Resnick M., 2008).

Desde el año 2008 el LINTI está trabajando en una una línea de investigación cuyo objetivo es promover la enseñanza de la programación en las escuelas secundarias, entendiendo que las habilidades que los estudiantes adquieren programando impactan positivamente en otras áreas disciplinares (Díaz J. et al, 2014). Saber programar contribuye al "pensamiento computacional" (Wing J, 2006), permitiendo a los estudiantes que asuman un papel activo en relación al uso de sus computadoras y de los elementos tecnológicos que los rodean. El "pensamiento computacional" promueve el pensamiento analítico, sistemático, fomenta la creatividad y el trabajo colaborativo, todas ellas habilidades consideradas fundamentales para la sociedad del siglo 21 (Resnick M. et al, 2009). Los estudiantes dejan de ser consumidores de tecnología y pueden dar respuestas imaginativas y divergentes y poner en juego sus significados personales.

RITA (Robot Inventor to Teach Algorithm) se propone como una herramienta didáctica, cercana a la cultura adolescente que promueve la enseñanza de la programación mediante un abordaje lúdico. Con RITA los estudiantes programan videojuegos de robots virtuales que compiten en un campo de batalla y cuyo desafio es programar una estrategia de robot ganador. RITA [2]es una herramienta de código fuente abierto desarrollada en el LINTI y actualmente está disponible en el repositorio de software libre github (https://github.com/vaybar/RITA)
Este artículo presenta las experiencias de enseñar a programar en la escuela mediante la aplicación de la herramienta didáctica RITA y se organiza de la siguiente manera:

- La sección ¿Qué es RITA? describe como fue concebida la herramienta, su construcción y su uso
- Las experiencias con RITA, donde se cuenta cómo se desarrolló la interacción con los distintos participantes desde el año 2012, cuáles fueron los resultados relevados de la puesta a prueba de la herramieta y cuál fué el material didáctico elaborado por el equipo de trabajo, para dar soporte y acompañamiento a los docentes en el aula de la escuela.
- Conclusiones del trabajo realizado

[1] Sitio oficial del Programa Nacional Conectar-Igualdad: http://www.conectarigualdad.gob.ar/

[2] El desarrollo de RITA se inicia a partir de la tesina de grado de Lic. en Informática de Vanessa Aybar Rosales

2 ¿Qué es RITA?

RITA es una herramienta didáctica, cercana a la cultura adolescente caracterizada por la fluidez digital, que permite introducir conceptos de "programación" y del lenguaje de programación JAVA. RITA permite crear videojuegos de robots virtuales que compiten en un campo de batalla. Los robots con mejores tácticas -programadas por los estudiantes- perduran en el juego y uno de ellos ganará la batalla.

El desafío propuesto por RITA resulta sumamente motivador para los estudiantes y alienta a pensar en estrategias superadoras que mejoran la calidad del aprendizaje impartido. A su vez, en el proceso de programación de las estrategias de combate se ponen en práctica conceptos de matemática como trigonometría que los alumnos aprenden en la escuela, articulando de esta manera contenidos disciplinares propios de la escuela secundaria.

RITA es una aplicación JAVA de código fuente abierto que extiende, integra y adapta las funcionalidades de dos frameworks también de código fuente abierto, Openblocks[3] y Robocode[4]. Openblocks brinda el soporte para la programación usando bloques gráficos que recrean el uso de piezas de LEGO[5].

En RITA, el framework Openblocks fue extendido para brindar soporte a las clases JAVA provistas por el framework Robocode, así como también proveer las estructuras de control, funciones y operadores de JAVA. De la misma manera que con las piezas de un LEGO, los conectores de OpenBlocks indican cómo pueden unirse los bloques. Los estudiantes comienzan programando con RITA simplemente encastrando bloques y obteniendo secuencias de bloques. De este modo el usuario de RITA puede programar la estrategia de un robot usando bloques gráficos sin preocuparse por la sintaxis y semántica del lenguaje JAVA. Automáticamente y en forma transparente para el usuario de RITA, la programación en bloques se traduce en código JAVA.

La Fig. 1 muestra la pantalla principal de RITA en la que puede observarse el ambiente de programación que propone la herramienta: el panel de la izquierda contiene los bloques disponibles y el panel central es el área de trabajo principal. La interacción que se utiliza es la de *drag&drop* (arrastrar y soltar), los bloques se arrastran del panel de la izquierda y se sueltan en el área de trabajo principal y de esta manera se programa la estrategia del robot. A su vez, en esta figura se muestra la programación de una estrategia de combate simple junto con el código JAVA correspondiente, generado automáticamente en RITA.

---

[3] OpenBlocks: http://education.mit.edu/openblocks

[4] Robocode: http://robocode.sourceforge.net/

[5] LEGO: http://www.lego.com/es-ar/

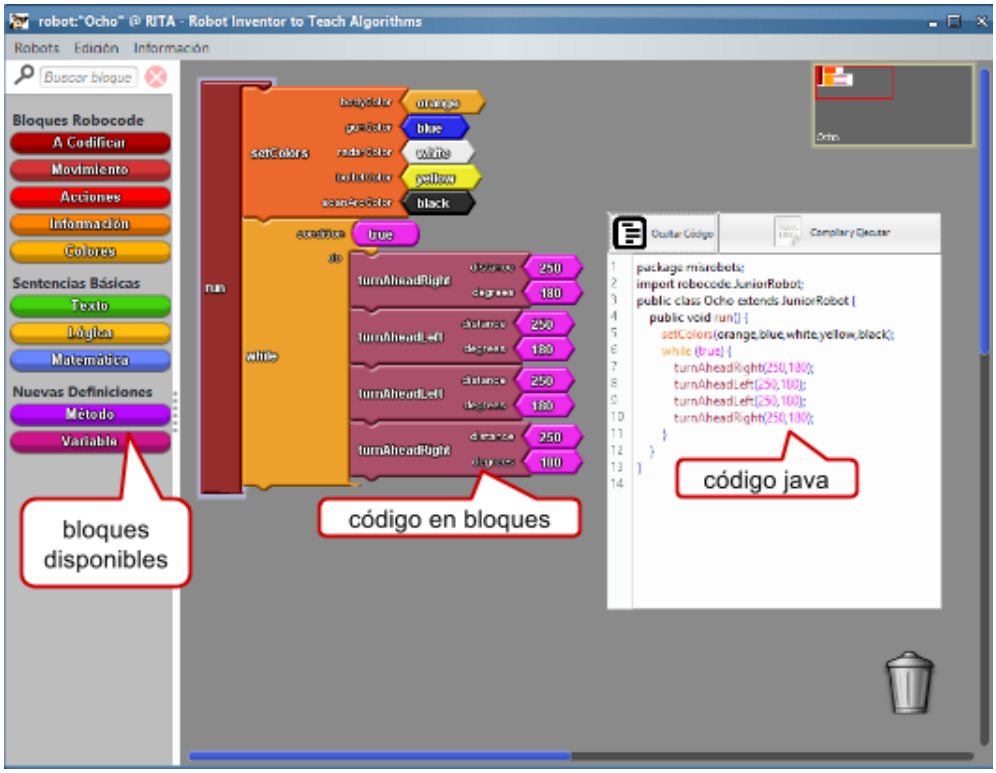


Fig. 1 Ambiente de programación de RITA. Los bloques disponibles del panel de la izquierda pueden arrastrarse al área central donde se estructura la estrategia del robot.

Robocode es un framework de código fuente abierto que brinda un conjunto de clases JAVA que representan a los robots y la interacción entre ellos, el soporte para la ejecución de los robots y el motor de ejecución de la batalla. Robocode es un juego de programación cuyo objetivo es programar la estrategia de un robot para competir contra otros robots en un campo de batalla. El jugador es el programador del robot y mediante el código que escribe le brindará inteligencia al robot, indicando cómo comportarse y reaccionar frente a eventos ocurridos en el juego. Robocode provee a RITA de un campo de batalla, donde pequeños robots virtuales compiten. Robocode propone una batalla, sin embargo no promueve la violencia, no se involucran personas, no contiene sangre ni se plantean situaciones de enfrentamiento racial; se promueve la competencia de las estrategias implementadas por cada jugador en un sentido positivo. Los alumnos prueban mediante el juego sus estrategias de combate. La Fig. 2 muestra una pantalla con robots compitiendo en el campo de batalla de Robocode.

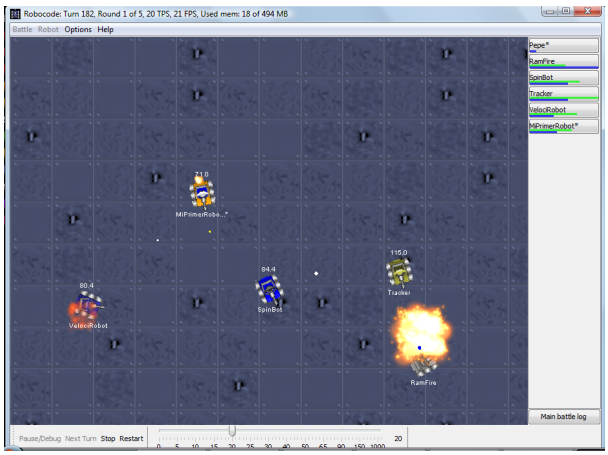


Fig 2. Campo de batalla de Robocode. En el margen derecho se muestra información de los robots participantes y su estado actual.

RITA provee un entorno de programación integrado, donde los estudiantes "programan con bloques" las estrategias de combate de sus robots y luego las prueban en el campo de batalla de Robocode. En esta experiencia de juego con RITA se exploran conceptos de programación. tales como: secuencia, iteración, eventos, paralelismo, estructuras de control, operadores matemáticos, expresiones lógicas, variables. A su vez, estos conceptos son puestos en práctica en la resolución de problemas usando un enfoque incremental e interactivo, testeando y corrigiendo errores, reusando, abstrayendo y modularizando.
RITA se encuentra disponible para equipos con sistema operativo Windows a partir de la versión XP y en diferentes distribuciones de Linux, entre ellas Huayra, Ubuntu y Lihuen[6]. RITA funciona correctamente en las netbooks del Plan Nacional Conectar-Igualdad, siendo estas últimas computadoras las que se usaron en la experiencias realizadas con alumnos de escuelas secundarias.

3 Las experiencias con RITA

Las experiencias con RITA comenzaron en el año 2012 con el proyecto de extensión *Articular universidad-escuela con JAVA para fortalecer la Educación-Técnica*[7] (Queiruga C. et al, 2012). En dicho proyecto se trabajó con docentes y estudiantes de cuatro escuelas secundarias técnicas de la Provincia de Buenos Aires. Dentro de esta propuesta se incorpora a modo de experiencia piloto RITA, como herramienta para facilitar el trabajo con contenidos de programación. Es importante destacar que los proyectos desde los cuales se profundizó el trabajo con RITA se enmarcaron dentro de la política de inclusión y divulgación que adopta la Facultad de Informática de la UNLP, específicamente desde la Secretaria de Extensión, con el objetivo de promover las carreras de grado.
En 2013 como resultado de la experiencia de 2012, se rearma la propuesta y se presenta el proyecto *Articular universidad-escuela con JAVA para fortalecer la Educación-Técnica. Conectar Saberes*,

[6] Lihuen: https://lihuen.linti.unlp.edu.ar
[7] Sitio del articular universidad-escuela: http://jets.linti.unlp.edu.ar/

el cual además de continuar la experiencia de trabajo con escuelas profundiza el trabajo con la herramienta incorporando escuelas de modalidad media. Asimismo en el año 2014, se trabajó específicamente en el espacio curricular NTICs de una escuela secundaria media, donde la propuesta de trabajo a partir de contenidos de programación requiere del uso de herramientas facilitadoras, teniendo en cuenta que la formación de los estudiantes no contempla estos contenidos. El trabajo con RITA en las aulas permitió no sólo la transferencia sino también la apropiación del contenido por parte de los alumnos, quienes en 2015 continúan participando en el marco del nuevo proyecto *Extensión en vinculo con escuelas secundarias,* donde participan ocho escuelas de la provincia de Buenos Aires, de modalidad técnica y media.
A partir de la experiencia de trabajo realizada durante estos 3 años daremos cuenta de algunos ejes de análisis, que nos permiten materializar la experiencia para poder reflexionarla y compartirla. Estos ejes de análisis son: los sujetos de la experiencia (docentes y alumnos) los materiales didácticos generados, y la complejidad del proceso de transmisión[8].

3.1 Acerca de los docentes

El eje en relación a los sujetos constituye un factor fundamental, no sólo para esta propuesta sino para las actividades educativas en general. En este caso el trabajo con docentes, nos permite conocer en profundidad el territorio de la escuela, sus complejidades, desafíos y cotidianeidades. Es a partir de la articulación con docentes y en algunos casos con directivos, que nos hacemos eco de las problemáticas dentro de las aulas y de los desafíos que presenta el desarrollo de las propuestas de contenidos en programación. En esta instancia de interacción con "colegas" interpretamos que mucho de lo que está en los diseños curriculares de la escuela secundaria, no puede llevarse a acabo por múltiples factores, entre los que se destacan; la incumbencia profesional de los docentes, la falta de herramientas, la falta de infraestructura, la falta de interés de los estudiantes respecto de contenido asociado a las ciencias exactas[9]. Sin embargo el entusiasmo y la receptividad de los docentes y autoridades de los distintos centros educativos, fue muy buena y permitio articular las necesidades con la herramienta. De esta forma el trabajo en las aulas estuvo impulsado por los docentes que en primer lugar se capacitaron respecto del uso de RITA y luego pensaron como incorporar los contenidos a la curricula.
A partir de este primer acercamiento, a las escuelas, a los docentes y a los estudiantes, se hizo posible la implementación de RITA dentro de las aulas. Los docentes destacaron que la propuesta les permitía acercar los contenidos de programación de una manera sencilla y motivadora, ya que a los estudiantes les resulta próximo el formato de juego y las pantallas forman parte de su escena más cotidiana.
Una vez constituido el equipo de trabajo con los docentes, se realizaron reuniones quincenales con el objetivo de diseñar y planificar los ejercicios para que los alumnos recorran la temática de los contenidos de programación en forma gradual.

