

Editorial. Responsabilidad de las Facultades de Ingeniería Frente al Desarrollo del País
Germán López Quintero

Arquitectura para la Extracción de Información de Espacios Urbanos
Alejandro Forero Guzmán y Carlos Parra Rodríguez

Modelado Conceptual de la Supervisión de un Sistema Holónico de Producción Continua
Carlos Parra Ortega

Fuzzy Control Strategy Applied To Adjustment Of Front Steering Angle Of A 4wsd Agricultural Mobile Robot
Henry Borrero Guerrero, Rafael Bueno Sampaio y Marcelo Becker

Enterprise Mashups: A New Approach For Business Solutions
Mario Paredes Valverde y Giner Alor Hernández

Cloud Computing: A Review Of Paas, Iaas, Saas Services And Providers
María Salas Zarate y Luis Colombo-Mendoza

Entorno de Pruebas para Servicios Interactivos de Televisión Móvil Basada en el Estándar DVB-H
Gabriel Chanchí Golondrino, José Arciniegas Herrera y Enrique Herrera Viedma

Sistema Híbrido de Energía utilizando Energía Solar y Red Eléctrica
Jorge Díaz Rodríguez, Luis Pabón Fernández y Aldo Pardo García

Criterios para la Evaluación de Superficies Multitáctiles y sus Aplicaciones
Yenny Méndez Alegría, Cesar Collazos, Alfredo Sánchez, Luis Merchán Paredes, Yazmín Magallanes y Ariel Molina.

Sistema de Manipulación Remota de Robots Arduino por Medio de Servicios Web Semánticos
Jaime Guzmán-Luna, Juan Meza Gonzales y Eimer Castro Hincapié



FUNDACIÓN
UNIVERSITARIA



Facultad de
Ingenierías



©Fundación Universitaria Luis Amigó

Lámpsakos

Nº 7, enero-junio de 2012

ISSN: 2145-4086

Rector

Pbro. José Wilmar Sánchez Duque

Vicerrectora de Investigaciones

Isabel Cristina Puerta Lopera

Decano Facultad de Ingenierías

Jorge Mario Gaviria Hincapié

Jefe Departamento de Fondo Editorial

Carolina Orrego Moscoso

Diagramación

Carlos Zapata Sepúlveda

Terminación

Arbey David Zuluaga Yarce

Corrector de estilo

Juan Carlos Rodas Montoya

Contacto editorial

Fundación Universitaria Luis Amigó

Transversal 51A #67 B 90. Medellín, Antioquia, Colombia

Tel: (574) 448 76 66 (Ext. 9711. Departamento de Fondo Editorial)

www.funlam.edu.co - fondoeditorial@funlam.edu.co

Órgano de divulgación de la Facultad de Ingenierías de la Fundación Universitaria Luis Amigó.
Hecho en Colombia.

Financiación realizada por la Fundación Universitaria Luis Amigó.

LÁMPSAKOS

Directora de la Revista

Mg. Ingrid-Durley Torres Pardo

Comité Científico

Ph.D. Daniel Borrajo Millán, Universidad Carlos III de Madrid, España.

Ph.D. Amit Chaudhry, Panjab University, India.

Ph.D. Oscar Sapena, Universidad Politécnica de Valencia, España.

Ph.D. Rubén Ángel Galindo, Universidad Politécnica de Madrid, España.

Ph.D. Giner Alor Hernández, Instituto Técnico de Orizaba, México.

Mg. Julián Alberto Patiño, Universidad Nacional de Colombia, Colombia.

Mg. Néstor Raúl Londoño, Universidad Pontificia Bolivariana, Colombia.

Comité Editorial

Ph.D. Marcelo Becker, Universidad de São Paulo, Brasil.

Ph.D. Ulises Juárez Martínez, Instituto Técnico de Orizaba México.

Ph.D. Aldo Pardo García, Universidad de Pamplona, Colombia.

Ph.D. Alexander García Castro, Universidad Nacional de Colombia, Colombia.

Mg. Mauricio López Bonilla, Fundación Universitaria Luis Amigó, Colombia.

Mg. Francisco Javier Aguirre, Fundación Universitaria Luis Amigó, Colombia.

Árbitros

Ph.D. Jovani Alberto Jiménez, Universidad Nacional de Colombia, Colombia.

Ph.D. Jaime A. Guzmán-Luna, Universidad Nacional de Colombia, Colombia.

Mg. Armando Ordoñez, Universidad del Cauca, Colombia.

Mg. Julián Alberto Patiño, Universidad Nacional de Colombia, Colombia.

Mg. Pablo Andrés Deossa, Universidad Nacional de Colombia, Colombia.

Mg. Byron Portilla-Rosero, Fundación Universitaria Luis Amigó, Colombia.

Mg. Néstor Raúl Londoño, Universidad Pontificia Bolivariana, Colombia.

Mg. Gabriel Posada Hernández, Fundación Universitaria Luis Amigó, Colombia.

Mg. Carlos Durango Yepes, Fundación Universitaria Luis Amigó, Colombia.

Edición

Fundación Universitaria Luis Amigó

Solicitud de canje

Biblioteca Vicente Serer Vicens

Fundación Universitaria Luis Amigó

Medellín, Antioquia, Colombia

Para sus contribuciones

lampsakos@funlam.edu.co

Facultad de Ingenierías. Fundación Universitaria Luis Amigó. Transversal 51A #67 B 90. Medellín, Antioquia, Colombia

ISSN: 2145-4086

No. 7, enero-junio de 2012

Fundada en 2009

Lámpsakos es una publicación adscrita a la Facultad de Ingenierías de la Fundación Universitaria Luis Amigó, Medellín, Colombia. Tiene por objetivo divulgar artículos originales e inéditos como resultados de investigación científica o tecnológica de las áreas de las ingenierías. Su periodicidad es semestral con alcance nacional e internacional. Está destinada a estudiantes, investigadores, profesionales y personas interesadas en estos temas.

Los autores son moral y legalmente responsables del contenido de sus artículos, así como del respeto a los derechos de autor. Por lo tanto, éstos no comprometen en ningún sentido a la Fundación Universitaria Luis Amigó.

La reproducción de los artículos se regirá conforme a lo descrito en <http://creativecommons.org/>

CONTENIDO

Editorial. Responsabilidad de las Facultades de Ingeniería Frente al Desarrollo del País.....	7
<i>Germán López Quintero</i>	
Arquitectura para la Extracción de Información de Espacios Urbanos	9
<i>Alejandro Forero Guzmán y Carlos Parra Rodríguez</i>	
Modelado Conceptual de la Supervisión de un Sistema Holónico de Producción Continua	19
<i>Carlos Parra Ortega</i>	
Estrategia de Control Fuzzy Aplicada al Ajuste de Ángulo de Dirección Delantera de un Robot Móvil 4WSD Agrícola	31
<i>Henry Borrero Guerrero, Rafael Bueno Sampaio y Marcelo Becker</i>	
Empresa Mashup: Un Nuevo Enfoque para Soluciones de Negocio	39
<i>Mario Paredes Valverde y Giner Alor Hernández</i>	
Cloud Computing: Una revisión de los servicios y proveedores PaaS, IaaS, SaaS	47
<i>María Salas Zarate y Luis Colombo-Mendoza</i>	
Entorno de Pruebas para Servicios Interactivos de Televisión Móvil Basada en el Estándar DVB-H	59
<i>Gabriel Chanchí Golondrino, José Arciniegas Herrera y Enrique Herrera Viedma</i>	
Sistema Híbrido de Energía utilizando Energía Solar y Red Eléctrica	69
<i>Jorge Díaz Rodríguez, Luis Pabón Fernández y Aldo Pardo García</i>	
Críterios para la Evaluación de Superficies Multitáctiles y sus Aplicaciones	79
<i>Yenny Méndez Alegría, Cesar Collazos, Alfredo Sánchez, Luis Merchán Paredes, Yazmín Magallanes y Ariel Molina.</i>	
Sistema de Manipulación Remota de Robots Arduino por Medio de Servicios Web Semánticos	85
<i>Jaime Guzmán-Luna, Juan Meza Gonzales y Eimer Castro Hincapié</i>	

CONTENTS

Editorial. Responsibility of the Faculties of Engineering Development Front Country	7
<i>Germán López Quintero</i>	
Architecture for the Extraction of Information of Urban Spaces	9
<i>Alejandro Forero Guzmán y Carlos Parra Rodríguez</i>	
Conceptual Modeling of the Supervision of Continuous Production Holonic Systems	19
<i>Carlos Parra Ortega</i>	
Fuzzy Control Strategy Applied to Adjustment of Front Steering Angle of a 4WSD Agricultural Mobile Robot	31
<i>Henry Borrero Guerrero, Rafael Bueno Sampaio y Marcelo Becker</i>	
Enterprise Mashups: A New Approach for Business Solutions	39
<i>Mario Paredes Valverde y Giner Alor Hernández</i>	
Cloud Computing: A Review of PaaS, IaaS, SaaS Services and Providers	47
<i>María Salas Zarate y Luis Colombo-Mendoza</i>	
Test Environment for Interactive Mobile TV Services Based on DVB-H	59
<i>Gabriel Chanchí Golondrino, José Arciniegas Herrera y Enrique Herrera Viedma</i>	
Hybrid Power System Using Solar Energy and Power Line Supply	69
<i>Jorge Díaz Rodríguez, Luis Pabón Fernández y Aldo Pardo García</i>	
Evaluation Criteria for Multi-Touch Surfaces and their Applications	79
<i>Yenny Méndez Alegría, Cesar Collazos, Alfredo Sánchez, Luis Merchán Paredes, Yazmín Magallanes y Ariel Molina.</i>	
Remote Manipulation System of Arduino Robots Through Semantic Web Services	85
<i>Jaime Guzmán-Luna, Juan Meza Gonzales y Eimer Castro Hincapié</i>	

EDITORIAL

RESPONSABILIDAD DE LAS FACULTADES DE INGENIERÍA FRENTE AL DESARROLLO DEL PAÍS

J. Germán López Quintero*

Cuando se habla de responsabilidad de las facultades de Ingeniería, es porque se asume que una facultad, dentro de la Universidad, es el ente responsable de mantener una disciplina o su conocimiento en las fronteras de desarrollo del mismo y eso no se puede lograr si no hay un cuerpo docente de planta, con las más altas calidades y organizado en grupos de investigación, que dinamicen el del conocimiento en diferentes áreas de interés para la región y el país. Es decir, la primera responsabilidad es ser auténticas facultades de Ingeniería que avancen en competencias, que cumplan con calidad y excelencia con lo que les corresponde respecto a las funciones misionales de la Universidad de investigación, docencia y extensión.

Respecto a la formación en Ingeniería es necesario estructurar programas de estudio pertinentes y con los recursos suficientes para formar ingenieros que, en su ejercicio profesional, contribuyan efectivamente con el desarrollo del país, en esta sociedad global denominada “del conocimiento”.

Todas las capacidades de las facultades de Ingeniería deben estar al servicio de la sociedad, para ello deben facilitarse los procesos que relacionan Universidad- Empresa-Estado-Sociedad, para que ese conocimiento superior que se alcance en los grupos de investigación de las facultades de Ingeniería sea transferido y utilizado de manera adecuada y mejore las condiciones de vida del entorno social y del país.

El gran reto de las facultades de Ingeniería del país es asumir con responsabilidad el conocimiento en sus disciplinas para el servicio de la sociedad. Por ello es preciso formar ingenieros con conocimientos actualizados y pertinentes, con una postura ética y de respeto con el medio ambiente. De otro lado, las facultades de Ingeniería deben mantener una estrecha relación con los múltiples actores sociales que requieran de su conocimiento.

* Vicerrector de Investigaciones, Innovación y Extensión. Universidad Tecnológica de Pereira - Colombia. Correo electrónico: viceiie@utp.edu.co

ARQUITECTURA PARA LA EXTRACCIÓN DE INFORMACIÓN DE ESPACIOS URBANOS

ARCHITECTURE FOR THE EXTRACTION OF INFORMATION OF URBAN SPACES

Mg. Alejandro Forero Guzmán
Pontificia Universidad Javeriana, Colombia
alejandro.forero@javeriana.edu.co

Ph.D. Carlos Parra Rodríguez
Pontificia Universidad Javeriana, Colombia
carlos.parra@javeriana.edu.co

(Artículo de **Investigación científica y tecnológica**. Recibido el 13/10/2011. Aprobado el 19/12/2011)

Resumen. El presente artículo presenta una arquitectura para la extracción de información de espacios urbanos, la extracción de información de espacios urbanos, la cual debe permitir la integración entre los sistemas de visión por computador que procesan imágenes o video con estrategias de bajo nivel, con la interpretación semántica que los humanos hacen de los espacios. En este documento también se presenta la estructura de esta arquitectura y las limitaciones que introducen los sistemas de adquisición con cámaras.

Palabras clave: Adquisición; representación de conocimiento; visión por computador.

Abstract. This paper presents an architecture for information extraction of urban spaces, which should provide integration between computer vision systems that process video images or low-level strategies, with the semantic interpretation that humans make of spaces urban. This document also presents the structure of this architecture and the constraints introduced by the camera acquisition systems.

Keywords: Knowledge acquisition/ representation, computer vision.

1. INTRODUCCIÓN

La cantidad de actividades que el ser humano realiza en las ciudades se incrementa y diversifica a medida que éstas crecen y se convierten en espacios más variados y, por ende, más complejos para describir y analizar. La movilidad es un ejemplo de acciones urbanas que se ha hecho más confusa y engorrosa con el crecimiento de las ciudades; el tráfico, los accidentes y la contaminación, son aspectos negativos que se acentúan con la masificación de la necesidad de desplazarse. La ciudad de Bogotá no ha sido ajena a esta situación, de hecho, el número de vehículos particulares registrados en la ciudad aumentó más del 60% entre 2008 y 2011 [1]; esta es una tendencia que se observa en todas las ciudades del mundo.

Afortunadamente, al tiempo que las necesidades surgen, también se generan alternativas tecnológicas que buscan mitigar dichas necesidades. Por ejemplo, los semáforos se implementaron para minimizar los problemas ocasionados por el tráfico. La incorporación de la tecnología pretende mantener bajo control las necesidades que crecen en las ciudades. Sin embargo, estas urgencias están interrelacionadas entre ellas: los problemas relacionados con la movilidad afectan la concepción misma de la ciudad, la forma en la que se planifica, se construye y se establecen regulaciones, no solo del tráfico sino de todos los aspectos de la vida urbana. Con esta perspectiva, las necesidades que se crean requieren más información. Y la cantidad de información debe crecer al menos al mismo ritmo con el que se efectúa cada actividad en la ciudad. Tener información suficiente permitirá planear y aplicar las medidas adecuadas para afrontar los requerimientos de la ciudad, entre ellos la movilidad o la seguridad.

Los requerimientos que plantea la movilidad, dio origen a la ubicación de unos sensores que tenían por objeto registrar la cantidad de vehículos que circulan por un lugar en un día. Los primeros sensores se desarrollaron con el principio de inducción electromagnética porque permitían registrar el paso de los vehículos; sin embargo, desde las últimas décadas, se ha privilegiado como sistema de adquisición de información las cámaras y procesamiento de video, en particular por la posibilidad de ampliar las variables para observar [2]. En la actualidad, tan importante como la información de los vehículos que circulan, es la información de los peatones y demás usuarios de la vía. Esto es especialmente notorio en espacios distintos a las autopistas, en las que la diversidad de

usos y usuarios de las vías hace que los sistemas basados en visión sean ideales para extraer la información de lo que sucede en estos espacios urbanos [3]. En la Fig. 1 se ilustra un ejemplo de la versatilidad de los sistemas sustentados en visión, (imagen tomada de [4]), donde simultáneamente se extrae información de los vehículos que circulan, se identifican, se siguen y se estima la velocidad; al mismo tiempo, se detectan y siguen los peatones y grupos de peatones, para luego medir distancias entre peatones y vehículos.

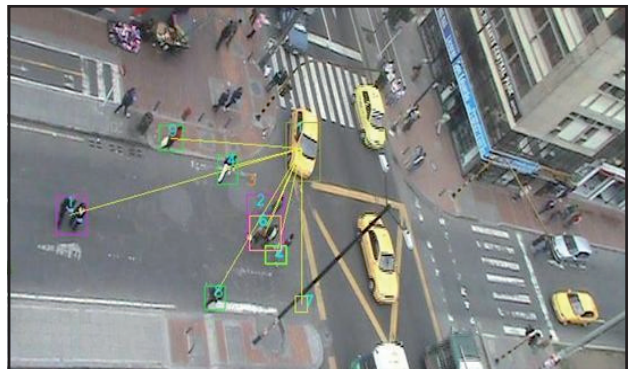


Fig. 1. Detección y seguimiento de múltiples variables por un sistema de Visión por computador

Hoy, los sistemas de cámaras y procesamiento de video se han convertido en sistemas de Visión por computador, encargados de extraer información del entorno, en este caso de los espacios urbanos [5]. La Visión por computador es el sensor privilegiado en la observación de las condiciones en la ciudad, el medio de adquisición de información para sistemas que tienen una complejidad mayor y que pueden incluir los sistemas de control de tráfico, de monitoreo urbano o de información de las condiciones existentes.

El avance de los sistemas de Visión por cámara, como sistema de adquisición de información de las condiciones observadas, se ha favorecido por los desarrollos en las capacidades de procesamiento en los sistemas de cómputo y por el creciente interés en los sistemas de visión en tiempo real. Sin embargo, no todos los avances realizados en estos campos tienen presente la naturalidad de la información que espera un usuario humano: “...there is still a big gap in analytical skills between a typical security guard and state-of-the-art image-processing algorithms” [6].

Esta tendencia de interpretar la información desde el procesamiento de imágenes sin integrar la interpretación de la información del lenguaje humano, produ-

ce dos problemas. Primero, aumenta la complejidad algorítmica para interpretar la imagen por no usar la información implícita que se tiene de la escena analizada. Por ejemplo, en la interpretación de una vía, varios de los sistemas de Visión por cámara no incorporan la percepción del conductor o del peatón de esa vía: como podrían ser la presencia de usuarios u obstáculos habituales.

Segundo, crea una ruptura entre los resultados generados por la interpretación de los sistemas de Visión por cámara, que se enfocan en el procesamiento de la imagen, y los usos y usuarios humanos, quienes precisan de esa información; esto es lo que algunos autores llaman “*semantic gap*” [7]. Visto de otra forma, estos despliegues no integran un aprendizaje de máquina que permita caracterizar los elementos del ambiente humano [8].

Dado que desde múltiples disciplinas se estudia, se planea o se ejerce algún control del espacio urbano y la forma de entender este espacio es en sí muy diversa, este trabajo presenta una arquitectura para adquirir información del espacio urbano a partir de la información entregada automáticamente por los sistemas de Visión por cámara. Esta estructura permitirá describir los fragmentos del espacio urbano a través de la interpretación de bajo nivel que hacen los sistemas de Visión por cámara, analizar píxeles o conjuntos de éstos para construir una descripción de la escena. Adicionalmente, la estructura incluye, en la descripción del espacio urbano, la información semántica que describe el espacio y el uso que se le da. Trabajos similares proponen una estructura topográfica y otra topológica, como [9] [10]. Esta idea se incorpora en la arquitectura que se propone.

Mediante la estructura con la que se describe el espacio urbano desde el sistema de Visión por cámara, se sugiere una arquitectura para almacenar la información de una escena, que se asocia con el espacio observado por una cámara de video. Además, en este trabajo, se definen las reglas encaminadas a conectar fragmentos del espacio urbano observado y cada fragmento sintetizado con la arquitectura propuesta. A largo plazo, su construcción debería permitir llegar a un modelo de un espacio urbano de tamaño arbitrario. Para tal propósito se parte de la descripción de múltiples escenas y de la síntesis de la información en la arquitectura que se propone.

Adicionalmente, si se enriquece la captura de información con las actividades en los espacios urbanos,

se plantea realimentar la estructura de datos propuesta con la descripción semántica de actividades del espacio observado. Así se incorporará a la arquitectura la información semántica que caracteriza el uso de un espacio urbano; un ejemplo de esta información es la descripción que se hace de una vía vehicular: el número de carriles que la componen, el tipo de tráfico que la recorre, el sentido del tráfico (p.ej. sur-norte), si la cruzan otras vías, o la velocidad máxima permitida, entre otros datos que se pueden asociar a la descripción de este espacio.

En la siguiente sección se formaliza el modelo con el que se describen las escenas. Después se detalla la arquitectura propuesta para almacenar la información de la escena desde la información de bajo nivel, luego, se describe la información semántica y, por último, se hace la integración entre las dos. En la cuarta sección se alude a los alcances y limitaciones de la arquitectura. Finalmente, se presentan las conclusiones de este trabajo.

2. FORMALIZACIÓN DEL MODELO DE LA ESCENA

El espacio urbano se considera como una unión de infinitud de espacios. De igual forma, todo espacio urbano está conformado por la unión de distintos elementos básicos. La descripción de un espacio urbano se construye desde la definición de los elementos que lo componen.

A partir de la definición de los elementos básicos que componen el espacio urbano se construye una representación de cualquier espacio que es observado en una ciudad. Esta idea es particularmente útil en la generación de un modelo que permita describir la ciudad a través de un sistema de Visión por computador (VC).

Al considerar el espacio urbano capturado en el plano imagen por una cámara de vista superior, se busca el conjunto de elementos básicos que se encontrará en el plano imagen, independiente del espacio urbano que se observa.

Dado que se intenta describir la dinámica que se observa en los espacios urbanos, se busca que los elementos básicos usados también sean relevantes para detallar las acciones que suceden en dicho espacio.

Teniendo presentes los criterios mencionados se tiene:

Sea E el conjunto de las distintas escenas que se capturan con el uso de una cámara de vista superior fija en el espacio, de forma que E_n , es una escena capturada y es un elemento de E .

El plano imagen que representa esta escena E_n , está conformado por m segmentos.

El conjunto de segmentos S , está compuesto por instancias de los elementos: vía (v), intersección (i), plaza (p) u obstáculo (o).

$$E_n \in \mathbf{E}$$

$$E_n = \bigcup_{j=1}^m (S_j)$$

$$S_j \in \mathbf{S}$$

$$\mathbf{S} = \{v, i, p, o\}$$

Los segmentos vía (v), se conforman como espacios de tránsito y se clasifican de acuerdo con los móviles que los recorren. Adicionalmente, se caracteriza por su estructura geométrica y las del tráfico que circula por el segmento, es decir, la velocidad y la densidad de los móviles.

Los segmentos intersección (i), son espacios en los que se encuentran dos o más vías y pueden clasificarse de acuerdo con los móviles que se encuentran en la intersección o con la estructura geométrica de la misma intersección.

Los segmentos plaza (p), se conforman como espacios de tránsito y/o estadía, primordialmente de peatones, que no se encuentran atravesados por una vía.

Los segmentos obstáculo (o), incluye cualquier segmento que no se clasifique dentro de uno de los tres segmentos anteriores. Estos segmentos son indispensables para agrupar a todos los segmentos sobre los que no es válido ningún desplazamiento de los móviles en la escena.

3. ARQUITECTURA PROPUESTA

Esta arquitectura sintetiza la interpretación del espacio urbano con el modelo que se plantea e introduce otros conceptos a partir de la interpretación de bajo nivel realizada por un sistema de Visión de cámara. Adicionalmente, intenta una estructura para enriquecer la descripción con la interpretación semántica del

espacio urbano observado basado en el uso que se hace de dicho espacio.

En general, cualquier sistema de Visión de cámara suele concentrarse en alguna característica semántica de una escena y trata de optimizar la identificación o extracción de la misma. La arquitectura que se propone no difiere de este principio básico, pero busca generalizarlo e incluir las actividades que se desarrollan en el espacio urbano o, al menos, producir una estructura capaz de incluir las actividades que se observan: datos básicos como el conteo de móviles que se desplazan (p.ej. vehículos o peatones) más elaborados como el registro de infracciones en un cruce semaforizado.

Con esta arquitectura se describe cómo está construido un espacio urbano, pero, además, se incluye la información de las acciones que se dan en ese espacio como la circulación de vehículos y los aspectos que caracterizan esa circulación, como las velocidades permitidas, el sentido de la circulación y demás normas que regulen esa actividad. A continuación se presenta cómo, desde la interpretación del plano imagen por un sistema Visión de cámara y la interpretación semántica del espacio se construye la arquitectura para extraer y almacenar información del espacio observado.

3.1 Interpretación de bajo nivel

La interpretación de bajo nivel está inspirada en los trabajos previos realizados en el grupo de investigación SIRP (Sistemas Inteligentes Percepción y Robótica) de la Pontificia Universidad Javeriana - Bogotá, en los que se han desarrollado técnicas de Visión de cámara relacionadas con la movilidad. Estos trabajos abarcan desde la detección del camino para una plataforma móvil [11], pasando por la adquisición de variables de tráfico en una escena particular [12], [13], [14], [15], [16], hasta la interacción de variables en la escena [16], [17].

En la interpretación del Plano imagen se genera un modelo del fondo de la escena observada [12], [18], [19]. Una vez estimado el modelo del fondo, se identifican los objetos que lo componen y se usa la información que suministra el usuario humano. Esta información corresponde a los objetos que físicamente conforman el espacio urbano observado. El modelo del fondo también se emplea para extraer el primer plano, el cual se usa para identificar los móviles que

recorren la escena. Para caracterizar los conjuntos que conforman la escena y los móviles se utilizan características basadas en HOG [20] y matrices de covarianza [21] que, aunque probados para la detección de peatones, por sus características, permiten identificar otros tipos de estructura [22], en especial cuando la información de los bordes es relevante.

Con el modelo de fondo se construye un nuevo Plano imagen, que se segmenta para distinguir los objetos que hacen parte del fondo. La técnica seleccionada para la segmentación debe permitir separar las vías, los edificios, la vegetación y una cuarta clase que agrupe a los demás objetos que se presenten en la escena, como el equipamiento urbano, entre otros.

El primer plano se segmenta para individualizar los móviles que transitan la escena. Posteriormente, los móviles deben ser clasificados, para la que se inicia con una tipificación dada por el tamaño de los móviles [2], [15], que se depura a medida que las capacidades de los algoritmos implementados y la calidad de los datos se vean mejorados. En el caso de peatones que caminan juntos, se clasifican como grupo de peatones [4].

En la Fig. 2 se presenta la estructura que conforma el núcleo de la arquitectura. Con el Plano imagen se construyen dos representaciones, el Fondo y el Primer plano. A su vez, cada una de estas representaciones se desglosa para agrupar los componentes que se presentan en la escena capturada en el Plano imagen.

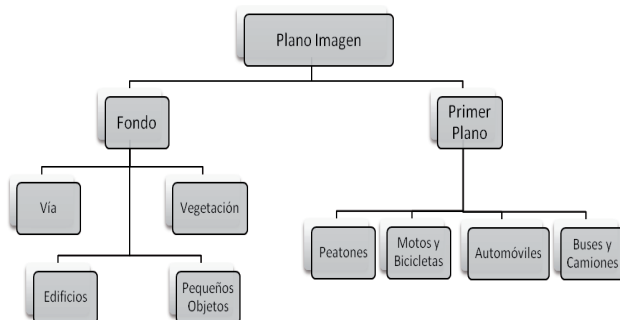


Fig. 2. Representaciones a partir del plano imagen

La representación del Fondo, se descompone en las clases que suelen componer la escena urbana, y de esta forma se construye una Clase vía y una Clase edificios (primera bifurcación en la Figura 2), que se encuentran en la mayoría de las escenas urbanas

que se observan desde la vista superior en ciudades como Bogotá. También, se adiciona una representación de la vegetación, como es observada por la cámara y extraída por el sistema de Visión de cámara, con lo que se construye la Clase vegetación. Se añade una representación que agrupa los objetos que completan la escena, como el equipamiento urbano. Esta representación puede incluir también pequeños objetos no previstos inicialmente y que hacen parte del fondo de la escena; con todo esto se construye la cuarta clase debajo de la Clase fondo.

De igual forma, se procede con la Clase Primer plano para descomponerla en cuatro clases, con el uso del tamaño de los objetos segmentados como primer criterio para agrupar los móviles detectados, tal como se muestra en la rama derecha de la Fig. 2. Otros criterios se pueden construir con la observación en el tiempo de los móviles.

3.2 Interpretación semántica

La interpretación semántica que el usuario humano construye del espacio urbano cuando se crean modelos cognitivos de apropiación de la información [23], se utilizará para enriquecer la descripción de la escena observada por la cámara [8], [24]. La arquitectura propone representar en forma de árbol la información semántica para facilitar el mapeo con la información extraída por el sistema de Visión de cámara. El árbol como estructura también permite incorporar la clasificación en conjuntos disyuntos que se hace tanto por la interpretación semántica como por el sistema de Visión de cámara.

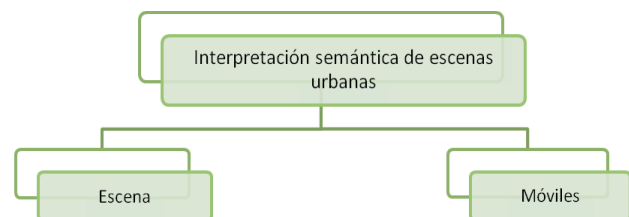


Fig. 3. Bifurcación de la interpretación semántica

Se generan dos clases: la Clase escena, correspondiente al espacio urbano observado, y la Clase móviles, que corresponde a los móviles que recorren dicho espacio. Así, la Clase escena se corresponde con el Fondo extraído por el sistema de Visión de cámara; al igual que la Clase móviles que se corresponde con el Primer plano. Esta primera bifurcación se ilustra en la

Fig. 3, desde la que se intuye la similitud con la interpretación de bajo nivel propuesta en la sección anterior.

Para aprovechar las similitudes entre la interpretación semántica y los resultados del sistema de Visión de cámara, tanto la Clase escena como la Clase móviles, se subdividen en cuatro clases. Con esto se busca favorecer la correspondencia entre la interpretación del Plano imagen representado por el Fondo y el Primer plano, con la descripción semántica que se hace del espacio y los móviles que la recorren.

Por ejemplo, la Clase vía especifica un subconjunto de los objetos que conforman el fondo de la escena en la interpretación de bajo nivel. La información semántica que enriquece la descripción de esta clase incluye: el número de carriles que tiene la vía, el tipo de tráfico que la recorre, el sentido del tráfico (p.ej. sur-norte), si la cruzan otras vías, la velocidad máxima permitida, entre otros datos que definen el espacio o los usos que se le dan. La Fig. 4 ilustra este ejemplo.

Adicionalmente, a los detalles particulares de la clase que se construye, las características generales de la escena y la ubicación de la cámara complementan la información que se busca agrupar para completar la descripción de la escena. Esta descripción se puede continuar en más subniveles, siempre que el sistema de Visión de cámara sea capaz de diferenciar las subclases que se incluyan.

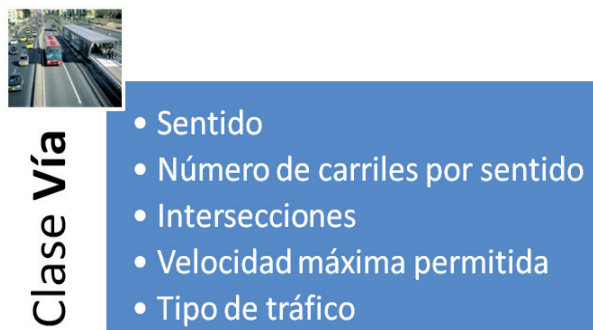


Fig. 4. Información semántica que enriquece la Clase vía

3.3 Integración de bajo nivel con la información semántica

La arquitectura que se propone busca enriquecer los datos que se extraen del plano imagen mediante el sistema de Visión de cámara, con la informa-

ción semántica que describe la escena. El objetivo es que cada una de las clases que conforman la interpretación de bajo nivel se enriquezca con la información semántica que se asocia con cada clase. Adicionalmente, la integración de estos datos se sintetiza en un repositorio que almacena la información que se extrae a medida que el sistema de adquisición de información funciona. De esta forma, se construye el conocimiento explícito que describe la escena, desde el nivel semántico hasta el nivel de procesamiento en el Plano imagen. En la Fig. 5 se observa un ejemplo para la Clase vía de cómo se integran los subsistemas y se almacena la información: en el eje vertical se depura un proceso de bajo nivel, mientras el eje horizontal adiciona la información semántica e integra el Repositorio de datos, donde se almacenan todos los descriptores de la clase.

El proceso descrito con la Clase vía se repite para cada una de las clases que componen la arquitectura, hasta alcanzar el nivel de detalle buscado en la descripción de la escena; el mismo procedimiento se sigue para completar la descripción de los móviles. De esta forma, se integra un esquema de aprendizaje que permite aprender los significados implícitos en el espacio para robustecer el sistema, al tiempo que se construye un lenguaje común con el usuario humano.