---

[8] Pensamos los procesos educativos desde la complejidad, a partir de las funciones de transformación y conservación propias de la educación, lo que nos obliga a pensar no sólo en las formas de transmisión sino en la posibilidad de creación que ofrece aquello que se transmite. Entendemos a la transmisión como posibilidad, en el caso especifico de esta experiencia, la pregunta respecto de cómo enseñar a programar, nos permitio problematizar la herramienta: RITA.

[9] En diálogo con los docentes, es recurrente la referencia respecto de que los estudiantes, manifiestan falta de confianza e iniciativa para trabajar contenido de las materias de: matemática, física y química, entre otras..

3.2 Acerca de los alumnos

Como mencionamos anteriormente RITA comenzó como una tesina de grado de la Facultad de Informática. Como tal, se debía sustentar su utilidad con trabajo de campo. Es así como se propone que sea evaluada por los docentes que participaban del proyecto *Articular universidad-escuela con JAVA para fortalecer la Educación-Técnica*. Los docentes pertenecían a las escuelas:

- Escuela de Educación Secundaria Técnica Nº 2 de Berisso
- Escuela de Educación Secundaria Técnica Nº 5 de Berazategui
- Escuela de Educación Secundaria Técnica Nº 3 de Mar del Plata.

Los docentes recibieron capacitación en el uso de RITA y dieron su aprobación para continuar realizando pruebas con alumnos pertenecientes a sus cursos.

En un primer encuentro -la prueba piloto de RITA con alumnos- realizado en julio del 2012 fueron convocados 11 alumnos de las escuelas tècnicas de Berisso y Berazategui. La jornada de trabajo consistió en una primera etapa de explicación de las reglas de juego y una segunda etapa donde los alumnos en equipos de 2 o 3 personas construyeron la estrategia de su robot que luego sería puesto a prueba. Los alumnos participantes se mostraron muy entusiasmados y realizaban consultas continuamente lo que demostraba su interés en aprender para poder realizar una estrategia ganadora. Al final de la jornada todos los robots construídos por los alumnos fueron puestos a prueba, los alumnos se mostraron expectantes de ver cómo funcionaba la estrategia elaborada. Finalmente, luego de la competencia se determinó al equipo ganador. En los días sucesivos, algunas de las apreciaciones y comentarios de los docentes que habían participado de los encuentros se referían al interés de los alumnos acerca de seguir trabajando con RITA.

Esta prueba piloto entusiasmó a los docentes quienes con muy buena predisposición cedieron espacio en sus respectivas escuelas, a la que se sumó la escuela de Mar del Plata para hacer pruebas de campo con alumnos. Considerando que el *feedback* recibido en el primer encuentro podría resultar subjetivo, en los sucesivos encuentros se relevaron a los alumnos a través de encuestas. Los resultados de estas encuestas evidenciaron que RITA es una herramienta interesante para introducir en el aula de escuela secundaria en los espacios curriculares propios de programación o NTIC.

La Fig. 3 muestra un cuadro comparativo obtenido de las encuestas en la que se buscaba conocer cuan simple resultó la experiencia con RITA. Participaron de esta encuesta 100 alumnos. Para entender este cuadro: *"Pensás que la explicación de Vanessa alcanza para empezar a usar RITA"*, debemos tener en cuenta que los alumnos no saben de qué se trata el encuentro y que es la primer vez que reciben información de RITA y las reglas de combate. Con eso deben crear una estrategia en 50 minutos.

Fig. 3. El cuadro demuestra que con una breve explicación, los alumnos se sintieron capaces de empezar a usar la herramienta.

La Fig. 4 muestra la capacidad de combate de los robots desarrollados por los estudiantes para el desafío propuesto. Más del 70% de los estudiantes encuestados pudieron construir un robot capaz de combatir, un porcentaje cercano al 25% consideró que su robot no estaba debidamente preparado para una batalla, sin embargo con más tiempo se hubieran animado a participar de la batalla. Un porcentaje muy chico de estudiantes no respondieron la pregunta.

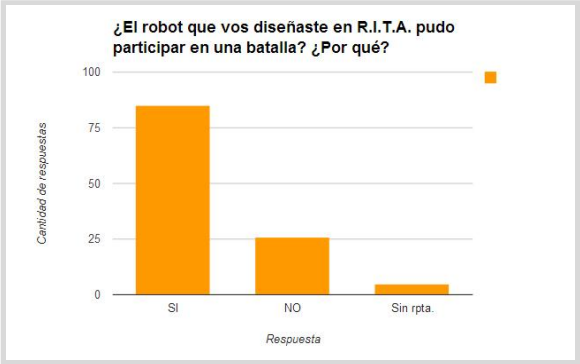


Fig. 4. el cuadro muestra la proporción de alumnos que pudieron construir un robot capaz de ponerse a prueba frente al de sus compañeros

Los alumnos participantes cuentan con experiencia en algún lenguaje de programación que no es JAVA. Sin embargo, manifestaron sentirse más cómodos usando la programación en bloques. La Fig. 5 muestra que más del 90% de los alumnos consideró más simple programar con bloques a pesar de tener conocimiento de programación.

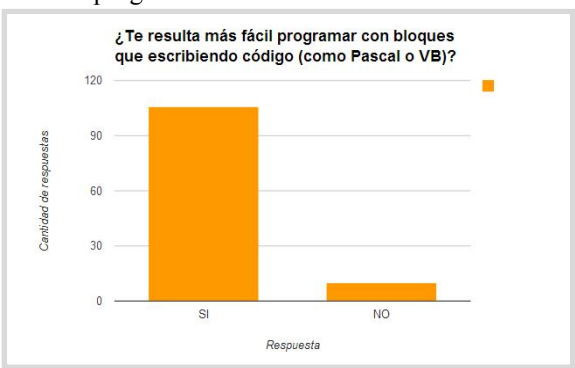


Fig. 5. El cuadro muestra la preferencia de usar de bloques frente a un lenguaje de programación.

La Fig. 6 buscó indagar sobre el grado de dificultad encontrado para entender el código JAVA generado automáticamente: más del 75% respondío que le resultó fácil entender el código JAVA a partir de la programación en bloques, un porcentaje cercano al 20% le resultó difícil y el resto consideró mediana la dificultad.

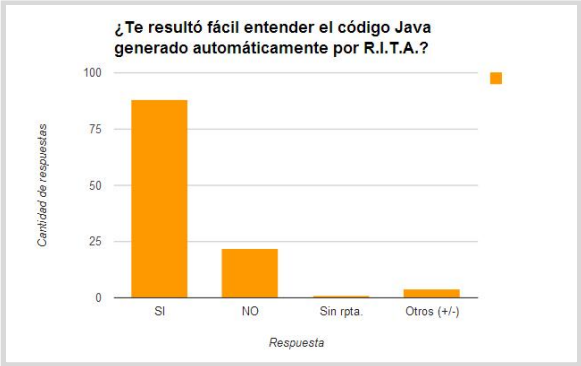


Fig. 6. El cuadro muestra que los alumnos encontraban correspondencia entre la estructura de bloques que ellos crearon y el código Java generado automáticamente por RITA.

Las conclusiones obtenidas a partir de las encuestas, fueron las siguientes:

- En 100 minutos el 93% de los alumnos encuestados, sin experiencia previa con RITA pudieron construir un robot con una estrategia.
- El 91% de alumnos encuestados, encuentran que programar por bloques les resultaba más fácil que hacerlo con un lenguaje de programación ya que se evitaban entender una sintaxis que en general les resulta difícil.
- El 73% de los estudiantes pudieron en un encuentro de 2 horas programar un robot capaz de combatir en un campo de batalla.
- El 76.5% entendieron el código JAVA a partir de los bloques, aún cuando nunca habían visto código JAVA antes.

Durante el año 2013 se continuó trabajando y mejorando RITA en función de sugerencias realizadas por los mismos docentes. También se usó RITA como herramienta de trabajo en jornadas de pasantías universitarias organizadas por la Secretaría de Extensión de la Facultad de Informática, donde alumnos de escuelas secundarias se acercaron a la facultad a trabajar en alguna temática relacionada a informática.

En el año 2014 se comenzó a trabajar con escuelas secundarias no-técnicas de la ciudad de La Plata. Se planificaron junto con los docentes del área NTIC de la escuela la realización de cuatro jornadas-taller con estudiantes de 4to año de la escuela secundaria, de entre 15 y 16 años. De estos talleres participaron 25 estudiantes, en el primero de ellos se les propuso que instalarán y probarán RITA en sus netbooks usando la conectividad de la escuela. En el segundo y tercer taller se trabajaron conceptos de programación y algorítmica proponiendo la resolución de problemas referidos al desplazamiento de los robots en el campo de batalla de RITA (plano) aplicando cálculos aritméticos y trigonométricos, cuestiones vinculadas a la administración de la energía los robots, a la detección de otros robots, etc. En el último encuentro se realizó una competencia entre todos los robots construidos, se pusieron en común todas los robots en un único juego, resultando un robot ganador de esta competencia. La experiencia fue muy positiva y valorada por la comunidad de la escuela, esto nos permitió participar en el evento Universo PROGRAM.AR realizado en Tecnópolis en octubre de 2014 (http://goo.gl/H08LV3) (http://goo.gl/GiUvhO). La encuesta aplicada para relevar la experiencia con RITA dio como resultado que a la mayoría de los estudiantes les resultó fácil -o regular en menos casos- trabajar programando con bloques, además prácticamente todos pudieron cumplir la consigna de crear una estrategia para su robot y ponerlo a

prueba en el aula. Con respecto a sus preferencias relacionadas a aplicaciones informáticas, prácticamente todos indicaron su inclinación por los videojuegos y las redes sociales.
Durante el año 2015 se realizaron en la Facultad de Informática de UNLP actividades de vinculación con escuelas secundarias en torno a diferentes temáticas informáticas, en el marco del proyecto "Extensión en vínculo con Escuelas". El taller "Programar videojuegos de robots con RITA" fue una de las actividades ofrecidas. Los alumnos participantes de este taller tenían entre 17 y 18 años y cursan 6to año y/o 7mo año de escuela secundaria. No todos provenían de escuelas con orientación en informática. Se realizaron 3 encuentros de 2 horas cada uno y la metodología de trabajo fue de taller. Los estudiantes se manifestaron interesados y motivador por programar su propio videojuego.

3.3 Materiales Didácticos

A partir de las experiencias obtenidas algunos docentes manifestaron interés en incluir RITA como instrumento didáctico en sus clases de NTIC o laboratorios de programación. En forma conjunta con docentes de escuelas secundarias comenzó un proceso de producción de materiales didácticos que pudiera dar soporte al desarrollo de actividades teóricas y prácticas en torno a la enseñanza de programación con foco en la programación en bloques y RITA. Actualmente se encuentra disponible:

- material en forma de diapositivas con abundantes ejemplos que fue empleado por el equipo para dictar la capacitación en las escuelas y que puede ser usado por los docentes para replicar el curso
- un manual para el docente con contenido teórico sobre los fundamentos de programación, su aplicación a la resolución de problemas con RITA y ejercicios resueltos, algunos de ellos propuestos en las diapositivas.

4 Conclusiones

La incorporación de una manera planificada y reflexiva de nuevas herramientas didácticas relacionadas al aprendizaje de la programación en la escuela secundaria, promueve habilidades que son útiles para otras áreas del conocimiento, amplía las posibilidades de las "cosas" que se pueden crear, inventar con la computadora y en general de las cosas que se pueden aprender.

El equipo multidisciplinario finalmente consolidado después de distintas experiencias realizadas, hace posible abordar la complejidad de enfrentar la adaptación a los nuevos contenidos y facilitar la transición de los alumnos de escuelas secundarias al ámbito universitario, adaptanto las prácticas educativas. Como plantea Freire: "la complejidad de la práctica educativa es tal, que nos plantean la necesidad de considerar todos los elementos que puedan conducir a un buen proceso educativo, nos impone la necesidad de inventar situaciones creadoras de saberes, sin las cuales la práctica educativa auténtica no podría darse" (Freire P, 2004).
RITA provee un puente entre pensar algorítmicamente y traducir el algoritmo a un lenguaje en particular. RITA es una herramienta que funciona como integradora en tanto permite el abordaje de contenidos de programación, de manera sencilla, a la vez que prepara a los estudiantes para la comprensión y utilización de un lenguaje de programación (JAVA). Esta herramienta permite introducir conceptos de programación procedural, de programación orientada a objetos y de programación orientada a eventos.

RITA brinda una respuesta a uno de los principales problemas de la educación secundaria y técnica en informática: *el interés*. Los alumnos trabajan a partir de la competencia, entienden que "aprender más lleva a un mejor robot que puede ganarle a mis compañeros". Además RITA es una herramienta que induce a los alumnos a aprender a programar explorando e indagando, a partir del agrupamiento de bloques. Los estudiantes acuerdan que "programar es divertido" basicamentente porque el juego les permite interactuar, competir y aprender .
La experiencia realizada además de la implementación y puesta a prueba de la herramienta nos permitio entender y comprender por qué es relevante y fundamental comenzar a trabajar estos contenidos en el nivel secundario. Nos permitio conocer el ambiente del aula en la escuela y aportar a los docentes una herramienta para enseñar.

5 Referencias

Berners-Lee Tim (2013): "Saber programación es la nueva brecha digital", BUSINESS TI | PCWorld España. 05/02/2013. http://www.pcworld.es/business-ti/saber-programacion-es-la-nueva-brecha-digital-segun-bernerslee [Último acceso 10/08/2015].

Díaz Javier; Banchoff Tzancoff Claudia; Queiruga Claudia; Martin Eliana Sofia (2014): "Experiencias de la Facultad de Informática en la Enseñanza de Programación en Escuelas con Software Libre", Congreso Iberoamericano de Ciencia, Tecnología, Innovación y Educación. Organizado por la Organización de Estados Iberoamericanos para la Educación, la Ciencia y la Cultura (OEI)., ISBN 978-84-7666-210-6. Artículo 1426. Buenos Aires, Argentina, del 12 al 14 de Noviembre de 2014.

Freire P (2004). Pedagogía de la autonomía. Ed. Paz e Terra, Sao Paulo.

Queiruga C, Fava L (2012) "Articular Universidad Escuela: una experiencia de intervención de la Facultad de Informática de la UNLP ". 10° Simposio sobre la Sociedad de la Información, JAIIO-SSI 2012.

Resnick Mitchel (2008): "Sowing the Seeds for a More Creative Society". Learning & Leading with Technology, 2008, 35(4), 18-22.

Resnick M, Maloney J, Monroy-Hernández A, Rusk N, Eastmond E, Brennan K, Millner A, Rosenbaum E, Silver J, Silverman B, Kafai Y (2009): "Scratch: programming for all", Communications of the ACM, Vol 52, Nº 11.

Wing J (2006) "Computational thinking". Communications of ACM. Vol 49, Nº 3.

Incorporando conceptos en la enseñanza de Concurrencia y Paralelismo utilizando el entorno CMRE

Laura De Giusti[1], Fabiana Leibovich[1], Franco Chichizola[1], Marcelo Naiouf[1], Armando De Giusti[1,2]

[1] Instituto de Investigación en Informática LIDI (III-LIDI) – Facultad de Informática –UNLP
[2] Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
Argentina
{ldgiusti, fleibovich, francoch, mnaiouf, degiusti}@lidi.info.unlp.edu.ar

Abstract. Se analiza el modo de incorporar los conceptos de heterogeneidad de los procesadores y su impacto sobre el speedup y la eficiencia de un sistema paralelo, así como el estudio de la eficiencia energética de algoritmos paralelos, en función de la potencia de los procesadores.
El entorno CMRE (Concurrent Multi Robot Environment) se extiende para poder considerar relojes virtuales diferentes por robot (procesador) y también el costo en tiempo y en consumo de energía de las operaciones que realizan los robots (Moverse, Depositar / Recoger / Comunicarse por Mensajes / Informar).
En el trabajo se analizan algunos casos ejemplo para mostrar el modo de presentar los conceptos para el alumno.

Keywords: Concurrencia, Paralelismo, Procesadores heterogéneos, Algoritmos paralelos, Consumo energético.

1 Introducción

La Concurrencia ha sido un tema central en el desarrollo de la Informática y los mecanismos de expresión de procesos concurrentes que cooperan y compiten por recursos ha estado en el núcleo curricular de los estudios de Informática desde la década del 70, en particular a partir de los trabajos fundacionales de Hoare, Dijkstra y Hansen [HOA85][DIJ78][HAN77]. Los conceptos se enseñaron tradicionalmente asumiendo la disponibilidad de un único procesador que podía explotar parcialmente la concurrencia del algoritmo, en función de la arquitectura física disponible (incluso con hardware específico como coprocesadores, controladores de periféricos o esquemas vectoriales que replicaban unidades aritmético-lógicas).
El paralelismo, entendido como "concurrencia real" en la que múltiples procesadores pueden operar simultáneamente sobre varios threads o hilos de control en el mismo instante, resultó durante muchos años una posibilidad limitada por la tecnología de hardware disponible [DAS89]. Las currículas informáticas clásicas [ACM68][ACM78][ACM99] contenían conceptos de concurrencia en diferentes áreas (Lenguajes, Paradigmas, Sistemas Operativos) pero omitían casi totalmente el tratamiento del paralelismo, salvo al plantear los conceptos de sistemas distribuidos.

Las actuales arquitecturas de procesadores, que integran múltiples “cores” o núcleos en un procesador físico han producido un notorio impacto, obligando a replantear el “modelo base” de un procesador. Esto ha llevado a reemplazar el formato de “máquina de Von Neuman” con un solo hilo de control, por un esquema que integra múltiples “cores” cada uno con uno o más hilos de control y varios niveles de memoria accesible en forma diferenciada [GEP06][MCC08].
Al mismo tiempo, los cambios tecnológicos han producido una evolución de los temas de mayor interés en Informática, fundamentalmente por las nuevas aplicaciones que se desarrollan a partir de disponer de arquitecturas y redes de comunicación de mayor potencia y menor costo [HOO13]. Por este motivo, las recomendaciones curriculares internacionales mencionan la necesidad de tratar los temas de concurrencia y paralelismo desde las etapas tempranas de la formación del alumno, dado que todas las arquitecturas y sistemas reales con los que trabajará son esencialmente paralelos [ACM13]. Sin embargo, la programación paralela (y los conceptos fundamentales de concurrencia) resulta más compleja para un alumno en las etapas iniciales de su formación, y es necesario contar con nuevas estrategias que permitan abordar.
Dados los estímulos que los alumnos reciben desde temprana edad, ya sea mediante juegos electrónicos, computadoras, celulares, tablets o cualquier otro dispositivo electrónico, la utilización de herramientas interactivas para la enseñanza de conceptos fundamentales a alumnos desde un curso CS1 [ACM04][ACM08][ACM13] se ha vuelto una herramienta fundamental [DEG14a]. En este sentido, la posibilidad dar los primeros pasos en el mundo de la programación mediante un entorno gráfico e interactivo permite reducir la brecha que tradicionalmente existió entre la abstracción y la posibilidad de ver gráficamente la aplicación de los conceptos estudiados en un entorno que conceptualmente es similar a los utilizados en la vida cotidiana [AMD09][HOO13].
El entorno gráfico CMRE, donde se cuenta con un conjunto de robots que se mueven en una ciudad, ha permitido incorporar la enseñanza de los conceptos básicos de concurrencia y paralelismo en un curso inicial de Informática. Sin embargo, existen características avanzadas (tales como heterogeneidad, costo en tiempo, balance de carga, consumo de energía, etc) que resulta de interés introducir a fin de utilizar el entorno en cursos de años superiores. Este trabajo presenta la extensión de CMRE para tener en cuenta los conceptos mencionados.
El trabajo está estructurado de la siguiente manera: la Sección 2 presenta el entorno CMRE actual. En la Sección 3 se aborda el problema de la heterogeneidad, mientras la Sección 4 se centra en el consumo energético. En la Sección 5 se detallan las conclusiones y líneas de trabajo futuro.

2 El entorno CMRE actual

Las características principales del entorno CMRE pueden resumirse de la siguiente manera [GEP06][DEG14b]:

- Existen múltiples procesadores (robots) que realizan tareas y que pueden cooperar y/o competir. Los mismos representan los “cores” de una arquitectura multiprocesador real. Estos robots virtuales pueden tener un reloj propio y diferentes tiempos para la ejecución de sus tareas específicas.

- El modelo de ambiente (“ciudad”) en la que desarrollan sus tareas admite áreas privadas, parcialmente compartidas y totalmente compartidas. En una área privada sólo puede moverse un único robot, en un área parcialmente compartida se especifica el conjunto de robots que pueden moverse en ella y en un área totalmente compartida todos los robots definidos en el programa pueden moverse dentro de ella.
- Si se instancia a un sólo robot en un área que abarque toda la ciudad, se repite el esquema del Visual Da Vinci.
- Cuando dos o más robots están en un área compartida (parcial o totalmente), compiten por el acceso a las esquinas del recorrido y a los recursos que allí existan. Para esto deben sincronizar.
- Cuando dos o más robots (en un área común o no) desean intercambiar información (datos o control) deben hacerlo por mensajes explícitos.
- La sincronización se da por un mecanismo equivalente a un semáforo binario.
- La exclusión mutua puede generarse con la declaración de las áreas alcanzadas por cada robot. Acceder a otras áreas de la ciudad, así como salir de ellas no está permitido.
- Todo el modelo de ejecución es sincrónico y permite la existencia de un reloj virtual de ciclos, que a su vez permite asignar tiempos específicos a las operaciones, simulando la existencia de una arquitectura heterogénea.
- El entorno permite ejecutar el programa de manera tradicional, o paso a paso por instrucciones, dando al usuario un control detallado sobre la ejecución del programa, de manera de poder controlar situaciones típicas de concurrencia tales como conflictos (colisiones) o deadlocks.
- En la ejecución paso a paso, el efecto de las operaciones se puede reflejar en los robots físicos, comunicados vía Wi-fi. Los robots físicos poseen un sistema operativo Linux que permite ejecutar un servidor http implementado en NodeJS [NJS15]. De esta manera el entorno se comunica con los robots (cada robot físico se corresponderá con uno virtual en el ambiente). La comunicación entre ellos es punto a punto, y bidireccional, es decir, el entorno envía las instrucciones al robot físico y luego este último envía su respuesta al entorno indicando la finalización de la instrucción indicada.

3 Heterogeneidad en arquitecturas paralelas

3.1- Conceptos generales

Históricamente se ha buscado incrementar el poder computacional de los sistemas de computación. Sin embargo, se ha llegado a la situación de que resulta difícil acelerar la velocidad de los procesadores incrementando la frecuencia de reloj de los mismos. Son dos los problemas que los arquitectos de hardware deben enfrentar: la generación de calor y el consumo de energía. La solución que presentaron los diseñadores a estos problemas ha sido integrar dos o más núcleos computaciones dentro de un mismo chip, lo cual se conoce como procesador multicore o multinúcleo. Los procesadores multicore mejoran el rendimiento de una aplicación al distribuir el trabajo entre los núcleos disponibles [GEP06] [MCC07].
En una clasificación general actualmente existen dos tipos de multicores de acuerdo a las características de sus núcleos:

- Arquitecturas Multicores Homogéneas: todos sus cores poseen las mismas características.
- Arquitecturas Multicores Heterogéneas: poseen cores con diferentes características en cuanto a rendimiento y consumo de energía, pudiendo utilizar o no distintos ISA (instruction set architecture).

Actualmente la investigación se está enfocando en este segundo tipo de Multicores, dado que tener cores de diferentes tipos permite optimizar el rendimiento, y al realizar una correcta distribución de las tareas entre los núcleos, se logra una mayor eficiencia en la relación rendimiento/energía.

En este tipo de arquitectura, la heterogeneidad se da en diferentes aspectos, los más importantes son la potencia de cómputo (velocidad de cómputo) de los cores; el tiempo de acceso a memoria; la velocidad de comunicación entre cores. Estos tres aspectos definen el Tiempo de Ejecución de las instrucciones en cada core, por lo que una misma sentencia ejecutada en dos núcleos diferentes puede insumir tiempos distintos. Por otro lado, al existir un cierto grado de independencia entre las características que producen la heterogeneidad, no todas las instrucciones se ven influenciadas en la misma proporción. Es decir, una operación de punto flotante ejecutada en el core A, puede tardar la cuarta parte del tiempo que en el core B, mientras que en el caso de una operación de escritura puede tardar la mitad al ejecutarse.

Para analizar el rendimiento logrado por las aplicaciones paralelas sobre estas arquitecturas, se utilizan métricas tradicionales: speedup y eficiencia [DON02].

El speedup (S) es una medida que permite cuantificar el beneficio relativo de resolver un problema en paralelo, es decir cuan "rápido" ejecuta el algoritmo paralelo respecto al secuencial. La función de speedup se define como la relación entre el tiempo del mejor algoritmo secuencial sobre un simple núcleo (Ts) y el tiempo requerido para resolver el mismo problema en una arquitectura paralela [GRA03].

$$S = \frac{TS}{TP}$$

La eficiencia (E) es una medida de la fracción de tiempo para la cual los cores son usados útilmente en la aplicación paralela. Esta métrica se define como la relación entre el speedup logrado y el speedup óptimo (S_{opt}) que se puede conseguir en la arquitectura [GRA03].

$$E = \frac{S}{S_{opt}}$$

Tradicionalmente, en las arquitecturas homogéneas, el speedup óptimo está dado por la cantidad de cores utilizados (p). Sin embargo, en el caso de las arquitecturas heterogéneas, se deben tener en cuenta la potencia de cómputo de los diferentes núcleos que la componen, por lo cual se redefine el speedup óptimo como:

$$S_{opt} = \sum_{i=0}^{p-1} \frac{Potencia\,(Core_i)}{Potencia\,(Mejor\ Core)}$$

3.2- Tratamiento de la heterogeneidad en CMRE.

La potencia de cómputo de un core está dada por el clock del procesador, en el caso

del multicore heterogéneo, se requiere que cada núcleo tenga uno propio. Esta situación se modela en el CMRE teniendo un clock general del sistema para la simulación (que es el más rápido) y múltiplos de ese clock para los robots. Esto se puede definir como relojes virtuales de cada robot/procesador.
Las operaciones de Depositar y Recoger se pueden asimilar a las de Write y Read en procesadores reales. Naturalmente una arquitectura heterogénea puede tener tiempos diferentes, que en el caso de CMRE serán múltiplos del clock general.
Asimismo hay que considerar los tiempos de la comunicación (Send y Receive) e incluso de operaciones tales como Informar que pueden tener distinto número de ciclos de reloj por procesador/robot.
Dentro del entorno CMRE, se le podrá configurar a cada robot una magnitud de procesamiento relativa a la velocidad patrón: “n veces más lento que”. Además permitirá definir cuantos ciclos de reloj consumirá cada una de sus instrucciones, es decir, configurar la heterogeneidad según la necesidad del ejercicio y del concepto que sea necesario brindar.
Es de hacer notar que para la introducción del concepto, se pueden definir diferencias simples en los relojes virtuales y posteriormente en el costo de ciclos de reloj en las operaciones por robot. Separar los dos aspectos ayuda al alumno a comprender que los tiempos tienen que ver no sólo con el procesamiento, sino también con los accesos a memoria y la comunicación.