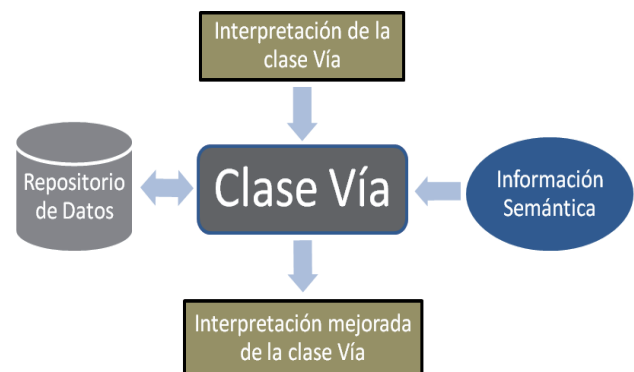


Fig. 5. Integración del procesamiento de bajo nivel con la información semántica y el Repositorio de datos.

Después de completar una estructura capaz de almacenar la información que describe la escena, se integra la información temporal que se produce con el desplazamiento de los móviles. La información temporal surge del procesamiento de los *frames* que conforman el video de observación. Así, los datos provenientes del sistema de Visión de cámara son producto de las posiciones de los móviles en cada uno de los planos imagen procesados (uno por cada

frame) y una etapa de seguimiento que reconstruye la trayectoria de los móviles.

La información temporal también se agrupa en niveles de abstracción que permite integrar la información semántica a los datos que se generan en la etapa de seguimiento del sistema de Visión de cámara. Así, se integra la información semántica que hace referencia a las posibles interacciones entre los móviles y la escena [24]. En el Plano imagen esta información ayuda a especificar las oclusiones que se pueden dar por la posición de la cámara, la estructura de la escena y la ruta de desplazamiento de los móviles.

Esta información temporal se agrupa en la Clase interacciones que conecta la información condensada en la Clase escena y en la Clase móviles, y permite un nivel de abstracción de la información extraída más allá de las posibilidades del sistema de Visión de cámara, puesto que da elementos para interpretar los datos capturados, aun en la eventualidad de oclusiones o ante posibles errores del sistema.

Una vez definida la escena, registrados los móviles y adicionada la información de las interacciones posibles entre móviles y móviles, y entre móviles y escena [2], se construye un registro del desplazamiento de cada móvil en la escena. Esta información adicional se condensa en una definición de eventos [2]. La Clase eventos es el último nivel de información que se almacena en la arquitectura que extrae la información de la escena. Esta clase condensa la información espacial y temporal de bajo nivel y la transforma en argumentos semánticos para ser interpretados por un usuario humano, tales como contravenciones de tránsito [25], accidentes, situaciones de riesgo, problemas de seguridad y, en general, descripciones del comportamiento de las personas que habitan los espacios urbanos y sus objetos [2], [6]. En la Fig. 6 se presenta una gráfica que ilustra cómo interrelacionan las Clases Escena, Móviles e Interacciones.

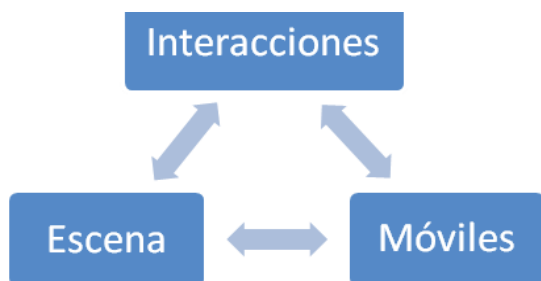


Fig. 6. Relación entre las clases Escena, Móviles e Interacciones

Las interacciones se constituyen en un tercer eje que agrupa la información en distintos niveles de abstracción hasta definir la Clase eventos, tal como se observa en la Fig. 7.

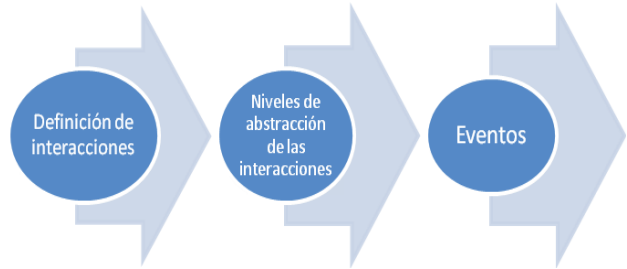


Fig. 7. De la definición de interacciones a la construcción de eventos

4. LIMITACIONES Y ALCANCES DE LA ARQUITECTURA

Las capacidades del sistema de adquisición de información de escenas urbanas, basado en la arquitectura propuesta, están limitadas por la observabilidad del espacio y la observabilidad de los eventos que se registren sobre cada espacio.

En cuanto a la observabilidad del espacio, la escena que captura el sistema de Visión de cámara y en particular el Plano imagen que se genera, está limitado por el tamaño mismo de la escena o por su región visible; por las características del sensor y de la óptica de la cámara; y por la ubicación misma de la cámara, en relación con el espacio que se observa. Dada la infinidad de combinaciones que se producen entre estos tres elementos, (escena, características de la cámara y ubicación relativa entre la cámara y la escena), las limitaciones del sistema se describirán sobre las características del Plano imagen, que es el resultado final de toda la etapa de captura.

Para el correcto funcionamiento del sistema de Visión de cámara se debe trabajar con un mínimo de información cuantificada por la cantidad de píxeles por objeto observado. También se deben tener en cuenta las oclusiones que se dan entre los objetos observados, pues éstas también limitan la cantidad de píxeles asociados con cada objeto. En últimas, se pretende definir un mínimo para que el sistema de Visión de cámara distinga los objetos en el Plano imagen y posteriormente sea capaz de seguir dichos objetos en el tiempo mientras recorren la escena.

Siguiendo esta idea, se propone trabajar con imágenes de 640 por 480 píxeles y limitar el espacio observado a 10m x 7,5m. Con esto se tendría un mínimo de 64 píxeles por metro lineal. (64 píxeles x 64 píxeles por m²). Con esta limitación del espacio, se podrían tener espacios más pequeños porque se aumenta la resolución. Esta limitación está dada para observar peatones, puesto que, en el caso de que el espacio sea únicamente vehicular, la resolución en píxeles podría bajar hasta 4 veces, es decir, observar espacios de hasta 40m x 30m y mantener la misma resolución en píxeles.

Si este espacio se toma sobre una vía vehicular, la cantidad de carros se limitaría a los que se observan, de acuerdo con el número de carriles de vía. También se propone mantener la estructura 4 a 3 del encuadre del plano observado. En cuanto a la densidad vehicular en la vía, la cantidad de vehículos presentes estará limitada por el tamaño de los mismos, teniendo presente que el tamaño promedio de un automóvil es 1,8m x 4,5m.

Si se observa un espacio peatonal, la evaluación del modelo semántico se limitará a espacios no atiborrados de transeúntes. En términos cuantitativos, sobre el espacio de 70 m² se espera que no haya más de 35 transeúntes simultáneamente, ($A [m^2]/2$) y que en ninguna parte se encuentren más de 3 personas por metro cuadrado. Estas limitaciones se plantean para concentrarse en la descripción semántica y no generar una dependencia del desempeño de los algoritmos de Visión de cámara de bajo nivel.

5. CONCLUSIÓN

Con este trabajo se busca avanzar en el despliegue de sistemas de adquisición de información en forma automática, con los que se enfrentan dos de los problemas que más preocupan a la sociedad en las grandes ciudades, como son: la seguridad y la movilidad. Sin embargo, la arquitectura propuesta plantea un sistema de adquisición de información que incluye las ventajas de los sistemas automáticos pero sin dejar de lado la interpretación humana de esa información. Desde esta perspectiva, la arquitectura debería servir para tomar decisiones sobre el uso de los espacios urbanos.

REFERENCIAS

- [1] Observatorio Ambiental de Bogotá. Secretaría Distrital de Ambiente. "Número de vehículos particulares-NVP" [Online]. Available: <http://oab.ambientebogota.gov.co/index.shtml?s=l&id=272&v=l>. Nov. 10, 2011.
- [2] P. Kumar, S. Ranganath, H. Weimin, and K. Sengupta, "Framework for Real-Time Behavior Interpretation From Traffic Video," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 43-53, Mar. 2005.
- [3] C. Machy, C. Carincotte, and X. Desurmont, "On the use of Video Content Analysis in ITS : A review from academic to commercial applications," in *2009 9th International Conference on Intelligent Transport Systems Telecommunications, (ITST)*, 2009, no. 23, pp. 574-579.
- [4] A. Forero, C. Diaz, A. Sanchez, L. Quintana, F. Mariño, and L. Barrero, "Implementación de una Herramienta para la Adquisición de Variables de Tráfico: Peatones y Vehículos, Utilizada en la Evaluación del Riesgo de Peatones que Cruzan la Vía," in *XIII STSIVA*, 2008, pp. 95-98.
- [5] C. Regazzoni, A. Cavallaro, Y. Wu, J. Konrad, and A. Hampapur, "Video Analytics for Surveillance: Theory and Practice [From the Guest Editors]," *IEEE Signal Processing Magazine*, vol. 27, no. 5, pp. 16-17, Sep-2010.
- [6] J. Candamo, M. Shreve, D. B. Goldgof, D. B. Sapper, and R. Kasturi, "Understanding Transit Scenes: A Survey on Human Behavior-Recognition Algorithms," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 1, pp. 206-224, Mar. 2010.
- [7] R. Porter, A. Fraser, and D. Hush, "Wide-Area Motion Imagery," *IEEE Signal Processing Magazine*, vol. 27, no. 5, pp. 56-65, Sep-2010.
- [8] D. Crever and R. Lepage, "Knowledge-Based Image Understanding Systems : A Survey," *Computer Vision and Image Understanding*, vol. 67, no. 2, pp. 161-185, 1997.
- [9] B. T. Morris and M. M. Trivedi, "Trajectory learning for activity understanding: unsupervised, multilevel, and long-term adaptive approach.," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 11, pp. 2287-301, Nov. 2011.
- [10] D. Makris and T. Ellis, "Learning semantic scene models from observing activity in visual surveillance.," *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and*

- Cybernetics Society, vol. 35, no. 3. pp. 397–408, Jun-2005.
- [11] A. Forero and C. Parra, "Extraction of Roads From Out Door Images," in *Vision Systems: Applications*, no. June 2007, G. Obinata and A. Dutta, Eds. InTech, 2007, pp. 101-112.
- [12] G. E. Urrego, F. C. Calderón, A. Forero, and J. A. Quiroga, "Adquisición de variables de Tráfico vehicular usando visión por computador," *Revista de ingeniería Universidad de los Andes*, no. 30, pp. 7-15, 2009.
- [13] J. Quiroga, N. Romero, C. García, and C. Parra, "Adquisición de variables de tráfico peatonal utilizando visión por computador Pedestrian traffic variables acquisition using computer vision," *Revista Facultad de Ingeniería Universidad de Antioquia*, no. 60, pp. 51-61, 2011.
- [14] D. Insuasti, J. Quiroga, and A. Forero, "Detección y Seguimiento de Vehículos Automotores en Video," in *XIII STSIVA*, 2008.
- [15] F. C. Calderón and A. Forero, "Low Complexity Algorithm For The Extraction Of Vehicular Traffic Variables," in *IEEE ITSC 2011*, 2011.
- [16] E. A. Romero, J. Quiroga, and A. Forero, "Seguimiento de Personas Basado en los Descriptores HOG," in *XVI STSIVA*, 2011, no. 2.
- [17] A. Forero, J. Quiroga, C. Parra, and C. L. Niño, "Información y caracterización del tráfico vehicular usando visión por computador," in *Reunión nacional ACOFI 2011*, 2011.
- [18] M. David and J. Quiroga, "Estudio y Análisis del Algoritmo de Sustracción de Fondo Codebook," in *XV STSIVA*, 2011.
- [19] K. Kim, T. Chalidabhongse, D. Harwood, and L. Davis, "Real-time foreground-background segmentation using codebook model," *Real-Time Imaging*, vol. 11, no. 3, pp. 172-185, Jun. 2005.
- [20] N. Dalal and B. Triggs, *Histograms of Oriented Gradients for Human Detection*. IEEE, 2005, pp. 886-893.
- [21] F. Porikli, P. Meer, and O. Tuzel, "Human detection via classification on riemannian manifolds," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2007, pp. 1-8.
- [22] F. M. Porikli, "Classification on Riemannian Manifolds," *videlectures.net*, 2010.
- [23] J. M. Zacks and B. Tversky, "Event structure in perception and conception.," *Psychological bulletin*, vol. 127, no. 1, pp. 3-21, Jan. 2001.
- [24] C. Hudelot, "Towards a Cognitive Vision Platform for Semantic Image Interpretation," *UNIVERSITE DE NICE - SOPHIA ANTIPOLIS*, 2005.
- [25] J. A. Vijverberg, N. A. H. de Koning, J. Han, P. H. N. de With, and D. Cornelissen, "High-Level Traffic-Violation Detection for Embedded Traffic Analysis," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, 2007*, Vol. 22, No. 12, p. 11-793-11-796.

MODELADO CONCEPTUAL DE LA SUPERVISIÓN DE UN SISTEMA HOLÓNICO DE PRODUCCIÓN CONTINUA

CONCEPTUAL MODELING OF THE SUPERVISION OF CONTINUOUS PRODUCTION HOLONIC SYSTEMS

Ph.D. Carlos Parra Ortega

Universidad de Pamplona, Colombia
carapa@unipamplona.edu.co

(Artículo de **Reflexión**. Recibido el 29/10/2011. Aprobado el 19/12/2011)

Resumen. La arquitectura de automatización PROSA es una descripción orientada hacia objetos que implementan sistemas de producción. Un enfoque basado en sistemas para eventos discretos, detección de eventos y la teoría de control supervisorio, representa el comportamiento de una unidad de producción controlada. Para obtener la descripción orientada a objetos de esta unidad, se aplicó modelado conceptual y se utilizó el Lenguaje unificado de modelado.

Palabras clave: Arquitectura de automatización PROSA; Control supervisorio; Detección de eventos; Modelado conceptual; Sistemas de eventos discretos; Lenguaje unificado de modelado.

Abstract. The PROSA Automation Architecture is an object-oriented template that implements production units. An approach based on Discrete-Event Systems, Event-Detection and theory of Supervisory Control represents the dynamical behavior of a controlled production unit described under the PROSA architecture. In order to obtain the description of this unit, conceptual modeling using Unified Modeling Language is applied.

Keywords: PROSA Automation Architecture, Supervisory Control, Event Detection, Conceptual Modeling, Discrete-Event Systems, Unified Modeling Language.

1. INTRODUCCIÓN

El desarrollo de sistemas con capacidad de supervisar el comportamiento de unidades de producción sobre la base tecnológica disponible es un problema de suma actualidad. Diversos enfoques se han propuesto para aplicar los mecanismos de control, supervisión y gestión para un proceso productivo en el que estos mecanismos proyectan su funcionamiento hacia la arquitectura computacional y de comunicaciones de la empresa, cuya condición ideal de operación para estas organizaciones ocurre cuando existe la integración de estos elementos y, en su implementación, se usan las tecnologías de informática y comunicaciones de avanzada. Uno de los beneficios que resultan de integrar control e información es obtener una unidad de producción autoajutable. Una condición para ajustar un sistema de producción es la recolección oportuna de datos del piso de planta, de manera que permita determinar su estado y las decisiones para controlarlo. Además de que lo pueda comunicar a otras unidades de producción con el fin de coordinar sus actividades. En la Fig. 1 se observa un esquema general de una unidad de producción cuya función es controlar un proceso por etapas y coordina las actividades de otros componentes que, a su vez, son unidades de producción que controlan un proceso continuo atómico. Cada una de estas unidades detecta eventos cuya ocurrencia es difícil de predecir con exactitud porque implica analizar datos históricos del piso de planta. Esta detección de eventos es condición necesaria para derivar acciones que permitan mantener una trayectoria deseada en las variables continuas que se están controlando y comunicar su estado actual a la unidad de producción que la contiene. Esta unidad observa los eventos y actualiza la imagen del global y, eventualmente, decide la ejecución de algunas acciones que comunicará a sus unidades. Una unidad de producción compleja puede conformarse por medio de la composición de unidades de producción básicas y su evolución dinámica en el tiempo resulta del acoplamiento de las dinámicas de las unidades componentes.

Otro esquema alternativo obvia la coordinación por parte de holones agregados, y en su lugar establece una relación de colaboración entre las unidades de producción que corresponden a etapas diferentes de producción. No existe una coordinación explícita sino un conjunto de diálogos entre holones básicos de producción de forma que cada uno solicita y ofrece recursos y productos terminados. Es necesario aclarar que un producto final de una unidad de produc-

ción se concibe como un recurso para una unidad de producción diferente, y así sucesivamente.

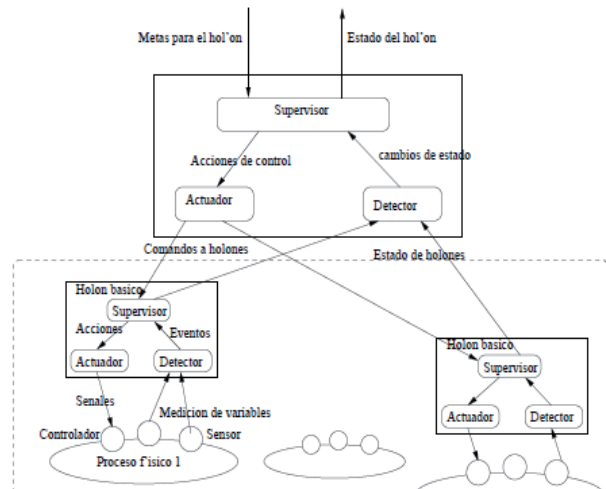


Fig. 1. Esquema general de unidad de producción supervisada

Es conveniente definir los conceptos de supervisión y de planta industrial. Por esta última se entiende que es el sistema que se ha de controlar, incluye la producción física, los insumos, el método de producción y los controles de equipamiento. El supervisor se define como el controlador de la planta a alto nivel, quien previene que la misma llegue a unas condiciones de operación no deseada. El proceso de la supervisión está asociado con la detección de eventos porque es menester analizar si ocurrió un evento que implique cambios en el estado en el que se encuentra la planta, las mediciones de sus variables o su estado actual. La dinámica del supervisor y la de la planta se describen por medio de un Sistema de Eventos Discretos (SED, de aquí en adelante) y las secuencias de eventos asociadas con estos SED conforman el lenguaje de la planta y del control supervisorio. En la Fig. 2 se aprecia la dinámica de un proceso supervisado.

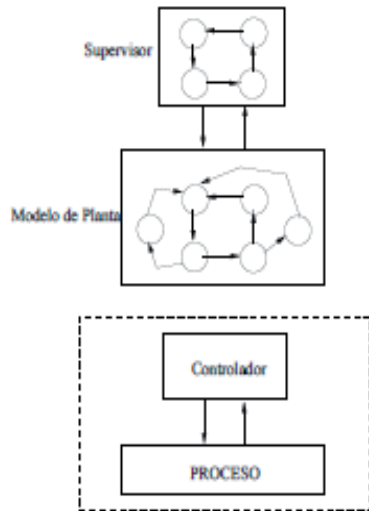


Fig. 2. Dinámica de un supervisor

El presente artículo es una propuesta para obtener el Modelo de estado de un sistema de manufactura holónico, con el fin de facilitar la implementación de supervisores para las unidades de producción en su forma más cercana al piso de planta (la más básica).

Antes de continuar, es necesario definir el concepto de Estado del proceso, que se asume como un valor de ciertas características en un tiempo determinado. Como varios actores observan y requieren información diferente, son necesarias varias perspectivas para modelar conceptualmente este estado. Las siguientes preguntas surgen como justificación para investigar sobre este tópico de modelado conceptual: ¿qué características deben asignarse a la descripción del estado del proceso? ¿Cómo debe ser el repositorio de datos? ¿Es factible una descripción general de este repositorio? ¿Cómo debe aplicarse un modelo general del repositorio en un caso práctico? Para responderlas, primero se modela el comportamiento del sistema supervisado, y luego se determina la estructura computacional estática de la unidad de producción que contiene al estado del proceso. Posteriormente, se detalla el metamodelo del producción.

2. ESTADO DEL ARTE

Dentro de los sistemas holónicos se han llevado a cabo proyectos de modelado conceptual. Algunos de ellos se han enfocado principalmente hacia la des-

cripción del estado del producto como en el proyecto HoMUCs [5], otros hacia la autosimilaridad de tres componentes fundamentales: Recurso, Producto y Orden como aparece planteado en PROSA, propuesto por Valckenaers, Van Brussel y otros investigadores [12], [10]. Chacón, Besembel y Hennem proponen la supervisión de unidades de producción con holones definidos con la arquitectura PROSA para gestionar un proceso industrial continuo [2]. El comportamiento dinámico de esta unidad de producción puede describirse por medio de un Sistema de Eventos Discretos (DES). La composición de varias unidades de producción implica un acoplamiento de DES, a los que se les sintetiza un supervisor que mantenga el comportamiento global dentro de unas metas deseadas. Blanc [1] aplica el paradigma PROSA en el monitoreo y planificación de un sistema de manufactura discreto, en el que el proceso productivo gira alrededor de la dinámica de las órdenes de producción. Aunque estos trabajos describen la supervisión, el control y la dinámica del estado del producto, no describen de la misma manera el estado de producción, el cual recibe especial énfasis en la propuesta descrita en este documento.

Para describir la evolución dinámica se tiene como fundamentación teórica el aporte de los sistemas de eventos discretos, bien por medio de autómatas [11] o bien por medio de redes de Petri [6] o, incluso, una combinación de ambos. El modelado conceptual de la unidad de producción se lleva a cabo con el uso de un lenguaje de modelado, como el UML propuesto por Booch, Rumbaugh y Jacobson [8]. Otros autores como Fishwick [4] establecen que el modelado conceptual orientado a objetos es un paso para hacer proyectos de simulación de la dinámica del sistema industrial, objeto de estudio. Otros van más lejos y proponen sistemas que controlan procesos productivos industriales.

Como base para obtener una descripción se utiliza el concepto de morfología, que es un método multidimensional de clasificación. Como ejemplo se tiene la tabla periódica en la que se describe cada elemento con sus propiedades, estructura y forma de relacionarse con su entorno. La idea es obtener una morfología que ayude a caracterizar el estado de un proceso industrial que tiene insumos, método de producción, producto final, equipamiento y servicios, entre otros.

3. MODELADO CONCEPTUAL DEL COMPORTAMIENTO DE UN SISTEMA SUPERVISADO

Por comportamiento de un sistema se concibe la manera o el modo en el que actúa y reacciona cualquier actividad que se registre y pruebe. Dentro del contexto de la automatización, se refiere a la descripción de lo que hace para llevar a cabo un proceso de producción y cómo cambian sus atributos de interés en el tiempo. Un modo para capturar el comportamiento del sistema es por medio de los casos de uso. Estos diagramas UML describen al sistema y sus relaciones con su entorno. En la Fig. 3 se muestra, para cada unidad de producción, un diagrama de caso de uso general, en el que se han identificado los actores y casos de uso que intervienen en la supervisión. Por actor se entiende que es todo lo que interactúa con el sistema, bien sea una máquina o una persona. Por caso de uso se establece que es una secuencia de actividades ejecutadas por el sistema y que produce valores medibles para un actor particular. El objetivo de este diagrama es presentar la funcionalidad del sistema supervisado y como base del modelado conceptual estructural y dinámico de la unidad de producción.

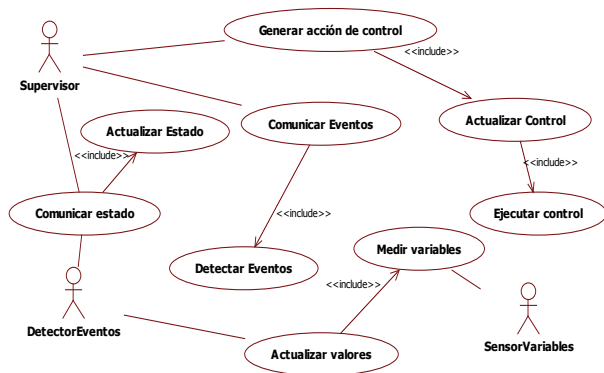


Fig. 3. Caso de uso general

Como se puede observar en la Fig. 3, se han identificado cuatro actores posibles dentro del holón de producción: sensor, detector de eventos, supervisor y actuador. Estos cuatro actores son roles que desempeña el sistema de automatización industrial en un momento determinado, y se consideran como sistemas autónomos, gestionados por unidades autónomas denominados agentes. La manera en que estos actores utilizan las funcionalidades del sistema proporciona los casos de uso identificados en la figura y que se relacionan en la siguiente lista:

- Actualizar valores. Incluye el caso de uso para medir variables del proceso físico. El actor detector de eventos solicita al sensor que mida las magnitudes de las variables para controlar y registra los valores medidos.
- Detectar eventos. Ocurre cuando el actor supervisor le solicita al actor detector de eventos si ha ocurrido alguno, lo que desencadena un flujo de eventos que incluye el caso de uso comunicar eventos cuando el detector da cuenta al supervisor de la ocurrencia de algún evento.
- Actualizar estado. Lo desencadena el actor supervisor cuando se ha detectado algún evento. La imagen se actualiza y el supervisor comunica el nuevo estado al detector de eventos, para dar cuenta del cambio de condiciones de operación. Para ello se incluye el caso de uso comunicar estado.
- Generar acción de control. Es un caso de uso que inicia el supervisor cuando el sistema se encuentra en un estado determinado y desea llevarlo a otro estado o iniciar una secuencia de eventos que impidan que el sistema llegue a un estado no deseado. Por medio de este caso de uso el actor supervisor le envía comandos al actor actuador. El caso de uso actualizar control está incluido en la generación de acción de control.
- Ejecutar acción de control. Es un caso de uso que utiliza el actor actuador cuando ya tiene un comando para ejecutar que le ha indicado previamente el actor supervisor.

Para describir las funciones de la unidad de producción se han establecido unos flujos de eventos para los casos de uso principales, de manera que se explique la interacción entre actores y casos de uso y detallar los objetos que intervienen en esta interacción y tener una idea de la estructura computacional que va a implementar una unidad de producción holónica. Para ello se usarán los diagramas de secuencia que proporciona el lenguaje de modelado UML [8].

3.1 Flujo de eventos en la detección de eventos

En la Fig. 4 se observa el flujo de eventos básico para la medición de variables, en el que se muestra que el puerto de un sensor constantemente (iterativamente) está midiendo variables físicas asociadas con el proceso industrial por medio de algún dispositivo físico. Este puerto comunica al objeto sensor, que, a su vez, envía valores de la lectura al detector de eventos, que

asume los roles del actor sensor, que es un dispositivo electrónico con capacidad de comunicación. Es posible hallar flujos de eventos alternos, como por ejemplo cuando el sensor no funciona correctamente; en este caso, no se registra la lectura sino una condición de falla que desencadena otros eventos. El detector obtiene el valor de las variables continuas y aplica los algoritmos de detección de eventos para determinar si han ocurrido cambios en las condiciones de operación. Si han ocurrido cambios informa al supervisor sobre la naturaleza del evento y éste se encarga de actualizar la imagen que representa el proceso que se está controlando. Cuando se actualiza la imagen se informa al actor detector para que las próximas ejecuciones del algoritmo de detección de eventos se hagan con base en este nuevo estado. El supervisor, cuando recibe la notificación de ocurrencia de un evento, ha actualizado el estado del proceso y solicita el estado en que se encuentra el control, con el fin de ejecutar un algoritmo que le permita derivar una acción que se pueda ejercer sobre el control y sobre el proceso. Si ha derivado esta acción, la convierte en un conjunto de comandos que envía al actuador para su ejecución. El actuador ejecuta estos comandos y notifica al supervisor de su ejecución, para que este último actualice los estados del control. Escenarios alternos: el actuador no logra ejecutar los comandos, el algoritmo de control no produce acciones ejecutables, o no se detectaron eventos. Dos objetos muy importantes que son la base de este artículo se encuentran en este diagrama de secuencia: el estado del proceso y el estado del control.

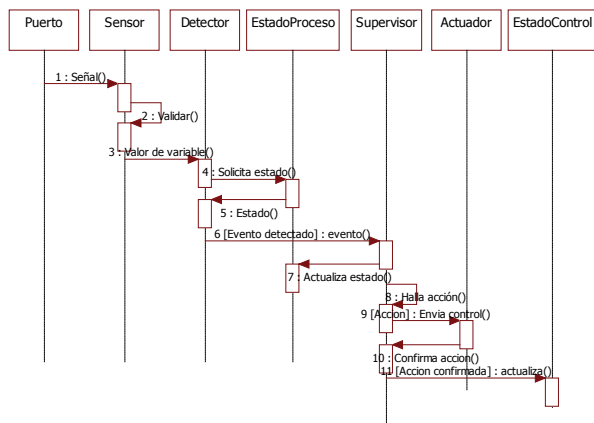


Fig. 4. Diagrama de secuencia de la medición de variables

Varios casos alternos pueden ocurrir cuando no se detectan eventos. En este caso, el detector le informa al supervisor que no han ocurrido eventos. También se puede dar el caso de que no haya acciones candidatas

por parte del supervisor, o no existen mediciones confiables (falla en sensores) de manera que no se registre el valor de una variable determinada.

3.2 Observación del proceso por parte de un holón agregado

En la Fig. 5 se aprecia el diagrama de secuencia que describe este flujo de eventos. El flujo principal de eventos es el siguiente: uno de los holones básicos envía un evento al objeto observador del holón agregado, que actualiza su estado abstracto y confronta el estado actual con el evento recibido. Un coordinador determina si hay acciones candidatas para mantener la trayectoria de estados, y para esto genera una acción que comunica al holón básico (o a otros holones básicos) y una vez recibe confirmación de esta acción actualiza el estado del control.

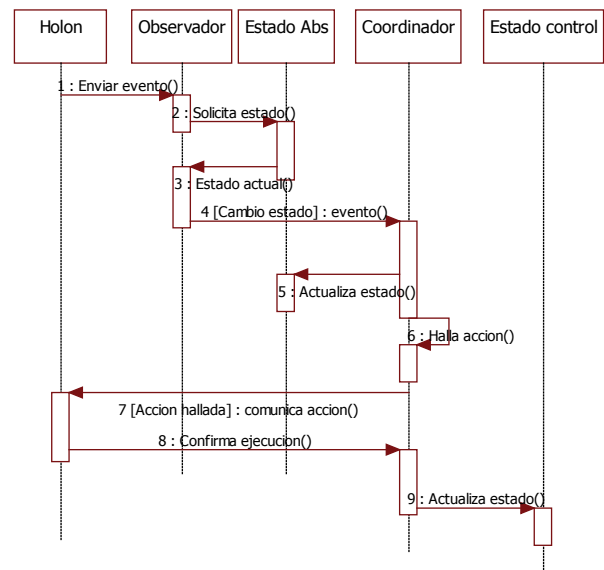


Fig. 5. Diagrama de secuencia de la observación del proceso por el holón agregado

4. MODELANDO LA CONFIGURACIÓN ESTÁTICA DE LA UNIDAD DE PRODUCCIÓN AUTÓNOMA

Para modelar la estructura computacional de una unidad de producción se recurre a los diagramas de clases [8], en los que se representan los objetos abstractos que manejan información relacionada con el funcionamiento de esta unidad. Para entender mejor estos diagramas se incluye una figura para mostrar los aspectos presentes en estos diagramas. Esto se muestra en la Fig. 6. El dibujo general de las clases representa al modelo de la estructura estática de un sistema, y las relaciones entre ellas. Los rectángulos

representan a las clases, que son plantillas de objetos, y las líneas representan las relaciones de agregación (componente/parte), especialización (herencia) y asociación. Esto ayudará a comprender mejor los diagramas subsiguientes, en los que se señalan las clases que intervienen dentro de una unidad de producción y sus relaciones.