3.3- Heterogeneidad y Balance de carga.

El balance de carga de una aplicación incide directamente en el rendimiento de la misma. Esto se debe a que si el sistema se encuentra desbalanceado, significa que hay cores que se encuentran ociosos esperando a que otros terminen de trabajar, incrementando el tiempo final de ejecución, y por consiguiente disminuyendo el speedup y la eficiencia. Esto depende de las características de la aplicación y de la arquitectura paralela que se esté utilizando.
Desde el punto de vista de la aplicación, en el caso del entorno CMRE, dependerá mucho de la actividad que cada robot desempeñe en su algoritmo, dado que, según las instrucciones que ejecute y también de la configuración particular de la ciudad, el trabajo podrá o no estar balanceado. Por ejemplo, si los robots deben juntar los papeles de un área privada (de igual tamaño para todos los robots), el balance de carga dependerá de la cantidad de papeles que exista en cada área.
Teniendo en cuenta la arquitectura paralela, la heterogeneidad de los cores es un factor fundamental que incide en el balance de carga. Por ejemplo, si dos robots deben recorrer una avenida de principio a fin, pero tienen diferentes velocidades, es indudable que el más rápido quedará ocioso hasta que el otro termina, lo cual afecta al balance de carga.
Si combinamos ambos aspectos, el problema del balance de carga se vuelve aún más complejo pero también más desafiante para el alumno a la hora de resolver un algoritmo que sea eficiente, lo que implica una motivación extra a la hora de implementarlo.

3.4- Ejemplo de un caso experimental sencillo para mostrar el impacto de la heterogeneidad.

Se tienen 4 robots con velocidades relativas de 1 paso por cuadra, 2 pasos por cuadra, 3 pasos por cuadra y 4 pasos por cuadra. Esto significa que el robot más rápido (R1) hará por ejemplo 100 cuadras de una Avenida en 100 unidades de tiempo y el más lento (R4) en 400 unidades de tiempo.
Si se tienen 4 Avenidas (Av1, Av2, Av3 y Av4) a recorrer totalmente (1 por robot), con una distribución de objetos a recoger diferente en cada una, por ejemplo 10, 20, 40 y 80 objetos respectivamente.
El tiempo de recoger un objeto 3T (donde T es el tiempo de dar un paso cada robot).

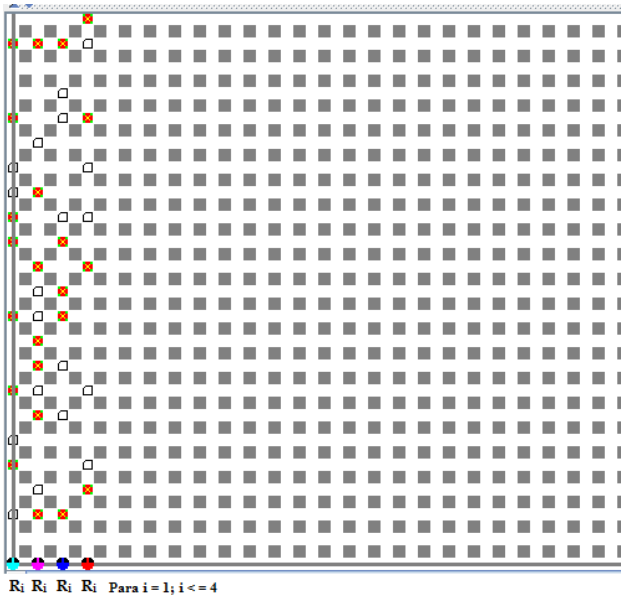


Fig. 1. Esquema del ejemplo a desarrollar.

La asignación del trabajo (en este caso la Avenida a recorrer) será significativa para el tiempo total del programa y para el desbalance de carga:

a- Si los robots los asignamos en forma directa relacionando la velocidad de los mismos con las Avenidas, según el trabajo esperado, se tiene:
   R1 recorre la Av4 y tarda 100 + 80x3 = 340 T
   R2 recorre la Av3 y tarda 200 + 40x3 = 320 T
   R3 recorre la Av2 y tarda 300 + 20x3 = 360 T
   R4 recorre la Av1 y tarda 400 + 10x3 = 430 T

   El programa tardaría 430 T y el máximo tiempo ocioso sería para R2 (110 T).
   La ociosidad total (que es una medida del desbalance de carga) sería de 270 T.

b- Si asignáramos los robots en forma inversa relacionando la velocidad de los mismos con las Avenidas, según el trabajo esperado, se tiene:
   R1 recorre la Av1 y tarda 100 + 10x3 = 130 T

R2 recorre la Av2 y tarda 200 + 20x3 = 260 T
R3 recorre la Av3 y tarda 300 + 40x3 = 420 T
R4 recorre la Av4 y tarda 400 + 80x3 = 640 T

Notar que en este caso el programa tardaría 640 T y el máximo tiempo ocioso sería para R1 (510 T).
La ociosidad total crecería a 1110 T ➔ Más de 4 veces el desbalance de la distribución anterior.

4 Consumo energético en algoritmos paralelos

El consumo energético es un punto clave en los procesadores actuales. En general la performance de un algoritmo paralelo no se mide sólo por el tiempo de ejecución del mismo, sino también por la energía consumida. Así aparecen índices que relacionan Flops/Watt o Flops/Joule según se relacione el cómputo con potencia instantánea o energía total [BAL13][BRO10].
Resulta importante en la formación de los alumnos de Informática hacer hincapié en las métricas de consumo como un indicador de calidad de los algoritmos. Asimismo, comprender los mecanismos automáticos que desarrollan los procesadores en función de la temperatura que alcanzan (que es una función directa de la energía consumida en un intervalo de tiempo) [BAL13].

4.2- La limitación del clock de los procesadores y su modelización en CMRE

En general todos los procesadores modernos tienen una curva de ajuste automático del clock en función del consumo (o de la temperatura interna que alcanzan) que sigue la forma de la Figura 2.

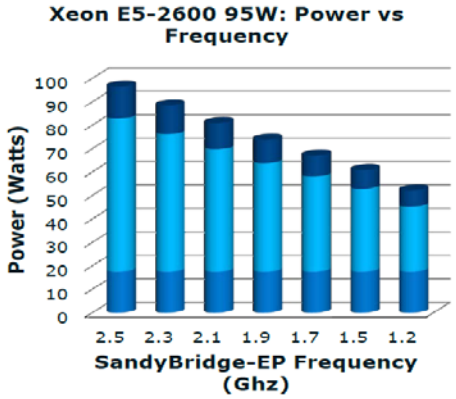


Fig. 2. Gráfico representativo de la curva de ajuste automático del clock.

En determinado punto el reloj se “enlentece” para disminuir el trabajo realizable por el procesador en cada unidad de tiempo y de ese modo alcanzar un punto de equilibrio con menor temperatura. En algunos casos es posible manejar la reasignación de tareas del procesador (por ejemplo controlando información de los contadores de hardware o mediante monitoreo directo), para evitar el cambio de la frecuencia de clock o para adaptarlo al cambio de frecuencia [BAL13].

Modelizar esto en CMRE requiere tener una medida de energía consumida en las operaciones (Mover / Recoger / Depositar / Comunicarse / Informar) y contabilizar la acumulación de energía de cada robot/procesador para ajustar su “velocidad” de acuerdo a un modelo previamente definido.
El esquema se asimila al de la Figura 2 donde se muestra una modelización discreta de la curva mostrada para procesadores reales.

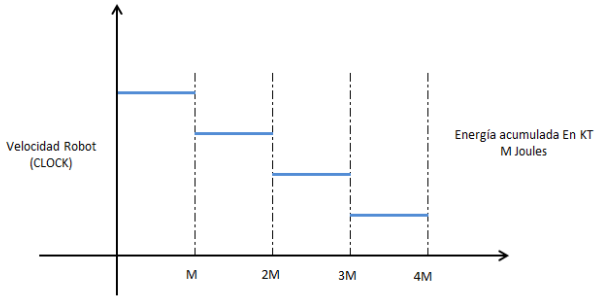


Fig. 3. Curva de ajuste automático de la “velocidad” de los robots.

Básicamente en CMRE esto significa “cambiarle” el reloj virtual al robot ajustando la velocidad de movimiento (y por ende la capacidad de trabajo por unidad de tiempo) de acuerdo a una métrica de energía gastada en las últimas K unidades de tiempo.
A los fines didácticos, se han desarrollado ejemplos con K=10 simplemente para considerar un período de acumulación de trabajo que genere cambios en la temperatura interna del procesador.
Con estas ideas se define un consumo energético por operación, partiendo de que MOVER requiere 1 Joule y se le da un costo energético a DEPOSITAR, RECOGER, ENVIAR, RECIBIR e INFORMAR. Con esto es sencillo computar la energía total de cada robot/procesador para un dado algoritmo, y también se pueden comparar algoritmos desde el punto de vista energético.
Es más complejo en el simulador adoptar una estrategia de “cambio de clock” en función del consumo. Actualmente se ha definido que si la energía consumida en los últimos 10 T de un robot cualquiera supera los M Joules (donde M es un parámetro que se fija según la curva indicada en la Figura 3, a ese robot se le reduce un escalón la velocidad (clock) por un mínimo de R ciclos (en este momento se utiliza R=3) y luego se reinicia el cálculo. Si el robot pasa a estar por debajo de los M Joules en los últimos 10 T, recupera su clock anterior.
Naturalmente esto implica diferente speedup, diferente eficiencia y diferente consumo máximo por intervalo de 10 T, mirando el algoritmo paralelo que se esté ejecutando. Asimismo, se tiene una medida (virtual) del consumo total de un dado algoritmo.

4.3- Ejemplo de ejercitación experimental

Es relativamente fácil extender el ejemplo relacionado con heterogeneidad (sección 3.4) al caso de consumo y considerar el cambio de clock en función del mismo. Al ejemplo mencionado se agrega que el consumo de una operación de RECOGER es de 5 Joules y la de MOVER de 1 Joule.

a- Con la primera asignación de trabajo planteada en el ejemplo del punto 3.4 (a) se tiene:

R1 recorre la Av4 y tarda 100+80x3 = 340 T y consume 100+80x5 = 500 Joules.
R2 recorre la Av3 y tarda 200+40x3 = 320 T y consume 200+40x5 = 400 Joules.
R3 recorre la Av2 y tarda 300+20x3 = 360 T y consume 300+20x5 = 400 Joules.
R4 recorre la Av1 y tarda 400+10x3 = 430 T y consume 400+10x5 = 450 Joules.

El programa tarda 430 T y consume en total 1750 Joules.
Es posible que si se aplica la restricción del reloj, posiblemente a R1 o R2, alguno de los robots DEBA disminuir su velocidad (según la distribución de la carga) y en ese caso será *constante el consumo total, pero el programa tardará más y puede cambiar el desbalance de carga*.

b- En la asignación inversa del ejemplo del punto 3.4 (b) el consumo sería:

R1 recorre la Av1 y tarda 100+10x3 = 130 T y consume 100+10x5 = 150 Joules.
R2 recorre la Av2 y tarda 200+20x3 = 260 T y consume 200+20x5 = 300 Joules.
R3 recorre la Av3 y tarda 300+40x3 = 420 T y consume 300+40x5 = 500 Joules.
R4 recorre la Av4 y tarda 400+80x3 = 640 T y consume 400+80x5 = 800 Joules.

Notar que en este caso el programa tardaría 640 T y consumiría en total la misma energía (1750 Joules). Esto es razonable porque hay un trabajo total igual en operaciones de MOVER y DEPOSITAR.
Sin embargo es menos probable que haya que reducir el clock de cualquiera de los robots…

5 Conclusiones y Líneas de Trabajo Futuro

Los temas de heterogeneidad y consumo energético en arquitecturas paralelas son de gran importancia y se ha presentado la modelización de los mismos sobre el entorno CMRE.
CMRE aparece como una herramienta muy útil para la introducción de estos conceptos, a partir de su empleo desde 2013 con robots/procesadores homogéneos.
Si bien hay modificaciones que se deben realizar en el entorno, resultan transparentes al alumno y la realización de trabajos experimentales orientados a estos temas es muy natural.
Actualmente se está trabajando en la implementación para 2016 de estas extensiones, considerando que al habilitar diferentes relojes para los diferentes robots, aparecen recorridos en los que los conflictos de concurrencia se pueden dar no sólo en las esquinas de la ciudad, sino también en puntos intermedios entre dos esquinas. De esta manera, el nivel de complejidad de los posibles escenarios aumenta, planteando un desafío mucho más ambicioso, que responde a la realidad tecnológica de los procesadores actuales.

6 Bibliografía

[ACM04] ACM/IEEE-CS Joint Task Force on Computing Curricula. “Computer Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering”. Report in the Computing Curricula Series. 2004.