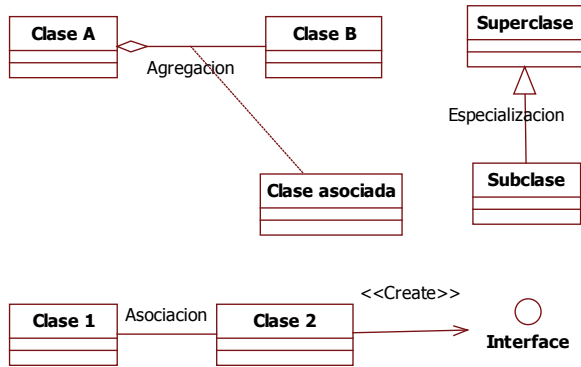


Fig. 6. Esquemas para interpretar un diagrama de clases

Una descripción basada en el lenguaje unificado de modelado (UML) para el estado del proceso se propone en esta sección y en las subsiguientes, para establecer un diseño que lleve a una posterior implementación del sistema holónico. La Unidad de Producción Autónoma (UPA de aquí en adelante) es un tipo particular de Holón, donde la información está relacionada con la disponibilidad de los recursos, objetivos de producción, métodos de producción y la dinámica del proceso continuo. Se asume que la UPA está basada en la arquitectura holónica de referencia PROSA [10], y el diagrama de clases que relaciona esta arquitectura con la propuesta de supervisión de la UPA se muestra en la Fig. 7. Se aprecia que el punto de partida son los holones *Recurso*, *Producto* y *Orden*, los cuales son especializaciones de la *clase holón*. Una UPA se compone de estos tres holones básicos, pero cuando se refiere a la supervisión de un holón básico se concentra en la información que generan las relaciones entre estos holones. Por ejemplo, a partir de la asociación entre holón *Recurso* y *Producto* se tiene la clase *Tareas*, que mantiene la información sobre el estado de los recursos, las tareas que hay que desarrollar en un recurso para obtener un producto, como se ejecuta un proceso en un recurso y las capacidades del recurso, entre otras. La clase *MetodoProduccion* surge de la asociación entre holones *Recurso* y *Orden* y

maneja información sobre cómo obtener un producto, dados ciertos recursos y las secuencias que se deben ejecutar. La clase *EstadoProceso* resulta de la relación entre *Recurso* y *Orden*, y contiene la información asociada con el progreso de la ejecución de los diversos procesos, como iniciar, reservar recursos, monitorear, interrumpir un proceso y reanudarlo, entre otras. El mecanismo de supervisión de la UPA está relacionada con estas tres clases, y a continuación se describe su estructura y comportamiento, así como su aplicación a sistemas de producción continuos.

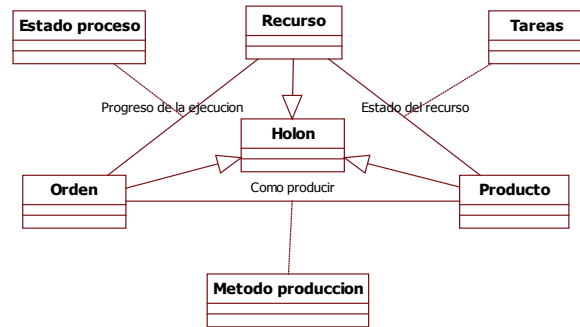


Fig. 7. Diagrama de clases para la Unidad de Producción dentro de la arquitectura PROSA.

4.1 Unidad de Producción y su relación con la configuración de la planta

La clase *Unidad de producción* se compone de otras clases que manejan información sobre el estado de la unidad de producción, los métodos de producción, los recursos, el supervisor, asociado con una misión de producción. En la Fig. 8, adaptada de [2], se puede apreciar la relación que tienen y se establece que la misión está asociada con la clase *Supervisión*, cuya función es mantener el estado de la unidad de producción dentro de los estados deseados. Estas clases son: *Recursos*, que maneja la información asociada con los recursos para desarrollar el proceso productivo, cuya información se describe en la clase *Método de producción*. La información asociada con el estado de la unidad de producción se encuentra en la clase *Estado unidad de producción*, y el sub-sistema de control para mantenerlo dentro del comportamiento deseado se proyecta dentro de la clase *Supervisor*. De la interacción resultante entre el *Supervisor*, los *Recursos* y el *Método de producción* se obtiene la clase *Configuración*, que describe cómo está organizada la planta en un instante deter-

minado para ejecutar una tarea. Además, se advierte que la clase *Unidad de producción* contiene otras unidades de producción y un estado propio que se debe monitorear.

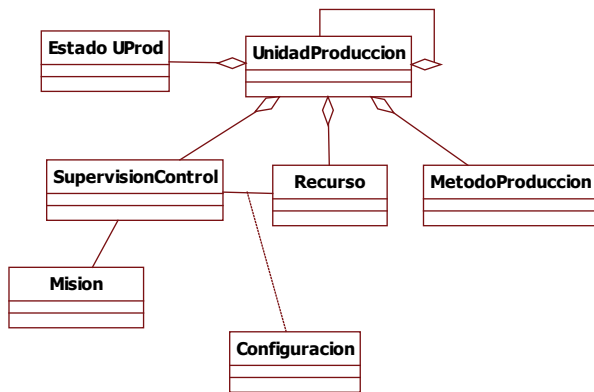


Fig.8. Diagrama de clases para la Unidad de producción.

4.2 Supervisión y control del Estado de la unidad de producción

Como se planteó anteriormente, la Unidad de producción debe monitorearse para que exhiba una trayectoria de variables deseadas. Las teorías de control, supervisión y detección de eventos son útiles para llevar a cabo la función de monitoreo y, por lo tanto, deben considerarse en la implementación del sistema de información que registra el comportamiento de la Unidad de producción. Para supervisar el comportamiento de esta unidad de producción, es necesario conocer su estado interno y para ello se consideran las clases *Estado unidad de producción* y *Supervisión y Control* en una asociación que permite relacionar los mecanismos de control con los estados en los que se encuentran los recursos y la productividad. El estado del supervisor también es un componente de la unidad de producción. Unas clases que son indispensables para controlar la evolución dinámica de la unidad de producción son las que implementan al supervisor, detector de eventos y actuador, las cuales interactúan con el estado global de la unidad de producción y con el estado de las actividades de control, y son parte de la clase *Configuración* y control, que actualiza la configuración de la planta.

- La clase *Recurso* tiene la información sobre los recursos que se van a utilizar.
- La clase *Configuración* maneja las configuraciones para lograr la meta de producción, dadas las condiciones de los recursos existentes y el método de producción seleccionado.
- La clase *Orden* maneja información sobre lo que se va a producir en cuanto a flujo y calidad del producto.
- La clase *Estado unidad de producción* describe las condiciones en las que se encuentra la unidad de producción en un momento determinado. Se compone de la clase Estado Recurso, que describe en qué estado se encuentra el recurso, y la clase Estado del proceso, que resulta de la relación entre los recursos y la orden de producción.
- La clase *Supervisión y Control* controla la ejecución del proceso productivo por parte de la unidad de producción, procura cumplir el objetivo de producción con las restricciones impuestas por los recursos y el método de producción, y determina una configuración que debe aplicar la clase *Actuador*, que es uno de sus componentes. La clase *Detector* obtiene el estado de la unidad de producción, y por medio de la medición de variables asociadas con el proceso indica a la clase *Supervisor* si ocurrió un evento que haga necesario cambiar su imagen. Otro componente es la clase *Estado del Control* que describe las medidas que debe tomar el supervisor ante la ocurrencia de un evento determinado, y el estado actual en el que se encuentran las medidas del control supervisorio.
- La clase *DES* describe a un sistema de eventos discretos y establece un marco común para describir los estados de los recursos y del control.

Conocer el estado de la Unidad de producción facilita la supervisión de su comportamiento, siempre y cuando exista un supervisor acoplado a esta Unidad de producción que ejecute ciertas acciones. Este supervisor es un sistema de eventos discretos, y tiene un estado interno, que consideraremos como el estado del control.

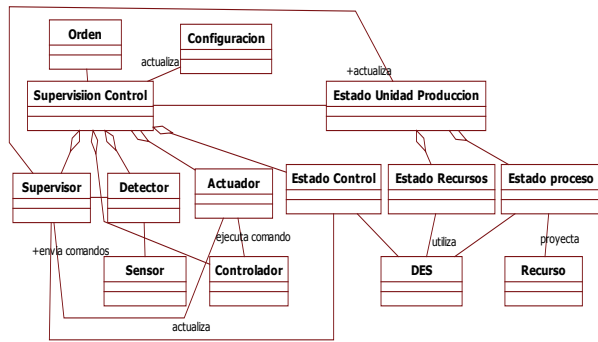


Fig.9. Diagrama de clases para representar la configuración estática de una unidad de producción.

Las relaciones antes mencionadas se observan en la Fig. 9, que muestra que las clases que representan al estado del proceso y al estado del control heredan características de la clase *DES*, diseñada para representar sistemas a eventos discretos de diferente naturaleza y sus componentes más importantes son los Autómatas de estados finitos, y las redes de Petri. Se profundiza en esta sección sobre lo que se considera como estado de los recursos.

4.3 El Estado de los recursos

Como Recursos se considera el equipamiento para ejecutar las operaciones productivas, la energía, los servicios de transporte, por ejemplo, y el recurso humano para ejecutar las actividades de producción. Un atributo en común de estos recursos es la unidad de medida porque establece cómo se cuantifica la cantidad o disponibilidad de un recurso. Implican restricciones sobre el proceso de producción y se proyectan a una configuración definida para llevar a cabo un método de producción determinado. Conocer el Estado de la UPA permite que el supervisor controle esta unidad y, a su vez, mantenga actualizada su imagen. Una breve descripción operacional tiene la siguiente forma: la clase *estado Recursos* recibe información de los estados de los recursos que están operando por medio de la clase supervisor, y algunos recursos interactúan con la clase detector de eventos a través del sensor que informa al supervisor sobre la ocurrencia de eventos que cambien este estado, con el fin de que ejecute acciones para mantenerlo con el comportamiento deseado. Otros atributos y operaciones de esta clase conforman otras clases, cuyo diagrama UML estático se ve en la Fig. 10.

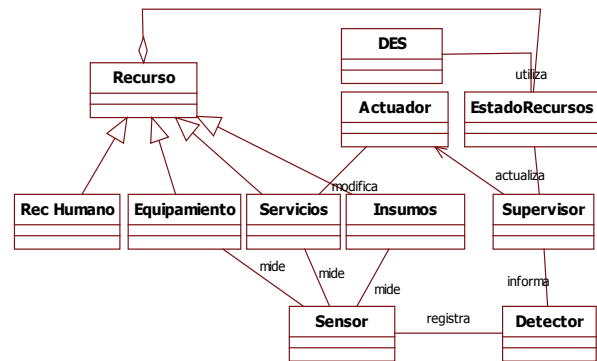


Fig.10. El Estado de los Recursos.

- Clase *RecHumano* describe el estado en el que se encuentra el recurso humano asociado con lo productivo. Tiene los atributos *Identificador*, y *Rol asignado*.
- Clase *Equipos* maneja la información concerniente al equipamiento involucrado en la producción. Sus principales atributos son fabricante, fecha de compra, fecha mantenimiento, número de partes, ubicación, que describen el equipamiento, la capacidad que maneja este equipo (potencia eléctrica, peso máximo, potencia de carga, volumen) y el valor asociado con esta capacidad. Las operaciones que maneja son: la lectura sobre los puertos de comunicación, que es una subclase que sirve para relacionar el estado del equipamiento con la clase *Sensor*, que, además, ejecuta las funciones de medir variables que se desea controlar y las guarda en una estructura de datos. Además de las variables y los valores para medir, es importante conocer el período de medición del sensor y un mecanismo para variarlo, como *setPeriod()*, por ejemplo.
- Clase *Servicios* maneja información concerniente con los servicios utilizados para llevar a cabo el proceso productivo. Dentro de sus atributos están la capacidad nominal, la confiabilidad del servicio, el costo por utilizar una unidad de servicio y la equivalencia entre la unidad de servicio y el trabajo realizado. Esto permitirá ejecutar las operaciones de calcular el costo de uso del servicio. En general, cada equipamiento está asociado con el uso de por lo menos un servicio, con el consiguiente costo asociado por hacer funcionar un equipo determinado.

- Clase *Insumos* maneja información sobre el material que fluye sobre los otros recursos y se mide en términos de cantidad, masa, volumen o volumen por unidad de tiempo.

4.4 Eventos en la operación de los Recursos

En la Fig. 11 se muestra una secuencia de eventos sobre cómo opera la supervisión de un recurso en particular. Un detector de eventos recibe datos periódicamente de un controlador, que, a su vez, los obtiene del recurso. Cuando este recurso es supervisado el detector de eventos indica al supervisor la ocurrencia de un evento. El supervisor produce una acción que ejecuta el actuador y modifica la configuración del recurso, y luego actualiza el nuevo estado del recurso. Dentro de un sistema holónico, los holones recurso y los de orden interactúan y producen el proceso, cuyo estado se está supervisando. De manera que esta dinámica no está aislada del contexto de producción. Esta secuencia de eventos se representa por medio de un sistema de eventos discretos, cuya dinámica se describe con el uso de algunas estructuras matemáticas y formales.

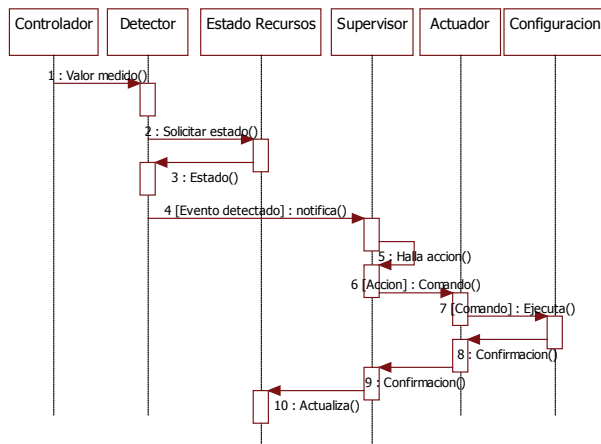


Fig.11. Secuencia de actividades dentro de un Holón Recurso.

4.5 Estado del proceso

Una orden de producción resulta de la relación entre los holones Orden y Recurso. El primero aplica un método de producción que requiere que reserve algunos recursos para ejecutarlo. La clase *EstadoProceso* indica en qué etapa se encuentra el método de producción y proporciona una medida del producto obtenido, bien en términos de unidades ab-

solutas o en términos de flujo por unidad de tiempo. El comportamiento dinámico de esta clase lo describimos por medio de la clase DES. Un método que se aplica es el de *cambiar()*, en el que se registra la evolución sobre cómo se ejecuta la orden de producción. La Fig. 12 ilustra cómo se relacionan las clases antes mencionadas con las clases que ejecutan la supervisión y el control de la unidad de producción: el supervisor, detector de eventos, sensor y actuador. Para monitorear el sensor periódicamente mide las variables del proceso productivo y las notifica al detector de eventos, que establece la ocurrencia de algún evento que informará al supervisor. Este supervisor genera una orden de producción que, en el caso de los sistemas continuos, implica aplicar nuevas leyes de control. Esta orden de producción cancela la orden que se ejecutaba hasta ahora. El supervisor cambia la configuración del sistema por medio del actuador. La información asociada con las actividades cuya ejecución se monitorea se maneja por medio de la clase actividades, donde a cada tarea individual se le asocia un tiempo de duración (en caso de que sea predeterminada), condiciones para ejecutar la tarea y cuáles tareas son habilitadas luego de su ejecución, es decir, las postcondiciones.

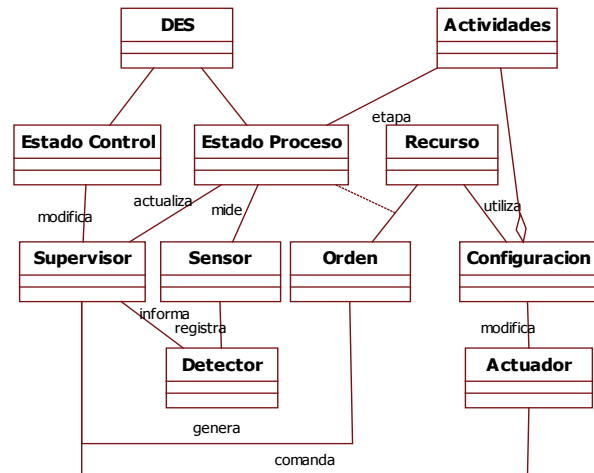


Fig.12. Clases utilizadas para describir el Estado del proceso.

4.6 Dinámica del Control supervisorio y los holones

Si se considera la ejecución del proceso de producción como un sistema dinámico, se pueden obtener aportes de la teoría de Control supervisorio propuesta por Wonham y Ramadge. Esta teoría establece

que la dinámica de un sistema evoluciona de acuerdo con la ocurrencia de eventos y matemáticamente se describe de la siguiente forma:

$$\Phi_D : X_D \times \Sigma \rightarrow X_D$$

X_D es el conjunto de estados de naturaleza discreta en que puede encontrarse el sistema en un momento dado, mientras que Σ es el conjunto de eventos que ocurren y ocasionan que el sistema cambie de un estado a otro. Este conjunto Σ está conformado por eventos controlables y no controlables, ambos disyuntos entre sí, es decir, un evento no puede ser controlable y no controlable a la vez. Los eventos controlables son aquellos que pueden generarse/inhibirse por un supervisor, mientras que los eventos no controlables dependen de la evolución del sistema dinámico, y generalmente están asociados con cambios de las variables de estado del sistema cuando pasan de una condición de operación a otra. Si se denota a los eventos controlables como Σ_c y a los eventos no controlables como Σ_{nc} se expresan estas relaciones como $\Sigma = \Sigma_c \cup \Sigma_{nc}$, y $\Sigma_c \cap \Sigma_{nc} = \{\}$. La función Φ_D matemáticamente corresponde a una función de transición, pero que en el campo de los sistemas de producción se hace corresponder con la aplicación de una ley de control, que se asocia con una configuración de la planta, particular para un estado determinado.