[ACM08] ACM/IEEE-CS Joint Interim Review Task Force. “Computer Science Curriculum 2008: An Interim Revision of CS 2001”. Report from the Interim Review Task Force. 2008.
[ACM13] ACM/IEEE-CS Joint Task Force on Computing Curricula. “Computer Science Curricula 2013”. Report from the Task Force. 2013.
[ACM68] ACM Curriculum Committee on Computer Science. “Curriculum ‘68: Recommendations for the undergraduate program in computer science”. Communications of the ACM, 11(3):151-197. 1968.
[ACM78] ACM Curriculum Committee on Computer Science. “Curriculum ‘78: Recommendations for the undergraduate program in computer science”. Communications of the ACM, 22(3):147-166. 1979.
[ACM99] ACM Two-Year College Education Committee. “Guidelines for associate-degree and certificate programs to support computing in a networked environment”. New York: The Association for Computing Machinery. 1999.
[AMD09] AMD. “Evolución de la tecnología de múltiple núcleo”. http://multicore.amd.com/es-ES/AMD-Multi-Core/resources/Technology-Evolution. 2009.
[AND00] Andrews G. “Foundations of Multithreaded, Parallel, and Distributed Programming”. Addison Wrsley, 2000.
[BAL13] Balladini J., Rucci E., De Giusti A., Naiouf M., Suppi R., Rexachs D., Luque E. “Power Characterisation of Shared-Memory HPC Systems”. Computer Science & Technology Series – XVIII Argentine Congress of Computer Science Selected Papers. Págs. 53-65. 2013.
[BRO10] Brown D. J., “Toward Energy-Efficient Computing”, Magazine Communications of the ACM Volume 53 Issue 3, March 2010
[DAS89] Dasgupta S. “Computer Architecture. A Moder Synthesis. Volume 2: Advanced Topics”. Jhon Wilet & Sons. 1989.
[DEG14a] De Giusti, L., Leibovich, F., Sanchez, M., Chichizola, F., Naiouf, M., De Giusti, A. “Desafíos y herramientas para la enseñanza temprana de Concurrencia y Paralelismo”. Congreso Argentino de Ciencias de la Computación (CACIC), 2014.
[DEG14b] De Giusti, A., De Giusti L., Leibovich, F., Sanchez, M., Rodriguez Eguren, S. “Entorno interactivo multirrobot para el aprendizaje de conceptos de Concurrencia y Paralelismo”. Congreso Tecnología en Educación, Educación en Tecnología. 2014.
[DIJ78] Dijkstra E. W. “Finding the Correctness Proof of a Concurrent Program”. In Program Construction, International Summer Schoo, Friedrich L. Bauer and Manfred Broy (Eds.). Springer-Verlag, 24-34, 1978.
[DON02] Dongarra J. , Foster I., Fox G., Gropp W., Kennedy K., Torzcon L., White A. “Sourcebook of Parallel computing”. Morgan Kaufmann Publishers 2002. ISBN 1558608710.
[GEP06] Gepner P., Kowalik M.F. “Multi-Core Processors: New Way to Achieve High System Performance”. In: Proceeding of International Symposium on Parallel Computing in Electrical Engineering 2006 (PAR ELEC 2006). Pags. 9-13. 2006.
[GRA03] Grama A., Gupta A., Karpyis G., Kumar V. “Introduction to Parallel Computing”. Pearson – Addison Wesley 2003. ISBN: 0201648652. Segunda Edición (Capítulo 3).
[HAN77] Hansen P. B. “The Architecture of Concurrent Processes”. Prentice Hall, 1977.
[HOA85] Hoare C. “Communicating Sequential Processes”. Prentice Hall, 1985.
[HOO13] Hoonlor A., Szymanski B. K., Zaki M. J., Thompson J. “An Evolution of Computer Science Research”. Communications of the ACM. 2013.
[MCC07] Mc Cool M, “Programming models for scalable multicore programming”, 2007, http://www.hpcwire.com/features/17902939.html.
[MCC08] McCool M. “Scalable Programming Models for Massively Parallel Multicores”. Proceedings of the IEEE, 96(5): 816–831, 2008.
[NJS15] https://nodejs.org/api/http.html

V ETHICOMP Latinoamérica

7460 | Caracterización del Nivel de Reputación de los Líderes Políticos a través del Proceso de Análisis Jerárquico
Fernando Bircher (UCSE)

7651 | Análisis de los modelos de Gobierno de Tecnologías de la Información y sus relaciones con el Modelo de Excelencia Iberoamericano
Vicente Merchan Rodriguez (ESPE),
Rocío Rodriguez (UNLaM)

Caracterización del Nivel de Reputación de los Líderes Políticos a través del Proceso de Análisis Jerárquico

Fernando Javier Bircher

Universidad Católica de Santiago del Estero, Departamento Académico Rafaela
Bv. Irigoyen 1502 (2300) Rafaela, Santa Fe, Argentina
fer.bircher@gmail.com

Resumen. La reputación se ha convertido en un tema de gran relevancia para el sector público y, en particular, para el seguimiento de la reputación de candidatos y partidos políticos en el entorno online. Sin embargo, la forma en la que la reputación política debe ser analizada, medida y administrada no está tan clara. Es por ello que en este estudio, tras proporcionar una visión general de la bibliografía publicada en materia de política, toma de decisiones y reputación, se presentan los resultados de la caracterización de la reputación online de dos líderes políticos que se han postulado como Candidatos a Gobernador de la provincia de Santa Fe (Argentina) en las Elecciones Primarias, Abiertas, Simultáneas y Obligatorias del 19 de abril de 2015. Para ello, se ha realizado un seguimiento de estos políticos acorde a un conjunto de lineamientos con el objetivo de proporcionar un retrato estadísticamente fiable y políticamente coherente.

Palabras Clave: Reputación. AHP. Redes Sociales. Políticos.

1 Introducción

Desde que a finales de la década de los noventa se popularizase en las sociedades occidentales el uso de Internet, su importancia no ha dejado de crecer. De acuerdo a Internet World Stats [1] existen aproximadamente 3.035.749.340 de internautas en todo el mundo. De todos ellos, 630 millones proceden de América, de los cuales 230 millones son de Sudamérica, y dentro de Sudamérica 32.268.280 son de Argentina, que representan el 77,9% de la población nacional, una cifra que previsiblemente irá en aumento durante esta década.

La Web se ha consolidado como un instrumento de búsqueda de información para los usuarios, que acuden a ella con intereses fundamentalmente lúdicos y comerciales.

En cuanto a los usos políticos de Internet, en los que englobaremos la información, la comunicación, la movilización y la participación, son minoritarios, aunque con una clara tendencia al alza [2]. La comunicación política a través de Internet ha cobrado una gran importancia en estos últimos años, especialmente desde la victoria de Barack Obama en las presidenciales norteamericanas de 2008. La experiencia de esta campaña electoral y la de las otras que la han seguido ha demostrado que las

posibilidades que ofrece Internet en términos de información y movilización política son innumerables.

Siguiendo con esta línea, la autora afirma que no todo son beneficios, ya que en este mundo cada día más globalizado y saturado de información, la reputación de los políticos se ha revelado mucho más frágil. El desarrollo en el último lapso de sistemas de medición y gestión de la reputación online para empresas y marcas ha despertado el interés de un gran número de teóricos y partidos políticos por establecer un modelo de medición de la reputación online adecuado al ámbito de la política.

El objetivo de este estudio consiste en diseñar un marco de trabajo basado en el Proceso de Análisis Jerárquico que permita determinar el nivel de reputación de una persona o entidad en función de las condiciones o características esenciales que se visualizan en Facebook y en Twitter, las dos redes sociales virtuales más importantes de Argentina.

2 Desarrollo

2.1 Redes Sociales

El concepto de redes sociales precede a Internet y a las herramientas tecnológicas, aunque el término no fuera popular en el pasado. Una red social es un conjunto de entidades que se conectan entre sí por alguna relación particular: parentesco, amistad, trabajo, casualidad; y dónde esas relaciones pueden ir evolucionando a otros estados. Generalmente esas entidades son personas u organizaciones [3].

Para este trabajo se asume como concepto que una red social es un conjunto de personas que se relacionan entre sí de manera organizada, focalizado en el ambiente de las tecnologías de la información y la comunicación para trabajar sobre las redes sociales virtuales, es decir, aquellas que se desarrollan sobre Internet.

Para caracterizar la reputación online de los líderes políticos en las redes sociales virtuales el análisis estará enfocado en Facebook y en Twitter.

Facebook es llamada la “red social de gran consumo” porque es la red social más popular y conocida, por lo tanto los líderes políticos utilizan las páginas de fans para llegar a un gran número de personas.

Mientras que Twitter, es la red social donde los políticos deben estar. Esto se debe a que a través de esta red pueden compartir, conversar, escuchar, reducir el tiempo de espera y crear una relación más directa con el ciudadano. De hecho, conscientes de la capacidad vírica que posee esta red, la mayoría de los dirigentes políticos poseen cuentas oficiales.

2.2 Reputación Online

El crecimiento y penetración de uso de los medios y contenidos generados por los usuarios ha creado un nuevo contexto, global y local al mismo tiempo, para la comunicación, la participación y el intercambio de información.

La web 2.0 se presenta como un factor acelerador y amplificador de la extraordinaria habilidad de las personas para comunicar significados e ideas abstractas de forma social y colectiva y aporta la singularidad de que evoca, o engendra, maneras de interactuar que antes no eran posibles. Esta situación abre el camino para la elaboración de nuevas teorías o ampliaciones conceptuales como la reputación online.

La reputación online es posible identificarla, extraerla, clasificarla y analizarla, a partir de las opiniones que los usuarios diseminan en los sitios que facilitan las reseñas y opiniones sobre marcas, personas, productos y servicios de las organizaciones y sus competidores. Lo más característico de la reputación online, es que los sujetos expresan sus opiniones en la web 2.0 sin la obligación de ceñirse y someterse a cuestionarios estandarizados o guiones que restrinjan su experiencia a favor de las cuestiones decididas de antemano por un investigador [4].

Itoiz [5] expresa que no hay muchas definiciones del término reputación online. Según algunos autores, la reputación se vincula a las opiniones. Sin embargo, si se refiere a la reputación online únicamente como el seguimiento de palabras o temas en la red en las diferentes plataformas 2.0, es algo incompleto. Se debe entender la reputación de forma más amplia donde el ámbito online y offline se vinculen formando uno.

La reputación online se puede gestionar en diversos ámbitos, siendo uno de ellos, el ámbito político. La dimensión online de los partidos políticos y de los líderes políticos es cada vez más importante ya que aquellos que quieren ganar influencia están potenciando su presencia online y generando contenidos en una variedad de formatos. Otro aspecto importante, es que la reputación intenta guiar la toma de decisiones de un determinado agente para, entre otras cosas, fijar cómo, cuándo y con quién interactuar.

2.3 Toma de Decisiones Multicriterio

La toma de decisiones multicriterio es un problema crítico de la vida real, ya que cualquier actividad involucra de una u otra manera, la evaluación de un conjunto de alternativas en términos de un conjunto de criterios de decisión, donde muy frecuentemente estos criterios están en conflicto unos con otros [6]. Además, en la mayoría de las ocasiones el responsable de tomar una decisión se encuentra influenciado por sus patrones o modelos mentales, por la influencia de quienes se encuentran en una posición jerárquica superior o inferior, incluyéndose también el estado de ánimo y sus relaciones familiares y sociales, lo cual determina las prioridades al momento de abordar el problema, y añaden desde luego, mayores elementos de complejidad.

Por ello, es necesario contar con herramientas que permitan discernir sobre las alternativas, de manera que sea considerado el efecto de los múltiples criterios, y la solución responda a todos ellos de manera global.

Entre estas herramientas se encuentran los modelos de preferencia, es decir, herramientas que permiten abordar el problema de decisión multicriterio de una forma sistémica y científica, buscando favorecer el proceso y ayudar a quien toma la

decisión, y dentro de estos modelos, se hará referencia específica al Proceso de Análisis Jerárquico.

2.4 Proceso de Análisis Jerárquico (AHP)

El método AHP fue propuesto por el Doctor en Ciencias Matemáticas Thomas Saaty en la década del ochenta, para resolver el tratado de reducción de armamento estratégico entre Estados Unidos y la Unión Soviética.

El AHP desmenuza un problema en sub-problemas y luego une todas las soluciones de estos en una conclusión. El propósito del método es el de permitir que el decisor pueda estructurar un problema multicriterio de forma visual, mediante la construcción de un modelo jerárquico, que básicamente contiene tres niveles: meta u objetivo, criterios y alternativas.

Una vez construido el modelo jerárquico, se realizan las comparaciones por pares entre dichos elementos (criterios, sub-criterios y alternativas) y se atribuyen valores numéricos a las preferencias señaladas por los decisores, entregando una síntesis de las mismas mediante la agregación de esos juicios parciales.

El fundamento del proceso de Saaty descansa en el hecho que permite dar valores numéricos a los juicios dados por los decisores, logrando medir cómo contribuye cada elemento de la jerarquía al nivel inmediatamente superior del cual se desprende, llegando a obtener una síntesis como resultado final, que da una noción al decisor de la alternativa que debería de elegir, como también un análisis de la sensibilidad de los datos, para anticipar una posible variación en los juicios dados por los decisores [7].

Por lo tanto, para este trabajo se ha elegido al método AHP porque permite una organización jerárquica del problema en distintos niveles, cosa que no pueden hacer los métodos que exigen comparaciones globales de las alternativas; incluye en el método la asignación de pesos a los atributos, lo cual facilita la tarea de "extraer" de la mente del decisor esta información de una forma confiable; detecta y acepta, dentro de ciertos límites la incoherencia de los decisores humanos; no necesita información cuantitativa acerca del resultado que alcanza cada alternativa en cada uno de los criterios considerados, sino tan sólo los juicios de valor del centro decisor.