La configuración de la planta resulta de la relación entre el Holón orden, (objetivo de producción para un producto dado). La información de los recursos humanos, equipos, insumos y del avance del objetivo. El estado de cada uno de los componentes está enmarcado dentro del conjunto de modelos que describen el comportamiento de cada componente del Holón. El comportamiento global resulta de la composición de la dinámica de los componentes particulares, y se describe como la composición de un DES, que se muestra en la siguiente ecuación:

$$\Phi_k(\Phi_{k-1}(\dots \Phi_1(\Phi_0(X_R, X_P, t_0, t_1)) \dots))$$

X_R y X_P son los estados de los recursos y del proceso de producción, respectivamente. La dinámica del control supervisorio determina cuál es la función Φ_i para aplicar en el intervalo (t_{i-1}, t_i) , hasta que la evolución del sistema indique que el supervisor deba aplicar otra configuración que corresponda a la función Φ_j , o que el sistema dinámico evolucione hasta llegar a ese estado. La aplicación de una configura-

ción la asociamos con la generación de una orden de producción por parte del supervisor, es decir, cada vez que sea necesario el supervisor interactúa con los diversos holones de la unidad de producción, y la secuencia de esta interacción se ilustra en la Fig. 13.

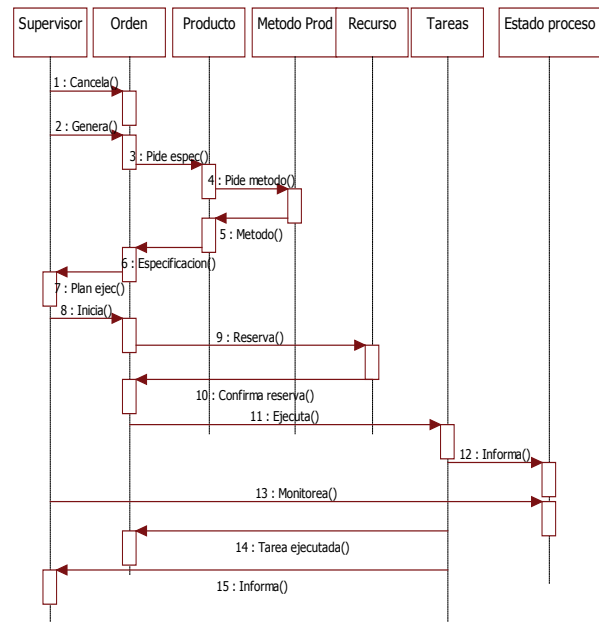


Fig.13. Secuencia de interacciones entre Supervisor, holones y clases asociadas.

Como se observa en la figura, el proceso inicia cuando el Supervisor cancela la orden actual y en su lugar crea una nueva orden (tal y como funciona en los sistemas continuos de un solo producto), para lo cual interactúa con el Holón Orden, que informa sobre las nuevas necesidades de producción. Este holón solicita al Holón Producto sobre las especificaciones del producto para producir, y el Holón Producto, a su vez, indica el método de producción que se aplicará. Con esta información, el Holón Orden informa al supervisor sobre cuál es su plan de ejecución. Una vez el Supervisor inicia la ejecución del proceso, el Holón Orden reserva los Recursos para ejecutar el método de producción seleccionado y empieza a ejecutar tarea por tarea. El Supervisor monitorea y actualiza el estado de su ejecución hasta que sea necesario generar otra orden.

5. LA DINÁMICA DE LOS SISTEMAS A EVENTOS DISCRETOS - DES

Para controlar una planta en la que ocurre un proceso industrial hay que llevar a cabo dos pasos: la

representación de su dinámica y la síntesis de un supervisor para procurar que exhiba una conducta deseada o, al menos, prevenir que ocurra una conducta no deseada. La dinámica del proceso industrial se representa por medio de un Sistema a Eventos Discretos (SDES), donde cada estado corresponde a un modo de operación, mientras que los eventos se consideran como el cambio de un estado a otro. Esta dinámica está muy relacionada con la evolución continua de las variables que rigen al proceso continuo, por lo que es necesario acoplar el SDES con el comportamiento de estas variables continuas. En todo proceso industrial las etapas y las condiciones de operación se expresan cualitativamente como estados, y el cambio de un estado a otro se denomina una transición. Si se etiqueta cada transición con un símbolo, el comportamiento de un proceso industrial se representa por medio de una secuencia de símbolos que representan las transiciones que hayan tenido lugar. Dentro de esta sección se muestra la síntesis del supervisor y la gestión del Holón.

Una manera de representar los Sistemas a Eventos Discretos de tal forma que permita su implementación dentro de una arquitectura computacional y de comunicaciones es por medio del diagrama estático UML de la Fig. 14. La clase principal, *SisEventosDiscretos*, está asociada con dos grandes clases: por un lado está la clase que describe a los Automatas de estado finito, que se denomina *AEF*, y, por otro lado, está la clase que describe a los SDES como una red de Petri, que es la clase *PetriNet*. Cada una de estas clases está diseñada para cumplir con la especificación formal de un autómata y una red de Petri.

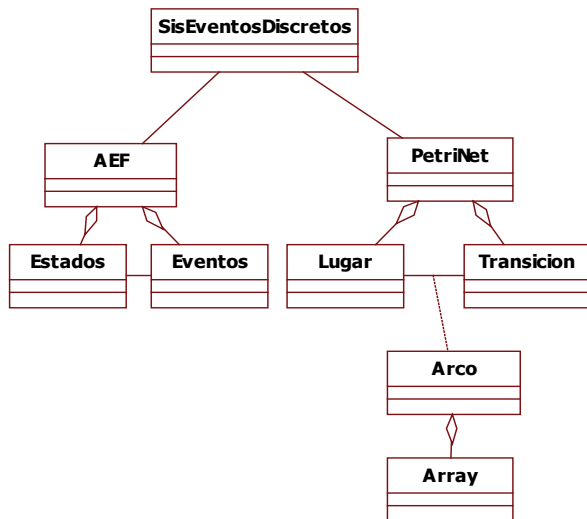


Fig.14. Diagrama de clases que implementa a un sistema de eventos discretos.

Antes de continuar con la especificación formal y estática de un AEF definido formalmente como un autómata, es necesario llevar a la formalización la definición de evento. Por evento se concibe el cambio de estado de un sistema y ocurre en un punto determinado en el tiempo. Generalmente, ese evento está asociado con alguna actividad que se despliega dentro del sistema y, si se denota cada evento por una letra de alfabeto, la evolución de un sistema crea un lenguaje, y así se obtiene una forma compacta de describir la dinámica de cualquier sistema. Una máquina G de estados finitos es una quintupla que tiene la forma [9]:

$$G = (X, \Sigma, f, I, X_m)$$

Donde:

X es un conjunto finito de estados denotados por x , x es discreto.

Σ es un conjunto de eventos de entrada, denotado cada uno con una letra de un alfabeto.

$f : X \times \Sigma \rightarrow 2^X$ es la función de transición o evolución del autómata

$I \subset X$ es el conjunto de estados iniciales

$X_m \subset X$ es el conjunto de estados deseados de G (estados marcados)

La clase *AEF* tiene como atributos el estado inicial, estado actual, y el evento entrante actual, por medio de los cuales implementa la operación *evolución* (), que actualiza a nuevos estados de acuerdo con el evento introducido. Los estados y eventos discretos conforman dos subclases del mismo nombre, y asociados con la clase principal *AEF*. Los eventos tienen otros atributos adicionales que permitirán su posterior control: unos indicadores de que sí son observables directamente y, a su vez, sí son controlables.

Otro enfoque para representar la dinámica discreta es el uso de Redes de Petri (o RdP, de aquí en adelante), ya que las especificaciones y la marcación de un sistema de producción se diseñan en la estructura de la Red de Petri. Las transiciones representan el comienzo o fin de una actividad, mientras que los lugares representan una operación o un estado del proceso. Los tokens, conocidos también como la marcación, son números enteros asignados a cada uno de los lugares y representan una disponibilidad o no disponibilidad para el status de un recurso o una cantidad para una operación.

Una Red de Petri es una quintupla [7], $PN = (P, T, F, W, M_0)$ donde:

$P = \{p_1, p_2, \dots, p_n\}$ es un conjunto finito de lugares
 $T = \{t_1, t_2, \dots, t_m\}$ es un conjunto finito de transiciones
 $F \subset (P \times T) \cup (T \times P)$ es un conjunto de arcos que relacionan lugares con transiciones y viceversa
 $W : F \rightarrow \{1, 2, 3, \dots\}$ es una función de asignación de peso (ponderación)
 $M_0 : P \rightarrow \{1, 2, 3, \dots\}$ es la marcación inicial
 $P \cap T = \emptyset$ y $P \cup T = \emptyset$

Para su implementación computacional, la clase *PetriNet* incorpora dos subclases: *Lugar* y *Transición* que reflejan al conjunto de lugares y de transiciones, respectivamente. Estas dos subclases están asociadas por medio de la clase *Arco*, la cual relaciona lugares con transiciones e implementa la función de asignación de pesos. Esta clase agrupa los pesos dentro de una matriz para obtener la matriz de incidencia, que utilizaremos para implementar las principales funciones de los lugares y las transiciones, a saber: el disparo de las transiciones (método *dispara()*) y la asignación de marcas a los lugares (función método *marca()*).

6. CONCLUSIONES

A partir de la confluencia de tres grandes áreas, como lo son las arquitecturas de referencia de automatización, la teoría de supervisión de procesos industriales y los métodos de modelado de sistemas, más experiencias obtenidas en la investigación sobre estas áreas, podemos llegar a la conclusión de que se puede hallar el estado en el que se encuentre para llevarlo a un esquema de persistencia de objetos que se implementa por medio de una base de datos y aplicaciones. Por medio de la aplicación de teoría de sistemas supervisorios se puede sintetizar una supervisión de una unidad de producción compleja por medio de la agregación de supervisores de unidades de producción básicas.

Si se pueden implementar en forma computacional sistemas a eventos discretos, entonces se reutiliza esta implementación para representar estados de los recursos, del proceso productivo y del control, simplificar su diseño y permitir su posterior escalamiento para sistemas de producción de mayor complejidad. Por otro lado, la dinámica de este estado del proceso se valida matemáticamente y por medio de simulación, lo que permite implementarla en un sistema productivo real.

REFERENCIAS

- [1] Blanc, P.: Pilotage par Approche Holonique d'un Systeme de Production de Vitres de Securite Feuilletées. Thèse de Doctorat. Université de Nantes, 2006.
- [2] Chacón, E., Besembel, I., Hennet, J.: Coordination and optimisation in oil and gas production complexes. *Computers in Industry* (bf 53) 17–37, 2003.
- [3] Chacón, E., Besembel, I., Narciso, F., Montilva, J., Colina, E. An Integration Architecture for the Automation of a Continuous Production Complex. *ISA Transactions* Vol. 41 No. 1 Jan. pp. 95-113. 2002.
- [4] Fishwick, P.: Simulation Model Design and Execution. Building digital worlds. Prentice Hall International Series in Industrial and Systems Engineering. 1995.
- [5] Larsen, M., Sorensen, C., Langer, G.: Development of a Production Meta Product State Model. Technical University of Denmark. 1999.
- [6] Moody, J., Antsaklis, P.: Supervisory Control of Discrete Event Systems using Petri nets. Kluwer Academic Publishers. 1998.
- [7] Murata, T.: Petri Nets, Properties, Analysis and Applications. *Proceedings of the IEEE*, vol 77, No. 4. 1989.
- [8] OMG.: The Unified Modeling Language. web: www.omg.org 1998.
- [9] Ruiz, M.: Control y Supervisión de Sistemas Dinámicos de Eventos Discretos. Tesis de Licenciatura, Universidad de los Andes, Facultad de Ciencias. 2001.
- [10] Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P.: Reference architecture for holonic manufacturing systems. *Computers in industry*. 37 255–274. 1998.
- [11] Ramadge R, J. and Wonham W.: Supervisory control of a class of discrete event processes. *SIAM J. Control and Optimization*, 25(1), pp. 206 - 230, 1987.
- [12] Wyns, J.: Reference Architecture for Holonic Manufacturing Systems. Katholieke Universiteit Leuven, 1999.

FUZZY CONTROL STRATEGY APPLIED TO ADJUSTMENT OF FRONT STEERING ANGLE OF A 4WSD AGRICULTURAL MOBILE ROBOT

ESTRATEGIA DE CONTROL FUZZY APLICADA AL AJUSTE DE ÁNGULO DE DIRECCIÓN DELANTERA DE UN ROBOT MÓVIL 4WSD AGRÍCOLA

Ing. Henry Borrero Guerrero

Universidad de Sao Paulo, Brasil
Laboratorio de Robótica Móvil
(LabROM)
h_borrelo@ieee.org

Mg. Rafael Bueno Sampaio

Universidad de Sao Paulo, Brasil
Laboratorio de Robótica Móvil
(LabROM)
rafaelc@sc.usp.br

Ph.D. Marcelo Becker

Universidad de Sao Paulo, Brasil
Laboratorio de Robótica Móvil
(LabROM)
becker@sc.up.br

(Article of **Scientific and technological research**. Received el 17/10/2011. Accepted el 19/12/2011)

Abstract. This paper presents the preliminary studies of the control strategy based in fuzzy logic, projected for the steering system of AGRIBOT project that consist of a wheeled autonomous mobile robotic in real scale endowed with four independent steering and driven wheels (4WSD). In this work we present a preliminary fuzzy controller design applied to front steering angle, using a multivariable plant which incorporates simplified linear model of lateral dynamics of a vehicle whose input are linear combination of rear and front steering angles. The fuzzy control strategy was decided because provides flexible way to deploy with embedded systems. Simulations are used to illustrate the designed controller performance. We use Ackerman geometry to trace front steering angle that allows the vehicle to perform correctly a given maneuver preserving a minimum level of stability and maneuverability. The goal is to establish a relationship between steering input commands and the control commands to the actuators so that it is possible to adjust the attitude of the actuators over the movement axis, as the trajectory change.

Keywords: Ackerman geometry, fuzzy controller, Two-Track Vehicle, Lateral dynamics, mobile robot, bike model, steering angle.

Resumen. Este trabajo presenta los resultados preliminares de la estrategia de control basada en lógica difusa, proyectada para el sistema de dirección del proyecto Agribot, que consiste en un robot móvil autónomo con ruedas en escala real dotado de dirección a las cuatro ruedas motrices independientes (4WSD). Se presenta un diseño preliminar de un controlador difuso aplicado a ángulo de dirección frontal, utilizando una planta multivariable que incorpora el modelo lineal simplificado de la dinámica lateral de un vehículo, cuya entrada es la combinación lineal de los ángulos de dirección trasera como delantera. Se eligió como estrategia el control difuso debido a que proporciona una forma flexible para desplegar los sistemas embebidos. Las simulaciones se usan para ilustrar el rendimiento del controlador diseñado. Se acudió a la geometría Ackerman para trazar el ángulo de dirección delantera que permite que el vehículo funcione correctamente en una maniobra dada la preservación de un nivel mínimo de estabilidad y maniobrabilidad. El objetivo es establecer una relación entre la entrada de órdenes de dirección y los comandos de control a los actuadores de modo que es posible ajustar la actitud de los actuadores sobre el eje de movimiento, como el cambio de trayectoria.

Palabras clave: Geometría Ackerman; Controlador difuso; Vehículos de pista; Dinámica lateral; Robot móvil; Modelo de bicicleta.

1. INTRODUCCIÓN

Nowadays is evident the use of a wide range of technological advances in various knowledge fields, which has allowed the adaptation of techniques developed to outdoor mobile robotics in fields as the agriculture or including motorsports, increasing profitability and reducing environmental impacts.

Today, a significant number of research groups in the world are interesting in this area, mainly due to the need of increasing food production and reducing costs. Currently, a keyword in this area is widely known as precision agriculture that encompasses not only the automation of the processes, but also the planting data acquisition, detection and diagnosis of problems (pests, plants malformation, etc.), and customized treatment [1], [2]. The technological application in the precision agriculture crates a scenario when the mobile robots can be applied to optimize some laborious activities, e.g.: harvesting, spraying, seeding, and soil preparation.

Some interesting examples of mobile robots developed for application in agricultural fields can be found in [3], [4], [5], [6]. Nevertheless, many challenges must be overcome in order to become the use of robots economically viable in agriculture. A good example of challenge is the robot navigation system. An agricultural mobile robot needs an onboard path planner that calculates a collision-free trajectory sufficiently precise that enables the robot to execute its tasks without damaging the crop.

The AgriBOT is a cooperation project between EMBRAPA (Empresa Brasileira de Pesquisa Agropecuária), EESC – USP (Escola de Engenharia de São Carlos – Universidade de São Paulo), and Jacto Company with financial support of FINEP (Financiadora de Estudos e Projetos), CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), and COLCIENCIAS (Departamento Administrativo de Ciencia Tecnología e Innovación de la República de Colombia). These institutions have been designed and built a multifunctional and modular robotics platform (the AgriBOT robot prototype) to work in the Precision Agriculture area.

The AgriBOT mobile robot is a vehicle that has as the main power source a 4 stroke cycle turbocharged diesel engine manufactured by Cummins Inc, and has a fuel injection electronic system providing 59.65 kW (80cv) at 2200 RPM. The use of a diesel engine offers autonomy, which in this case can reach up to

20 hours, and ability to refuel quickly. The fuel tank has a capacity of 140 liters of diesel oil. There is also a power system of 12Vcc and 170 Ah, consisting of three batteries connected in parallel, totaling 510Ah, reefered for an alternator set with the diesel engine [6].

This paper presents a fuzzy front steering angle control projected to the AgriBOT mobile robot. This control structure is based on a simplified linear model of the lateral dynamics to control the orientation rate (yaw rate) and the slide slip angle. As was mentioned, the control design is based in the fuzzy logic theory and a multivariable plant which incorporates the model of the lateral dynamics mentioned above and whose input are the linear combination of the rear and front steering angles [7], [8]. The proposed control structure has been applied to design a sideslip and yaw rate controller using an accurate lateral dynamics model of a 4 wheel steering car. This model incorporates the tire force dynamics and the steering actuators. Simulations are used to illustrate the performance of the designed controllers. We present the synthesis and the analysis of a fuzzy controller to adjust the steering angles by acting over the angular position of a respective actuator.

2. AGRIBOT MOBILE ROBOT

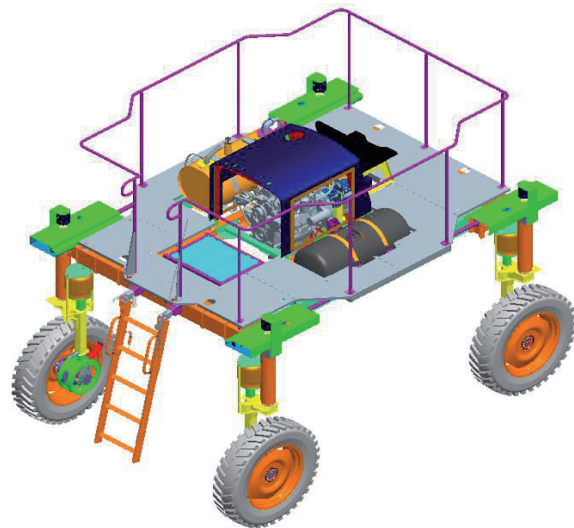


Fig.1. AgriBOT prototype. Pro-Engineer drawing

In Fig. 1 one may observe a Pro-Engineer drawing of the mobile robot platform and in Fig. 2 a photo of the same robot when it was delivered by Jacto Company in December, 2010.

The AgriBOT is a 4WSD vehicle because the basis for the robotic platform is the mobility capability provided by four wheel module mechanism that are powered by a diesel engine, each one of the four

identical wheel modules includes a hydraulic motor actuator for propulsion and other hydraulic motor for the steering, these actuators provides direct drive to create a two-degree-of-freedom system for each module.

Communication between electronics devices and electromechanical actuators is made using the famous Controller Area Network (CAN) protocol. The electronics wheel steering controller is based on a commercial agricultural job computer and handles the local position servo controller for the steering and provides torque control of the driven motor. The driver motor electronics allows speed and current (torque) feedback while the steering servo system provides a steering angle feedback [6].

The 4WSD configuration was introduced in order to provide a flexible platform the research, but the improved mobility also supplies a number of more practical benefits. By using 4WSD configuration, it is possible to produce parallel displacement of the vehicle during turns because it decouples adjustments in position from adjustments in orientation. It also allows both the front and rear of the vehicle to follow a specific path precisely and may maintain a fixed orientation relative to the crop rows. Steering on all wheels also minimizes side slip of the wheels resulting in reduced wear on the vehicle and less damage to the field.



Fig. 2. AgriBOT prototype. Photo of the robot when it was delivered by Jacto company in December 2010

3. SIMPLIFIED LATERAL DYNAMICS

The development of chassis control schemes has been a major area of study for automotive control engineers over the past 30 years. The volume of published literature is large, exceeding 1000 papers. Of this literature, there are 250 examining yaw and sideslip control [9].

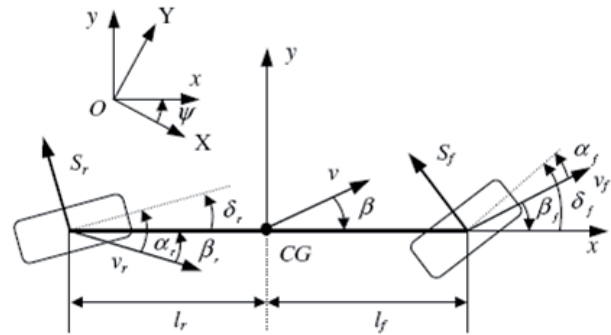


Fig. 3. Single-track model of a 4-wheel steering car.

Dynamics of 4-Wheel Steering Car Throughout this paper it is assumed that the essential features of the lateral dynamics of the car can be described using the single-track model show in [7], [8]. In the single-track model, the two front wheels are lumped into a single imaginary wheel located at the center of the respective axle; the same is done with the two rear wheels. The resulting front and rear wheels are interconnected by a one-dimensional rigid element with the car as show in Fig. 3 where the set of reference axes $CG-xy$, with origin at the centre of gravity CG , is fixed to the vehicle and $O-XY$ is an inertial reference frame; v is the velocity of the vehicle with respect to $O-XY$; v_f and v_r are the velocities at the front and rear axle, respectively, with respect to $O-XY$; ψ is the yaw angle and β is the sideslip angle. It is assumed that the front (and respectively, rear) steering angle of the single-track model, δ_f and respectively, δ_r in Fig. 3, corresponds to the steering angle at the two front and rear wheels. In this paper we are not concerned with the longitudinal motion of the single-track model; due to this we only consider tire-road interaction forces perpendicular to the wheel plane, i.e. cornering forces. The force S_f and S_r in Fig. 3, are the lateral tire forces on the front and rear wheels respectively.

In order to derive the equations governing the linearized lateral dynamics of the single-track model, assume that the front and rear steering angles are small, which in turn results in the angles β , β_f , α_f , β_r and α_r (as shown in Fig. 3) also being small. Under this premise, the application of the equations

of motion of a rigid body to the single-track model results in:

$$\dot{\beta} = \psi - \frac{S_f + S_r}{mv_x} \dot{\beta} = \psi - \frac{S_f + S_r}{mv_x} \quad (1)$$

$$\dot{\psi} = \frac{S_f l_f + S_r l_r}{I_{zz}} \psi = \frac{S_f l_f + S_r l_r}{I_{zz}} \quad (2)$$

Where m is the mass of the vehicle, I_{zz} is its moment of inertia with respect to the vertical axis, l_f (and respectively, l_r) is the distance from the center of gravity to the front (and respectively, rear) axle and v_x is the projection of the velocity vector along the CG - x axis, i.e. the vehicle longitudinal velocity, which we hereafter refer to as the vehicle speed. For small α_f and α_r values, S_f and S_r can be approximated by the following equations [7, 8]:

$$S_f = K_f \alpha_f S_f = K_f \alpha_f \quad (3)$$

$$S_r = K_r \alpha_r S_r = K_r \alpha_r \quad (4)$$

The constant K_f is the result of modifying the combined stiffness of the two front tires to take into account the caster effect of the steering system at the front axle. Since no caster effect is generated at the rear axle, the constant K_r is simply the cornering stiffness [7], [8]. We have to consider the caster effect as it is assumed that the front steer-by-wire function is integrated with a conventional steering system.

Considering the kinematics of the single-track model as a rigid body, the angles α_f and α_r are obtained as follows:

$$\alpha_f = \delta_f + \beta - \frac{l_f \dot{\psi}}{v_x} \alpha_f = \delta_f + \beta - \frac{l_f \dot{\psi}}{v_x} \quad (5)$$

$$\alpha_r = \delta_r + \beta - \frac{l_r \dot{\psi}}{v_x} \alpha_r = \delta_r + \beta - \frac{l_r \dot{\psi}}{v_x} \quad (6)$$

$$\begin{aligned} \dot{x} &= Ax + Bu \dot{x} = Ax + Bu \\ y &= Cx + Du y = Cx + Du \end{aligned} \quad (7)$$

$$A = \begin{bmatrix} -\frac{K_f + K_r}{mv_x} & \frac{K_f l_f - K_r l_r}{mv_x^2} + 1 \\ \frac{K_f l_f - K_r l_r}{I_{zz}} & -\frac{K_f l_f^2 + K_r l_r^2}{I_{zz} v_x} \end{bmatrix}$$

$$B = \begin{bmatrix} -\frac{K_f}{mv_x} & -\frac{K_r}{mv_x} \\ \frac{K_f l_f}{I_{zz}} & -\frac{K_r l_r}{I_{zz}} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

This construction allows for the introduction of a safety management system that reverts to normal steering in case of failure of the steer-by-wire function.

Equations (1), (2), (3), (4), (5), and (6) can be rearranged into the state-space representation of a linear time-invariant system with two inputs (δ_f and δ_r) and two outputs (β and ψ). The resulting state-space representation is given is show in (7).

The linear time-invariant system introduced above describes the lateral dynamics of the single-track model around the trajectory given by zero sideslip, zero yaw rate, zero steering angles and constant vehicle speed [7, 8].

4. FUZZY LOGIC GENERALITES

The Fuzzy Logic tool was introduced in 1965, also by Zadeh, and is a mathematical tool for dealing with uncertainty. It offers to a soft computing partnership the important concept of computing with words. It provides a technique to deal with imprecision and information granularity. The fuzzy theory provides a mechanism for representing linguistic constructs such as many, low, medium, often, few. In general, the fuzzy logic provides an inference structure that enables appropriate human reasoning capabilities. On the contrary, the traditional binary set theory describes crisp events, events that either do or do not occur. It uses probability theory to explain if an event will occur, measuring the chance with which a given event is expected to occur [10].

Real world situations are too complex, and this complexity involves the degree of uncertainty – as uncertainty increases, so does the problem complexity. Traditional system modeling and analysis techniques are too precise for such problems (systems), and in order to make complexity less daunting we introduce appropriate simplifications, assumptions, etc. (i.e., degree of uncertainty or Fuzziness) to achieve a satisfactory compromise between the information we have and the amount of uncertainty we are willing to accept. In this aspect, fuzzy systems theory is similar to other engineering theories, because almost all of them characterize the real world in an approximate manner. For more details about the concepts related to fuzzy logic and its application with Matlab™, you should follow the following reference [10].

5. CONTROL SYSTEM ARCHITECTURE

The utility of fuzzy sets lies in their ability to model uncertain or ambiguous data, so often encountered in real life. It is important to observe that there is an intimate connection between Fuzziness and Complexity. As the complexity of a task (problem), or of a system for performing that task, exceeds a certain threshold, the system must necessarily become fuzzy in nature. Zadeh, originally an engineer and systems scientist, was concerned with the rapid decline in information afforded by traditional mathematical models as the complexity of the target system increased.

5.1 The FUZZY Controller Design and Preview Results

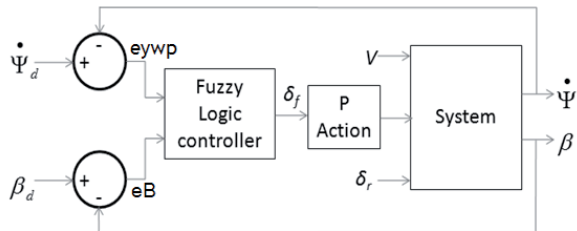


Fig. 4. Control system architecture.

Fuzzy logic is a new and innovative technology being used to enhance control engineering solutions. It allows complex system design directly from engineering experience and experimental results,

thus quickly rendering efficient solutions. Fig. 4 shows a blocks diagram of a general control system where the controller is a fuzzy logic block. For our case, the plant corresponds to the state space expression (eq. 7) and we know that the inputs to the system are the front and rear steering angles $u = [\delta_f, \delta_r]^T$ and if we want to achieve the path tracking of a straight line, for this goal is desirable to maintain an orientation of zero and likewise for the steering angles. In this work we are assuming that the system has the necessary sensors to provide the immediate information about the orientation, steering and side-slip. Thus, if we have some desired behavior, as we can see in fig. 4 the fuzzy controller must provide the necessary signal to the plant for achieve the desired behavior and error zero.

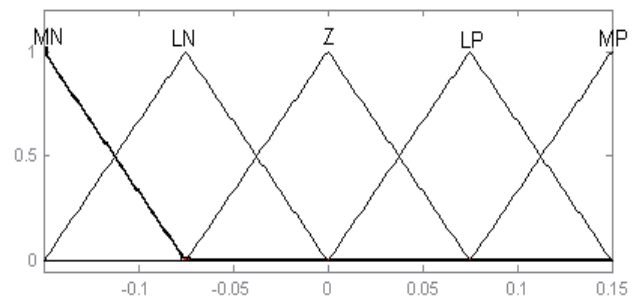


Fig. 5. Membership functions for the slide slip angle error (eB).

For the problem in this paper the synthesis of the fuzzy controller is based on the fuzzy logic theory and implemented for simulation with the fuzzy Logic toolbox of Matlab™ [10]. The scheme shown in the Fig. 4 represents the control architecture used in this work [11]. For the deployment control system the inputs are desired orientation rate ($\dot{\Psi}_d$) and the desired slip angle (β). It is possible to see that eB is representing the sideslip angle error and it is the difference between desired sideslip angle and the real measured sideslip angle. In this order, $eywp$ represents the orientation rate error. The block named system represents the dynamic model expressed with the equation shown in eq. 7 and for this model we have the velocity (V), the front steering angle and the rear steering angle as inputs of the vehicle model. The fuzzy controller receives the information respective to the error in the orientation rate and the sideslip angle error and acts providing the front steering angle action to a proportional controller. The proportional controller output is applied to the front

steering angle input of the dynamic model. Is very important to note that was assumed that rear wheel was considered fixed with zero steering angle.

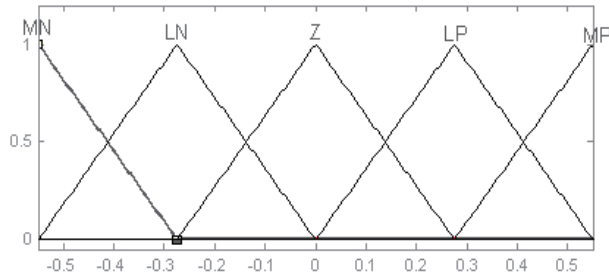


Fig. 6. Membership functions for the yaw angle rate error.

The development of the fuzzy controller requires the definition of the membership functions for the inputs and outputs of our system, thus, for this work the inputs are the error of orientation angle and the error of the slip angle. Fig. 5 shows a basic representation of the fuzzy logic controller with membership functions for “eB”.

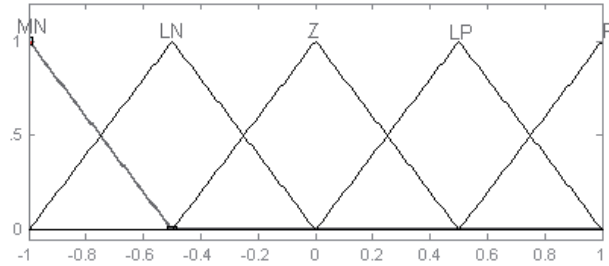


Fig. 7. Membership functions for the front steering angle command.

In this paper was desired use the vehicle parameters provided in [8] without forget the importance of the AgriBOT parameters in future works. These parameters are understood by the following values [8]: $m = 1573$; $I_{zz} = 2873$; $L_f = 1.1$; $L_r = 1.58$; $K_f = 80000$; $K_r = 80000$ and was assumed a constant velocity of 10m/s.