2.4.1 Expert Choice

Para el presente trabajo se utiliza el software Expert Choice, un sistema para el análisis, síntesis y justificación de decisiones y evaluaciones complejas supervisado por Thomas Saaty. Este software hace posible mirar los elementos de un problema en forma aislada: un elemento se compara contra otro con respecto a un criterio. Este es el proceso de decisión reducido a sus términos más sencillos: comparaciones apareadas [8]. El Expert Choice permite estructurar un problema, ordenarlo, sintetizarlo y combinar todos los juicios de modo de priorizar claramente las alternativas de mejor a peor.

2.4.2 Metodología del AHP

La técnica AHP se compone fundamentalmente de tres etapas [7].

1. Estructuración del modelo jerárquico: consiste en el análisis y la modelización del problema a resolver mediante una estructura jerárquica en la que se definen sus principales elementos como son el objetivo que se pretende alcanzar, los criterios y sub-criterios (optativos) que se emplearán para la evaluación de las diversas alternativas planteadas como soluciones factibles del problema.

2. Evaluación del modelo: consiste en la emisión de juicios de los expertos mediante comparaciones por parejas. Estos juicios, reflejan la importancia relativa, medida según la escala (Tabla 1) propuesta por Saaty [9], que para el experto representan los elementos de un nivel inferior con respecto a los elementos del nivel superior. Estos juicios se recogen en una serie de matrices de comparaciones por pares a partir de las cuales se obtienen las prioridades locales de cada elemento de la jerarquía.

3. Resultado final: se deduce una vez que se hayan producido todas las comparaciones por pares con sus respectivos juicios existentes, para luego hacer la síntesis del modelo jerárquico y en caso de existencia de variaciones en los juicios del decisor el análisis de sensibilidad.

Tabla 1. Escala de Preferencias de Saaty.

Escala Numérica	Explicación
1	Los dos elementos contribuyen igualmente a la propiedad o criterio.
3	El juicio y la experiencia previa favorecen levemente a un elemento frente al otro.
5	El juicio y la experiencia previa favorecen notablemente a un elemento frente al otro.
7	Un elemento domina fuertemente. Su dominación está probada en la práctica.
9	Un elemento domina al otro con el mayor orden de magnitud posible.
2,4,6,8	Valores intermedios entre dos juicios adyacentes. Se aplican cuando es necesario expresarlo.
Inverso de todos los valores anteriores	Si al comparar i con j se ha obtenido uno de los valores anteriores, al comparar j con i se le asigna el inverso.

2.5 Selección de un Líder Político mediante la Aplicación del Método AHP

2.5.1 Identificación del Problema

En primer lugar, los ciudadanos no tienen suficiente información como para evaluar a los políticos.

En segundo lugar, los líderes políticos se ven inducidos a escoger políticas que a su juicio habrán de ser evaluadas positivamente por los ciudadanos a la hora de la próxima elección.

Los dos puntos de vista son problemáticos, por lo tanto con esta aplicación del método AHP se brinda mayor información a los ciudadanos para evaluar a los líderes políticos y se permite a dichos líderes prever con mayor exactitud el juicio de los electores.

2.5.2 Definición del Objetivo

El objetivo genérico se define como: "Seleccionar el mejor líder político".

Mientras que el objetivo específico consiste en "Seleccionar el mejor líder político en la interna de los Candidatos a Gobernador del Frente Progresista Cívico y Social en las Elecciones Primarias, Abiertas, Simultáneas y Obligatorias (PASO) de la Provincia de Santa Fe (Argentina)".

2.5.3 Identificación de los Criterios

Los criterios que he propuesto para la evaluación de los líderes políticos son: "Presencia en Redes Sociales" y "Características Personales y Curriculares".

El primer criterio se compone por los siguientes subcriterios: "Nº de me gusta en fan page", "Nº de seguidores en perfil de Twitter", "Cantidad de me gusta, comentarios y shares en las publicaciones en Facebook", "Cantidad de retuits", "Nº de comentarios positivos en Facebook", "Nº de comentarios positivos en Twitter", "Nº de comentarios negativos en Facebook" y "Nº de comentarios negativos en Twitter".

Mientras que al segundo criterio, lo conforman los siguientes subcriterios: "Edad", "Género", "Trayectoria política", "Nivel de educación", "Propuestas" y "Partido político".

2.5.4 Identificación de las Alternativas

Las alternativas de decisión se enumeran de la siguiente manera: Candidato Político 1 y Candidato Político 2. Ambos candidatos son líderes políticos que tienen presencia en las redes sociales desde hace varios años, con cuentas activas en las cuales brindan información sobre sus propias gestiones. Mediante su presencia en Internet, buscan dar a conocer todo su potencial y conseguir un mayor número de personas interesadas en su capacidad de gobernar.

2.5.5 Evaluación del Modelo

El decisor hace uso de la información obtenida para emitir la opinión más consistente posible y así establecer las prioridades entre los criterios y/o alternativas.

Estos juicios se recogen en una serie de matrices de comparaciones por pares a partir de las cuales se obtienen las prioridades locales de cada elemento de la jerarquía.

El decisor cree, por ejemplo, que el Número de comentarios negativos en Facebook es más importante que el Número de fans de Facebook, por lo que se asigna un 5 a la posición (2,1) de la matriz que se visualiza en la Tabla 2. Como resultado de esta asignación, automáticamente se concluye que el Número de fans de Facebook es 1/5 de importante con respecto al Número de comentarios negativos en Facebook y este valor se asigna a la posición (1,2) de dicha matriz. Obsérvese que en la diagonal de esta matriz aparecen valores unitarios, ya que corresponde a la comparación de la importancia de un subcriterio sobre sí mismo.

Además, cada uno de los subcriterios afecta a la elección de cualquiera de las alternativas, por lo que se evalúa la importancia que tiene cada una de estas respecto de cada subcriterio. Nuevamente, para comparar las dos alternativas, en cada subcriterio, se emplea la Escala de Preferencias definida en la Tabla 1. Como ejemplo, en la Tabla 3 se muestra la matriz de comparación por pares que define la importancia que tiene cada alternativa con respecto al subcriterio que se está evaluando.

Tabla 2. Importancia de los subcriterios respecto a los criterios.

	N° de fans de Facebook	**N° de comentarios negativos en Facebook**
N° de fans de Facebook	1	1/5
N° de comentarios negativos en Facebook	5	1

Tabla 3. Importancia de las alternativas con respecto al Número de Fans de Facebook.

	Candidato Político 1	**Candidato Político 2**
Candidato Político 1	1	1/3
Candidato Político 2	3	1

2.5.6 Resultado Final

Una vez que el decisor haya producido todas las comparaciones por pares con sus respectivos juicios existentes se procede a realizar la síntesis del modelo jerárquico, para lo cual se ha utilizado el software Expert Choice.

2.5.6.1 Síntesis

Es la combinación de todos los juicios emitidos por el decisor, llegando a obtener como resultado un ranking de las alternativas que dicho decisor ha presentado.

La síntesis aplicada a este problema define como ganador al Candidato Político 2 con un 56,4% de aceptación, sobre el 43,6% obtenido por el Candidato Político 1.

2.5.6.2 Interpretación de los Resultados

Las prioridades que se han conseguido de la aplicación del modelo de decisión en el Expert Choice sugieren que el mejor líder político en las internas del Frente Progresista Cívico y Social en las Elecciones PASO de la Provincia de Santa Fe ha sido el Candidato Político 2, obteniendo una prioridad del 56,4%.

3 Conclusiones y Futuras Líneas de Investigación

3.1 Conclusiones

Internet, en un principio, y las redes sociales más tarde, han dejado claro que los medios o canales en los que la política se venía desarrollando no son los mismos ni siguen las mismas pautas. Lo que se sabía acerca de cómo implementar una eficaz comunicación política en campaña electoral ya no sirve, porque en Internet los patrones de comportamiento y difusión parecen no obedecer a las premisas, estudiadas y probadas, que se habían adquirido.

Por lo tanto, a partir del increíble protagonismo que han adquirido las redes sociales y todo lo relacionado con "ser social" nace el término Reputación Online. Este término hace referencia al conjunto de opiniones buenas y malas, de una persona, marca o empresa en Internet. Es decir, es todo aquello que se comenta por cualquier razón, en cualquier lugar de la red y por cualquier persona.

Medir esa Reputación Online no es algo excesivamente complicado, ya que se limita a escuchar todo el "ruido" presente en Internet y observar si lo que se dice sobre una persona u organización es bueno, o no, para, en los casos negativos poner solución o detectar porqué se produjeron y en los casos positivos, intentar que se produzcan de manera más frecuente. Todo este revuelo de las redes sociales y de la obsesión por "ser social" ha sido importante en diversos ámbitos, entre ellos, el ámbito político.

Los líderes políticos deberán concientizarse de que la manera de comunicarnos ha cambiado. En un futuro cercano, todos aquellos líderes políticos que quieran mantener su vida profesional, deberán incluir algún perfil en las redes sociales y sobre todo mantener ese perfil actualizado. Sin embargo, tener únicamente presencia no será suficiente, sino que se deberá medir, gestionar y analizar su imagen en Internet, para conseguir una buena Reputación Online.

En referencia al objetivo perseguido, se ha diseñado un marco de trabajo basado en el Proceso de Análisis Jerárquico que permite determinar el nivel de reputación de un líder político en función de las condiciones o características esenciales que se visualizan en las redes sociales. En este marco de trabajo se ha establecido una ponderación específica para cada uno de los criterios y subcriterios utilizados, determinando así un orden de relevancia. Si bien podría argumentarse que los valores de ponderación incluyen un grado de subjetividad de aquel que los asigna, estos se han establecido teniendo en cuenta conclusiones de estudios publicados por diferentes autores.

De todos modos, los resultados obtenidos en la selección de las redes sociales y de los líderes políticos permiten determinar la potencialidad del AHP, el que con la ayuda del Expert Choice ha dado una solución novedosa e integrada a un problema de análisis de alternativas.

Considerando los Candidatos a Gobernador, se ha seleccionado al mejor líder político en las internas del Frente Progresista Cívico y Social en las Elecciones PASO de la Provincia de Santa Fe.

En coincidencia con los resultados obtenidos en este estudio, en el escrutinio definitivo de las PASO del 19 de abril, el Candidato Político 2 ha vencido al Candidato Político 1 con el 70,64% de los votos dentro del Frente Progresista Cívico y Social. Esta situación determina que el marco de trabajo que se ha propuesto también puede ser utilizado como método de predicción electoral, ya que las redes sociales se presentan como un reflejo de la sociedad en la que nos encontramos y además tienen la capacidad de influir con un efecto positivo en los resultados electorales, de manera tal, que los candidatos con mejor reputación online pueden aventajar a sus oponentes.

Puede que tengamos entre manos una buena fórmula, pero que sólo funciona bajo unas condiciones muy particulares y debe ser ajustada en cada elección. También es importante reconocer que no todas las franjas de edad son usuarios activos en estas redes sociales, apareciendo los mayores de 50 años como los menos propensos a participar y compartir así sus opiniones y tendencias de voto, y no encontrándose representados por tanto en estos canales de difusión. No obstante, la mayoría de los investigadores coinciden en no desestimar la capacidad de este tipo de análisis.

Por último, es importante que medios de comunicación como la televisión o la radio presten mucha atención también a la cantidad de seguidores que poseen las páginas de Facebook de los candidatos políticos, además de sus publicaciones, y realicen sondeos con estas cifras, ya que comunica que en los futuros procesos electorales es muy probable que las redes sociales jueguen un papel mucho más importante debido a que publicar por redes sociales reduce los gastos y permite crear ese vínculo más cercano entre el ciudadano y el candidato que va a hacer que la población sepa por quién quieren votar el día de elecciones.

3.2 Futuras Líneas de Investigación

Dadas las conclusiones encontradas en el presente trabajo, se hace necesaria la labor de nuevos investigadores que continúen perfeccionando las metodologías iniciadas incorporando sus propios métodos y aportando nuevas y decisivas conclusiones. Una propuesta de mejora en la investigación podría fijarse en los siguientes puntos:

• Las herramientas y los métodos de análisis, novedosos hasta el momento, deben ser testeados y continuamente mejorados para asegurar el veraz resultado de las informaciones desprendidas.

• Desarrollar un sistema automático para recoger, almacenar, analizar y visualizar de manera agregada información publicada en medios de comunicación sobre ciertas personas u organizaciones.

• La incorporación al análisis de otras redes sociales, como Instagram, tratando de encontrar similitudes y divergencias entre sus resultados, siendo un mismo proceso electoral, el ámbito de estudio para todas ellas.

• Categorizar a los participantes de las redes sociales, definiendo su sesgo político y su volumen de actividad.

• Tomar a un actor influyente de las redes sociales (que no pertenezca al equipo de campaña pero que se tome por influyente en ese tema) y seguir el flujo de la difusión para analizar su patrón de conducta.

De esta manera, se abre un campo de especial interés y relevancia de cara al futuro. Campo en el que esta investigación espera haber contribuido con sus resultados, pretendiendo un desarrollo de metodologías de análisis más sofisticadas.

Referencias

1. Internet World Stats. (30 de Junio de 2014). Recuperado el 23 de Mayo de 2015, de http://www.internetworldstats.com/
2. Bravo Galán, M., "Desarrollo de un índice de reputación política online. Un estudio piloto aplicado a las elecciones autonómicas y municipales españolas del 22 de mayo de 2011", Madrid, Universidad CEU San Pablo, 2011.
3. Zenobi, R., Balbiano, M., Karchesky, D., Bircher, F. y Bressán, M., "Caracterización de los perfiles biográficos digitales en Facebook de adolescentes de Rafaela y Sunchales", Rafaela, Argentina, 2014.
4. Del Fresno García, M., "Infosociabilidad: monitorización e investigación en la web 2.0 para la toma de decisiones", El Profesional de la Información, Vol. 20, Nº 5, pp. 548-554, 2011.
5. Itoiz, M., "En la era digital, ¿reputación o reputación online?", España, 2014.
6. García Gómez, P., "Una propuesta metodológica para la aplicación del Proceso Analítico Jerárquico en la selección de aerogeneradores", Madrid, Universidad CEU Cardenal Herrera, 2010.
7. Maurtua Ollaguez, Diego E., "Criterios de Selección de Personal mediante el uso del proceso de análisis jerárquico. Aplicación en la selección de personal para la Empresa Exotic Foods S.A.C.", Lima, Universidad Nacional Mayor de San Marcos, 2006.
8. Paredes González, J., "Toma de decisiones. Proceso Analítico Jerárquico aplicado a la selección de herramienta de educación virtual", FPUNE Scientific, Nº 6, pp. 95-102, 2014.
9. Saaty, Thomas L., "Decision making with the analytic hierarchy process", International Journal of Services Sciences, Vol. 1, Nº 1, pp. 83-98, 2008.

Análisis de los modelos de Gobierno de Tecnologías de la Información y sus relaciones con el Modelo de Excelencia Iberoamericano

Vicente Merchán Rodríguez[1], Rocío Rodríguez[2]

[1] Departamento de Ciencias de la Computación
Universidad de las Fuerzas Armadas ESPE
Av. Gral. Rumiñahui s/n, Sangolquí, Ecuador
vrmerchan@espe.edu.ec

[2] Departamento de Ingeniería e Investigaciones Tecnológicas
Universidad Nacional de La Matanza - UNLaM
Florencio Varela 1903, San Justo, Buenos Aires, Argentina
rrodriguez@ing.unlam.edu.ar

Resumen. En los últimos años se han realizado múltiples esfuerzos en mejorar el entendimiento sobre los beneficios que genera el uso de las Tecnologías de la Información (TI) al interior y exterior de las organizaciones privadas y públicas, principalmente desde el punto de vista estratégico en donde juega un rol clave el Gobierno de TI. Esto ha propiciado el aparecimiento de nuevas formas de gobernanza que apoyen a las organizaciones a adquirir habilidades sobre cómo obtener el mejor provecho en el uso de las TI. El presente artículo busca identificar los aspectos más relevantes de esos modelos de gobierno, partiendo del propio concepto hasta llegar a los estándares y marcos aplicables que permitirán visualizar de manera global los actuales esquemas de Gobierno de TI en la organización y que faciliten una aproximación para medir la excelencia basada en valor.

Palabras claves: Gobierno de TI, Marcos de Referencia, Valor, Modelo de Excelencia.

1 Introducción

El concepto de gobierno de empresa ha evolucionado en la última década a través de definiciones que se relacionan con la composición de dos dimensiones: Gobierno corporativo y gobierno de negocio [1].

En la figura 1 (adaptada de [1]) se muestra el gobierno de empresa, en donde se especifican los roles y los procesos en que se soporta. El gobierno corporativo se soporta en los procesos de conformidad [2] y el gobierno de negocio en los procesos de ejecución.

El panorama para las TI pretende ser claro a nivel de alcance [3], lo cual, se convierte en un reto ya que se debería tener formas de poder involucrar estratégicamente a las TI en el contexto empresarial, para minimizar el riesgo de una desarticulación con las actividades que demandan el uso de las TI y el cumplimiento de objetivos.

Gobierno de Empresa	
Gobierno Corporativo	**Gobierno de Negocio**
Procesos de conformidad	Procesos de ejecución
Presidente / CEO	Planificación Estratégica y Alineamiento
Directores no ejecutivos	Toma de decisiones estratégicas
Comité de Auditoría	Gestión Estratégica del Riesgo
Comité Remuneraciones	Cuadro de Mando Integral
Gestión del Riesgo	Sistemas estratégicos de la empresa
Auditoría Interna	Mejora continua global
Responsabilidad **Seguridad**	**Creación de valor** **Utilización de recursos**

Fig. 1. Gobierno de Empresa.

En la figura 2 se muestra el nivel de autoridad y responsabilidad del Gobierno de TI, tanto a nivel corporativo como a nivel de negocio. Para la administración de TI es necesaria la utilización de estándares y marcos de trabajo. Esto se debe al hecho de que las TI están presentes para el apoyo organizacional [3]. No obstante, el esquema mostrado plantea dos problemas: ¿Qué se debería hacer en las empresas para articular las actividades de TI con sus objetivos estratégicos? y, ¿Cómo los modelos de Gobierno de TI se encuentran relacionados con los criterios de excelencia con la finalidad de garantizar la entregar de valor?.

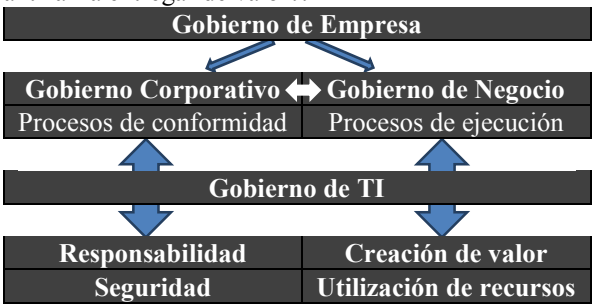


Fig. 2. Dimensiones del Gobierno de Empresa

Encontrar respuestas a los interrogantes planteados es la motivación de este análisis basado en la investigación documental y juicio de expertos. El resultado permitirá soportar a aquellos gobiernos de TI que buscan el cumplimiento con

excelencia en su rol estratégico apoyados en ISO/IEC 38500:2008, COBIT 5 y el Modelo Iberoamericano de Excelencia; principalmente en los aspectos que permitan precisar la entrega de valor, que constituye el objeto esencial que perciben los clientes y/o ciudadanos a cambio de lo que ellos entregan a la organización. Por lo tanto, la importancia de estos modelos de gobierno es mejorar su calidad [4] [5]. Ante este hecho, tenemos la oportunidad de que un modelo de excelencia, como el Modelo de Excelencia Iberoamericano, sea utilizado para medir la calidad, pudiendo ser aplicado a todo nivel dentro de la organización [6]. Por ejemplo, en el estratégico en donde el equipo de la alta dirección juega un papel importante en los planes de acción y toma de decisiones para la empresa.

Este artículo analisa compartivamente las últimas versiones de los modelos de Gobierno de TI, con perspectiva a definir las pautas para medir la calidad de sus procesos enfocados en valor.

Este documento se encuentra estructurado en secciones. En la sección 2 se introduce en los fundamentos. En la sección 3 se analiza las características. En la sección 4 se hace el análisis comparativo. En la sección 5 el análisis de las relaciones. Para culminar, en la sección 6 se presentan las conclusiones y trabajos futuros.

2 Fundamentos

Para un entendimiento claro y preciso, diversos autores, así como organizaciones, han formulado una serie de definiciones de Gobierno de TI. En este sentido, algunos autores y organizaciones importantes son: Van Grembergen [7]; Antonio Fernández & Faraón Llorens [8]; COBIT 5 [9], [10]; ISO/IEC 38500:2008 [11]; y, el Instituto Tecnológico de Massachusetts [1].

Por lo visto, el Gobierno de TI permite que surjan nuevas estructuras y procesos en los que otros puedan gestionar sus actividades de forma eficaz. Definiendo y difundiendo mecanismos de trabajo necesarios para asegurar que los objetivos de alineamiento de negocio y de TI se cumplan. Estas definiciones permiten fortalecer las aproximaciones al concepto de valor que se debe lograr, de manera articulada, en todos los niveles jerárquicos de una organización, y por consiguiente, en las áreas funcionales y de conocimiento que se agrupan en estos niveles.

No es desconocido que el valor se encuentra asociado con el uso de las TI en diferentes formas, generando beneficios tangibles e intangibles y que por lo tanto deben ser enfatizados y evaluados en el Gobierno de TI [12], [13], [14], [15], de manera permanente con el fin de generar acciones y tomar decisiones oportunas.

3 Análisis de las características

3.1 ISO/IEC 38500:2008

La norma proporciona principios que guían a los directores de las organizaciones en su responsabilidad con respecto al uso de las TI. Fué declarada norma internacional

en el año 2008, cuyo propósito es promover el uso eficiente, eficaz y aceptable de las TI en todas las organizaciones, ya sean públicas o privadas, entidades gubernamentales, entidades sin ánimo de lucro, entre otras; a través de:

- Una dirección que brinde confianza a las partes interesadas de la organización de TI, a través de la preparación e implementación de planes y políticas.
- Una evaluación del uso de las TI.
- Un monitoreo de todos los procesos que sirva para informar y orientar al gobierno corporativo sobre el uso de las TI en la organización y el cumplimiento de la dirección.

La norma asegura a las organizaciones que el seguimiento a los principios rectores ayudará a sus directores a sopesar los riesgos y fomentar oportunidades derivadas de la utilización de las TI.

Los principios rectores se muestran en la tabla 1. Cada uno es aplicable en la mayoría de organizaciones. Estos principios corporativos expresan preferentemente la conducta que guiará a los directores ejecutivos en la toma de decisiones.

Tabla 1. Principios del Estandar ISO/IEC 38500:2008

Principio	Descripción
Responsabilidad	Todas las personas deben comprender y aceptar sus responsabilidades en el contexto de TI.
Estrategia	La estrategia del negocio toma en cuenta las capacidades actuales y futuras de las TI. Los planes estratégicos de TI satisfacen las necesidades actuales y previstas derivadas de la estrategia del negocio.
Adquisición	Las adquisiciones de TI se hacen por razones válidas, en base a un análisis apropiado y continuo, con decisiones claras y transparentes. Hay un equilibrio adecuado entre beneficios, oportunidades, costes y riesgos tanto a corto como a largo plazo.
Rendimiento	Las TI están dimensionadas para dar soporte al negocio, proporcionando los servicios, niveles de servicio y con la calidad adecuada para cumplir con las necesidades actuales y futuras del negocio.
Conformidad	La función de TI cumple todas las regulaciones y normas aplicables. Las políticas y prácticas al respecto están claramente definidas, implementadas y exigidas.
Comportamiento humano	Las políticas de TI, prácticas y decisiones demuestran respeto al talento humano, incluyendo las necesidades actuales y emergentes de toda la gente involucrada.

El Gobierno de TI deberá demostrar el cumplimiento de los principios apoyado en un proceso de desarrollo clave para su éxito, este incluye las tres fases cíclicas de: Evaluar, Dirigir y Controlar o Monitorear (EDM). En donde el control, más allá de una ley, es la base de un buen gobierno porque inspira la confianza de los interesados o accionistas.

Cabe destacar que, la norma carece de métricas y mediciones para controlar el cumplimiento de metas y/o objetivos de TI.

3.2 COBIT 5

El cambio holístico en COBIT 5, lo ha llevado a constituirse en un marco propicio de negocio para el Gobierno y la Gestión de TI. Catalogado como un marco integral que apoya a las organizaciones a lograr sus metas y entregar valor mediante un gobierno y una gestión efectiva de las TI.

Los beneficios por los cuales se esfuerzan sus ejecutivos está en mantener la información con criterios de alta calidad para soportar las decisiones, generar valor con las inversiones en TI, alcanzar la excelencia operativa, controlar los riesgos, optimizar los costes de los servicios y tecnología de TI y, cumplimiento con las regulaciones y políticas aplicables.

La base del marco se encuentra compuesta por cinco principios básicos totalmente cubiertos e, incluye una guía exhaustiva sobre los denominados catalizadores para el gobierno y la gestión de TI. Estos principios se muestran en la tabla 2.

Tabla 2. Principios del Marco COBIT 5

Principios	Descripción
Satisfacer las necesidades de las partes interesadas	Las necesidades de los accionistas están alineadas con los objetivos empresariales, objetivos de TI y objetivos habilitadores, optimizando los recursos con un nivel aceptable de riesgos.
Cubrir la organización de forma integral	Tanto el Gobierno de TI como la Gestión de TI están cubiertos de una manera global, desde una perspectiva de extremo a extremo basándose en los 7 habilitadores de COBIT.
Aplicar un solo marco integrado	COBIT integra los mejores marcos de ISACA, permitiéndose alinear con los principales estándares: ITIL, TOGAF, PMBOK, PRINCE2, ISO y COSO.
Habilitar un enfoque holístico	COBIT identifica 5 habilitadores clasificados en 7 categorías que tienen como perspectiva a la empresa de extremo a extremo. Actúan como factores que influyen para que el gobierno y la gestión de TI operen de acuerdo a las necesidades del negocio.
Separar el Gobierno de la Gestión	COBIT distingue entre Gobierno y Gestión. Estas dos disciplinas engloban diferentes tipos de actividades, requieren diferentes estructuras organizativas y sirven a diferentes propósitos.

En la tabla anterior, se observa el quinto principio que de manera particular diferencia el gobierno y la gestión. Hasta la versión anterior de COBIT ambos conceptos se conjugaban en un solo proceso marco y de distintos dominios.

COBIT adopta los principios de ISO/IEC 38500:2008, siendo operativos a través de propuestas de procesos, metas y métricas de TI.

El Gobierno de TI, en línea con el estándar ISO/IEC 38500:2008, se apoya en un proceso de desarrollo compuesto por las tres fases cíclicas: EDM, las mismas que actúan sobre un conjunto de cinco procesos referenciales de gobierno que se encuentran en el modelo de procesos (incluyendo los procesos de valor).

4 Análisis comparativo

A modo de resumen, se presenta una comparativa de las principales características de cada modelo. Tal como se muestra en la tabla 3.

Tabla 3. Comparativa entre los Modelos de Gobierno de TI

	ISO/IEC 38500	COBIT
DEFINICIÓN	Define el *qué* hacer del buen gobierno	Define el *qué* y *cómo* lo debe hacer
ALCANCE	Tiene como alcance el *qué* del uso aceptable de las TI por parte de gobierno corporativo de TI	Tiene como alcance la integralidad representada en el *qué* del uso aceptable de las TI por parte del gobierno corporativo y el gobierno de gestión de TI.
AMBITO	Se aplican a todas las organizaciones, pequeñas o grandes, públicas o privadas, con fines y sin fines de lucro	se aplican a todas las organizaciones, pequeñas o grandes, públicas o privadas, con fines y sin fines de lucro
PRACTICAS	Evaluar, Dirigir y Monitorear.	Evaluar, Orientar y Supervisar.
BENEFICIOS	Genera beneficios prestando la debida atención al modelo y aplicando correctamente los principios	Genera beneficios en el soporte a las decisiones, valor en TI, alcance de excelencia operativa, control de riesgos, costes óptimos y, cumplimiento regulatorio y político.
PRINCIPIOS	6	5
METRICAS	No	Si
ENFOQUE	Actividades	Procesos

5 Análisis de relaciones

El procedimiento de construcción analítica sigue tres pasos: (1) Identificación y clasificación de criterios que caracterizan la excelencia basada en valor; (2) Relacionar los criterios de cada modelo de gobierno con los criterios de excelencia; y (3) Generar recomendaciones en función de las relaciones identificadas. Estas recomendaciones son insumos para que las organizaciones puedan autoevaluarse y mejoren las decisiones de Gobierno de TI.

Este análisis se soportó en el juicio de expertos del Modelo Iberoamericano de Excelencia para Administraciones Públicas [6].

El Modelo de Excelencia se compone de 5 procesos facilitadores y 4 criterios de resultados considerados efectos de los procesos facilitadores que se relacionan adecuadamente. Además, establece 20 subcriterios de excelencia, 4 por cada proceso

facilitador. Los 20 subcriterios facilitadores, producen efectos, a veces directos y de fácil relación y, otros indirectos y de difícil identificación. 9 de ellos resaltan características que se relacionan con la entrega de *valor* de la organización, área o servicios de TI y que son considerados importantes para obtener buenos efectos de *valor* percibido (ver Tabla 4).

Tabla 4. Modelo Iberoamericano - Facilitadores y Criterios de Calidad que promueven la entrega de Valor

Facilitadores	Criterios de Calidad
Liderazgo	**1a**: Se promueve de manera formal la definición de valores. Se modeliza el papel de la organización, con claros valores y expectativas, en línea con los principios de servicio público, la misión, visión y valores de la organización. **1b**: Se definen las actividades y/o procesos claves de la organización; en especial aquellos que generan mayor valor para el cliente y la misma organización, área o servicio.
Estrategia	**2b**: Se asume las responsabilidades y valores de la sostenibilidad en lo que se refiere a aspectos económicos, sociales y ambientales; así como legales. **2c**: Se evidencia el desarrollo de estrategias coherentes con la misión, visión y valores, y basada en las necesidades y expectativas de los grupos de interés y en información obtenida a través de actividades basadas en la investigación y la innovación.
Desarrollo de las personas	**3d**: Se promueven actividades sociales y culturales que identifican y aportan valor a la organización, área o servicio.
Recursos y asociados	**4b**: Se cultiva, desarrolla y protege la propiedad intelectual singular, con el fin de maximizar su valor para el cliente. **4d**: Se establecen mecanismos que permitan conocer y mejorar el involucramiento y resultados de los proveedores para maximizar el valor añadido de los mismos y su incorporación en la cadena de valor. Se estructuran las relaciones de asociación para crear una sinergia de valor.
Procesos y Clientes – ciudadanos	**5a**: Se evidencia un sistema que diseña, gestiona, mide y mejora los procesos y/o actividades orientado a aumentar el valor percibido por clientes y stakeholders **5c**: Se evidencia aumentos en las propuestas de valor y mejoramiento del ciclo de vida útil de los productos y servicios públicos, teniendo en cuenta su impacto en la salud pública, la seguridad y el medio ambiente.

En este momento, la decisión de haber identificado los 9 subcriterios de valor del modelo de excelencia asume que las relaciones congruentes con los criterios de los modelos de Gobierno de TI serán para contar con una aproximación que vincule los tres modelos a fin de simplificar la cultura de excelencia basada en valor en la administración pública.

La tabla 5 muestra el mapeo resultante entre ISO/IEC 38500:2008 y el Modelo de Excelencia (ver zona marcada con color gris). Con esto podemos tener una aproximación de alto nivel para evaluar el Gobierno de TI y medir la calidad basada en Valor. Los criterios 4d y 5c no identifican una relación de conveniencia con ISO/IEC 38500; así como, los principios de Compromiso y Comportamiento Humano de este con criterios del Modelo de Excelencia.

La tabla 6 muestra el mapeo resultante entre COBIT y el Modelo de Excelencia (ver zona marcada con color gris). Cada proceso de COBIT al menos mantiene una relación de conveniencia con un criterio de excelencia, no así los criterios de excelencia 4d y 5c que no mantienen relación con ningún proceso de COBIT.

Tabla 5. ISO/IEC 38500:2008 y Modelo de Excelencia.

ISO/IEC 38500	**MODELO EXCELENCIA**								
Tarea – Principio	**1a**	**1b**	**2b**	**2c**	**3d**	**4b**	**4d**	**5a**	**5c**
E. RESPONSABILIDAD	■				■				
D. RESPONSABILIDAD	■				■				
M. RESPONSABILIDAD	■				■				
E. ESTRATEGIA	■	■	■	■					
D. ESTRATEGIA	■	■	■	■					
M. ESTRATEGIA			■	■					
E. ADQUISICIÓN					■				
D. ADQUISICIÓN					■				
M. ADQUISICIÓN					■				
E. DESEMPEÑO	■	■	■	■		■		■	
D. DESEMPEÑO	■	■	■	■		■		■	
M. DESEMPEÑO	■	■	■	■		■		■	
E. CONFORMIDAD									
D. CONFORMIDAD									
M. CONFORMIDAD									
E. COM. HUMANO									
D. COM. HUMANO									
M. COM. HUMANO									

Table 6. COBIT 5 y Modelo de Excelencia.

COBIT	**MODELO EXCELENCIA**								
Procesos	**1a**	**1b**	**2b**	**2c**	**3d**	**4b**	**4d**	**5a**	**5c**
Asegurar el establecimiento y mantenimiento del marco de referencia de gobierno					■				
Asegurar la entrega de beneficios	■	■	■	■	■	■		■	

Asegurar la optimización del riesgo									
Asegurar la optimización de recursos									
Asegurar la transparencia hacia las partes interesadas									

Por lo analizado anteriormente, queda en evidencia de que existe un panorama oportuno para diseñar un modelo que evalúe la calidad en función de modelos que impulsan la mejora. Sin embargo, hay que tomar este trabajo como una primera aproximación válida que demandará ajuste en la medida en que se construye e implementa.

6 Conclusiones y trabajos futuros

Ambos modelos de gobierno, de manera independiente, afianzan las relaciones jerárquicas (estrategia, táctica y operación) de la organización a través de actividades y procesos de gobierno y gestión que permitan controlar los resultados de acuerdo a los intereses de valor perseguidos.

Lo interesante de este análisis es conocer que el Gobierno de TI debe demostrar la capacidad en la integración de TI a satisfacción de los objetivos estratégicos a través de la comunicación y medición, como factores claves de éxito.

Nuevamente, se confirma en estas versiones que ni ISO/IEC 38500:2008 ni COBIT 5 están diseñados para medir la calidad sino para mejorarla; en ese sentido es mejor utilizar un modelo adecuado como el Iberoamericano que ha sido diseñado exclusivamente para evaluar y medir la excelencia, propiciando planes de acción que aporten en la toma de decisiones de la alta dirección.

Los resultados, aquí expuestos, demuestran que efectivamente existen aspectos que identifican vínculos y que resultan ser adecuados entre los criterios vinculados en los tres modelos. Siendo así, se podría considerar que el siguiente paso a seguir sería la validación e identificación de mecanismos de medición e indicadores que permitan monitorear el cumplimiento de los criterios.

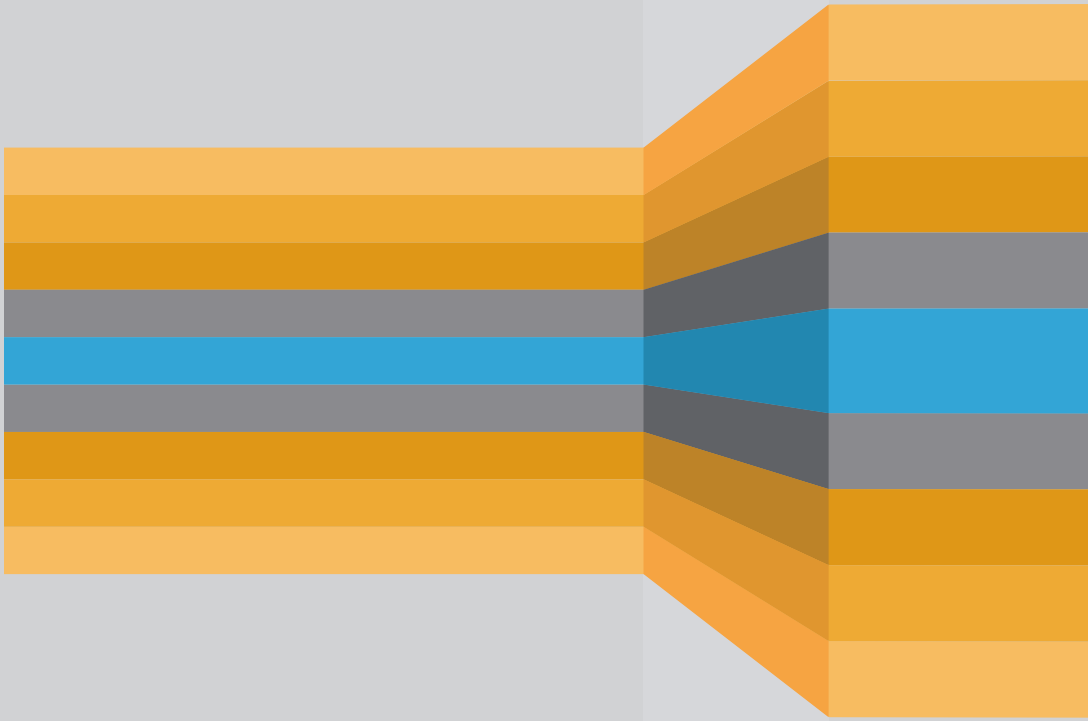
Si bien los criterios de excelencia que promueven valor permitirán identificar vínculos hacia un buen Gobierno de TI, queda la labor de establecer la manera de calcular técnicamente el valor asociado a cada requerimiento que tenga la organización y que deba cumplir el área de TI.

Referencias

1. P. Weill y J. Ross, IT Governance: How Top Performers Manage IT Decision Rights for Superior Results, Boston M.A.: Harvard Business Press, (2004).
2. OCDE, «Principios de Gobierno Corporativo,» Ministerio de Economía y Hacienda para la edición española, Madrid, (2004).

3. J. Carrillo, «Definiendo el alcance del gobierno de TI,» itSMF - Universidad UPM, 2009. [En línea]. Available: www.youtube.com/watch?v=xUL8IBalh9I&index=4&list=PL5DC4FE665E5C7D23. [Último acceso: 12 Abril 2014].
4. J. J. Sánchez Peña, «Evaluación de la calidad en el ámbito del Gobierno y la Gestión de Servicios de TI,» Alcalá de Henares, (2013).
5. C. Juiz, C. Guerrero y I. Lera, «Implementing Good Governance Principles for the Public Sector in Information Technology Governance Frameworks,» *Open Journal of Accounting*, vol. 3, pp. 9-27, (Enero 2014).
6. Fundación Iberoamericana para la Gestión de la Calidad, «FUNDIBEQ,» 2015. [En línea]. Available: http://www.fundibeq.org. [Último acceso: 2015].
7. W. Van Grembergen, S. De Haes y E. Guldentops, «Structures, Processes and Relational Mechanisms for IT Governance,» de *Strategies for information Technology Governance*, Hershey . London . Melbourne . Singapore, Idea Group Publishing, (2004), pp. 1-36.
8. A. Fernández Martínez y F. Llorens Largo, Gobierno de las TI para las universidades, Madrid: CRUE, (2012).
9. ISACA, COBIT 5: Un Marco de Negocio para el Gobierno y la Gestión de las TI de la Empresa, Quinta ed., Madrid: ISACA, (2012).
10. ISACA, COBIT 5: Procesos Catalizadores, Quinta ed., Madrid: ISACA, (2012).
11. Organización Internacional de Estándares (ISO) y la Comisión Electrotécnica Internacional (IEC), «ISO/IEC 38500:2008. Corporate Governance of Information Technology,» 2008. [En línea]. Available: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=51639. [Último acceso: 18 07 2014].
12. S. P. Robbins y M. Coulter, Administración, Octava ed., México: Pearson educación, (2005), p. 640.
13. W. Catton, Exploring Techniques for Measuring Human Values, Washington: American Sociological Review (19:1), (1954), pp. 49-55..
14. R. Keeney, Value-focussed thinking: a path to creative decision making, Cambridge:Massachusetts: Harvard University Press, (1992).
15. M. G. Piattini Velthuis y F. Hervada Vidal, Gobierno de las tecnologías y los sistemas de información, RA-MA 2007 ed., Madrid: RA-MA, (2007).

CACIC
2015
XXI



RedUNCI

ISBN 978-987-3724-37-4

CEDI - CENTRO DE EDICIÓN Y DISEÑO