1. If (eB is Z) and (e_yawp is MP) then (deltaF is P) (1)
2. If (eB is Z) and (e_yawp is LP) then (deltaF is P) (1)
3. If (eB is Z) and (e_yawp is Z) then (deltaF is Z) (1)
4. If (eB is LP) and (e_yawp is MN) then (deltaF is MN) (1)
5. If (eB is Z) and (e_yawp is MN) then (deltaF is MN) (1)
6. If (eB is LN) and (e_yawp is LN) then (deltaF is LN) (1)
7. If (eB is Z) and (e_yawp is LN) then (deltaF is LN) (1)
8. If (eB is Z) and (e_yawp is MN) then (deltaF is MN) (1)
9. If (eB is LP) and (e_yawp is MN) then (deltaF is MN) (1)
10. If (eB is LN) and (e_yawp is MN) then (deltaF is MN) (1)

Fig. 8. Rules defined to Fuzzy Logic Controller.

Fig. 6 shows the definition of the membership functions to yaw angle rate error (eywp). Fig. 7 shows membership functions definition for the front steering angle command.

The membership functions of slide slip angle error and the orientation rate error are between small values because the goal of control is maintain these small values provided for the side slip angle and yaw rate.

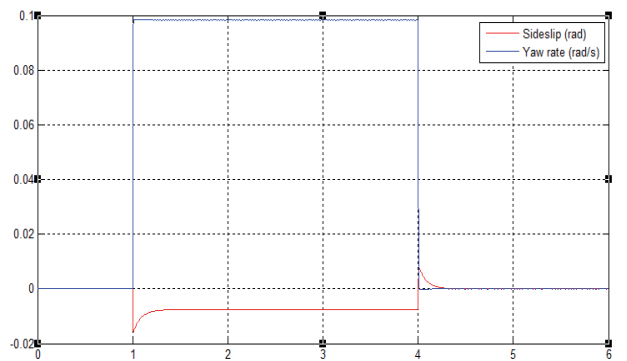


Fig. 9. Response of the controller to the desired inputs.

The membership functions was defined as MN (very negative), PN (little negative), Z (zero), PP (little positive), MP (very positive). In this simulation was defined also the rules for the fuzzy controller, in Fig 8 are shown this rules defined with the Matlab™ Fuzzy logic toolbox. The inference components are the eB (slide slip angle error), eywp (orientation error) and with the Mamdani method is decided is generate a magnitude to act on the operating front steering angle. Mamdani Fuzzy inference method is the most

commonly used in applications, due to its simple structure of 'min-max' operations [10].

6. EVALUATION OF THE RESULTS

To illustrate the performance of the designed controller, the response of the controlled vehicle to a given control task has been simulated. The simulations have been carried out using the state expression of eq. 7 and the fuzzy controller developed in the previous section with a constant speed of 10 m/s. The control task considered consists of tracking a yaw rate reference signal in the shape of a single rectangular pulse of 0.1 rad/s amplitude and 3s width while maintaining zero sideslip. The respective results are shown in Fig. 9, the red trace corresponds with the orientation rate (yaw rate) and the blue trace corresponds to the slip angle. As can be seen for the desired inputs the system has an acceptable response due to the work performed by the fuzzy controller.

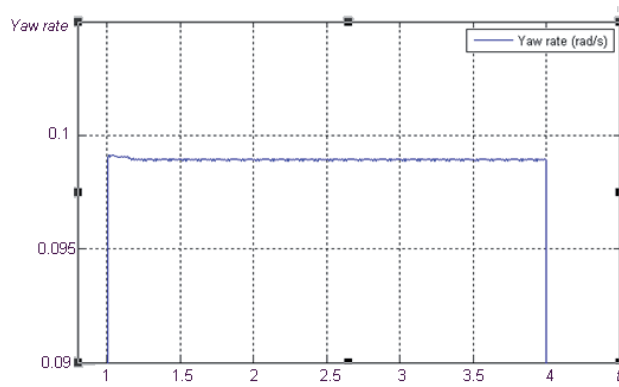


Fig. 10. Expanded form of the behavior of the yaw rate when is desired track a yaw rate of 0.1 rad/s.

In the Fig. 10 is show the expanded form of the behavior of the yaw rate basically when is desired track a yaw rate of 0.1 rad/s, this figure is included for evaluate the acceptable performance of the fuzzy controller about the required orientation variability during the respective time.

The Fig. 11 show the expanded form of the behavior of the slip angle basically when is desired track a yaw rate of 0.1 rad/s, also, this figure is included for evaluate the acceptable performance of the fuzzy controller about the required slip angle during the respective time.

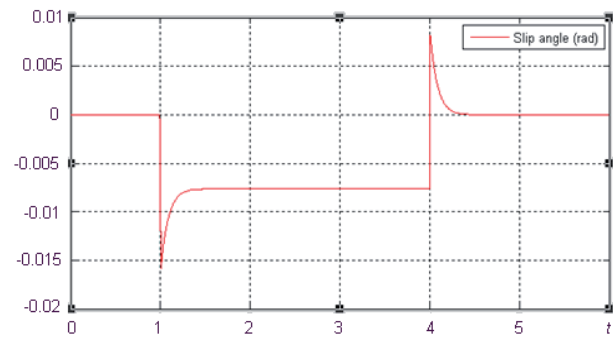


Fig. 11. Expanded form of the behavior of the slip angle when is desired track a yaw rate of 0.1 rad/s.

The Fig. 12 shows the performance of the system under the change of the desired yaw rate. In this case during 1s the desired yaw rate is zero; on the second 2 the desired yaw rate is 0.1rad/s during 3s, after during 1s the required yaw rate is zero, subsequently, the yaw rate is -0.1rad/s during 4s. Again is possible to evaluate the acceptable performance of the controller.

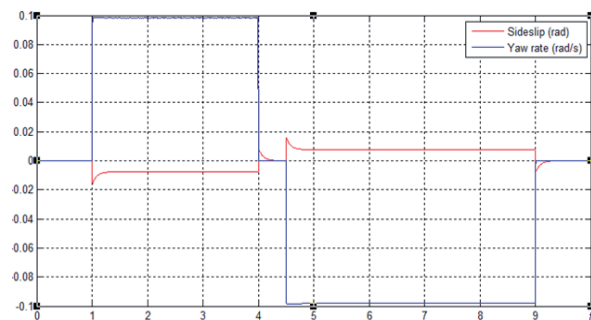


Fig. 12. Response of the controller to the desired inputs

Is necessary to mention that the response is acceptable, but the has a small steady-state error and in other works as [12], the results achieved are better and is proved that the appropriate definition of the membership functions help to improve the performance of a fuzzy controller, for this appropriate definition is possible use tools derived of the artificial intelligence.

7. CONCLUSION AND FUTURE WORK

In this work we presented the results of the design of an initial fuzzy controller applied on the stabilization problem of the yaw rate and sideslip angle in a 4WSD

Autonomous Vehicle. Based on tests described in the literature and in previous works was made a preliminary study that is a good basis that must be improved for its direct deployment on the AgriBOT mobile robot. It's sure that this previews controller needs more evaluations to decide to use it in the real deployment.

When it comes to future works, is a research group interest the deployment of the controller in an embedded processor. We want to evaluate the performance for controlling the AgriBOT mobile robot. As the AgriBOT prototype has mobility similar to holonomic mobile robot, and it was designed to work in the outdoor environments like farm fields, the vehicle model parameters can vary. Due to this, we consider essential the implementation of intelligent control systems that can deal with this.

The controller proposed integrates the slip angle and the yaw rate that is relatively easy using fuzzy logic, the dynamic model used in this work use parameter obtained in vehicle dynamics bibliography. Is desired assume dynamic models for off road vehicles.

AGRADECIMIENTOS

The authors would like to thank FINEP, CNPq, CAPES, COLCIENCIAS (PhD Scholarship), EESC-USP, EMBRAPA Technology, and Jacto S.A for their support in this work.

REFERENCES

- [1] Heraud, J., Lange, A., 2009, "Agricultural Automatic Vehicle Guidance from Horses to GPS: How We Got Here, and Where We Are Going", ASABE Distinguished Lecture #33, pp. 1-67, Agricultural Equipment Technology Conference February 8-11, Louisville, Kentucky, USA.
- [2] Sigrimis, N., Antsaklis, P., Groumpos, P.P., 2010, "Advances in control of agriculture and the environment," Control Systems, IEEE , vol.21, no.5, pp.8-12, Oct 2001. Singapore.
- [3] UNIBOTS., http://www.unibots.com/Agricultural_Robotics_Portal.htm, accessed in Jan. 2011.
- [4] Bak, T., Jakobsen, H., 2009 "Agricultural Robotic Platform with Four Wheel Steering for Weed Detection", Biosystems Engineering, Volume 87, Issue 2, February 2004, Pages 125-136.
- [5] Bakker, T. "An autonomous robot for weed control", design, navigation and control", PhD thesis - Wageningen University.
- [6] Tangerino, G., Godoy, E., Tabile, R., Souza, R., Inamasu, R., 2011, "Controle de Esterçamento de Robô Agrícola Móvel de Quatro Rodas Guiáveis", X Simpósio Brasileiro de Automação Inteligente (SBAI).
- [7] Vilaplana, M., Leith, D., Leithead, W., Kalkkuhl, J., 2003, "Control of Sideslip and Yaw Rate in 4-Wheel Steering Cars Using Partial Decoupling and Individual Channel Design", Proceedings of the European Control Conference 2003, 1-4 September 2003, Cambridge, UK. The IEE..
- [8] Rajamani, R., 2005 "Vehicle Dynamics and Control", Springer-Verlag, 2005.
- [9] Manning, J., Crolla, D., 2007, "A review of Yaw Rate and Sideslip Controllers for Passenger Vehicles", Transactions of the institute of measurement and control, pp. 117-135.
- [10] Sivanandam, S., Sumathi., Deepa, S., 2007, "Introduction to Fuzzy Logic using MATLAB", Springer-Verlag Berlin Heidelberg.
- [11] Hakima, A., Ameli, S., 2010, " Designing a Fuzzy Logic Controller to Adjust the Angle of Tires in Four Wheel Steering Vehicles", 2010 11th Int. Conf. Control, Automation, Robotics and Vision, Singapore.
- [12] Zhang, J., Zhang Y., Chen L., Yang J., 2007 , "A fuzzy control strategy and optimization for four wheel steering system," Vehicular Electronics and Safety, 2007. ICVES. IEEE International Conference on , vol., no., pp.1-6, URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4456359&isnumber=4456353>.

ENTERPRISE MASHUPS: A NEW APPROACH FOR BUSINESS SOLUTIONS

EMPRESA MASHUP: UN NUEVO ENFOQUE PARA SOLUCIONES DE NEGOCIO

Mg. Mario Paredes-Valverde
Instituto Tecnológico de Orizaba, México
marioparedes@acm.org

Ph.D. Gine Alor-Hernández
Instituto Tecnológico de Orizaba, México
galor@itorizaba.edu.mx

(Article of Scientific and technological research. Received el 17/10/2011. Accepted el 19/12/2011)

Abstract. A mashup is a Web application that integrates content from different providers in order to create a new service which is not offered by the content provider. The development of this kind of applications involves activities such as accessing heterogeneous sources, combining data from different data sources and building graphical interfaces. This factor limits non-experienced computer users to develop these applications. However, nowadays there are enterprise-oriented tools that allow non-experienced user for building mashups in order to respond business needs in an easy and rapid way. Due to this, the enterprise mashup approach has been widely adopted by a large number of enterprises. This paper presents an overview of the enterprise mashup approach, as well as a review of four enterprise-oriented tools which provide a set of features that allows non-expertise users developing mashups into an enterprise. Finally, we present the challenges to be addressed by enterprise-oriented mashup tools in order to provide an easier and faster way of developing mashups.

Keywords: Enterprise Mashups, Web applications.

Resumen. Un mashup es una aplicación web que integra el contenido desde diferentes proveedores con el fin de crear un nuevo servicio que no es ofrecido por el proveedor de contenido. El desarrollo de estas aplicaciones incluye actividades tales como el acceso a fuentes heterogéneas, combinación de datos procedentes de diversas fuentes y la creación de interfaces gráficas. Este factor limita a los usuarios no experimentados en el desarrollado de aplicaciones con en el uso de computadores. Sin embargo, hoy en día hay empresas orientadas a crear herramientas que permiten a los usuarios sin experiencia la creación de mashups para responder a las necesidades de negocio de una forma fácil y rápida. Debido a esto, el enfoque de mashup empresarial ha sido ampliamente adoptado por un gran número de empresas. Este artículo presenta una visión general del enfoque de mashup empresarial, así como una revisión de cuatro empresas orientadas a la producción de herramientas que ofrecen un conjunto de características que permite a los usuarios sin experiencia el desarrollo de mashups en una empresa. Por último, se presentan los desafíos que han de tenerse en cuenta por las empresas orientadas a la creación de herramientas de mashup empresarial con el fin de proporcionar una manera más fácil y más rápida de desarrollo de mashups.

Palabras clave: Empresas massups; Aplicaciones Web.

1. INTRODUCTION

In recent years, the adoption of Web technologies has been incremented, one of the goals of these technologies is to make easier tasks such as create, use, describe, share, and reuse resources on the Web. Due to this, a new approach for Web development called Mashup has emerged. Magazinius et al. [1] define a Web mashup as a Web application that integrates content from different providers to create a new service which is not offered by the content providers. Mashups are gaining popularity and they are applied in a large number of domains, they can be used from normal internet users to professionals or applications developers. This has spurred interest of using these technologies to empower enterprise employees to build applications that provides support to business processes. Mashups approach opens new and broader opportunities for data and service consumers, however, mashup development requires knowledge on three main tasks which are: 1) accessing to heterogeneous data sources such as Web Services, databases, RSS/Atom feeds, HTML pages, to mention but a few; 2) combining data from different sources and 3) building graphical interfaces. Nowadays, there are many research works and industrial tools which aim facilitating the mashup development, in particular to users which do not have programming knowledge allowing them respond to a personal or business needs.

This paper presents an overview of the enterprise mashup approach, as well as an overview of four enterprise-oriented mashup tools. This work is structured as follows. Section 2 presents related works of enterprise-oriented mashup development. Section 3 provides an enterprise-oriented mashup classification. Section 4 presents an overview of enterprise-oriented mashup tools. In section 5, the challenges on enterprise-oriented mashups tools are presented. Finally, we present our conclusions and emphasize our contribution.

2. RELATED WORKS

Nowadays, there are research efforts which aim to help enterprise employees develop applications that integrate data from heterogeneous data sources in order to solve business needs. Some of these efforts are focused for providing a mashup architecture. Chudnovskyy et al. [2] present an approach to support business processes integration and execution using services that provide communication and collaboration support and Web mashup techniques; also, they define a mashup reference architecture and derive a dedicated mashup execution platform. Soriano et al. [3] present EzWeb, a reference

architecture and implementation of an Enterprise 2.0 Collaboration Platform that allows users co-produce and share instant applications. Maraiakar et al. [4] present a service composition platform called SABRE. This platform provides tools to combine, filter, and transform Web services and data sources in order to build situational applications for business.

Other efforts addressed to develop enterprise-oriented mashup tools have been reported in the literature. López et al. [5] present the design of an enterprise-oriented mashup tool that provides components for accessing heterogeneous sources, a component to combine data from different sources and component for building the graphical interface. Altinel et al. [6] present a lightweight enterprise data integration service called Damia, this service allows users creating and cataloging high value data feeds for consumption by situational applications into an enterprise. Huang et al. [7] propose a system called SituMash which includes context representation, situation reasoning, workflow planning and mashup optimization. This system provides support for the automatic composition of widgets in response to situation changes without user manual selection. Abiteboul et al. [8] present MatchUp, a system that supports rapid, on-demand, and intuitive development of mashups, based on an autocompleting mechanism. MatchUp predicts, given a user's partial mashup specification, what are the most likely potential completions for the specification.

Hoyer et al. [9] propose a reference model for Enterprise Mashups that provides a foundation to develop and analyze grassroots Enterprise Mashups environments from a managerial and collaborative perspective. Xu et al. [10] propose a lightweight process modeling approach for process enterprise mashup applications; this approach pretends support collaboration among users with different levels of modeling skills and expertise in a virtual enterprise environment.

In table 1, a comparison of the aforementioned works is presented; these works represent a great effort in order to involve all enterprise users on the development of applications. However, these works have several drawbacks, such as: a) a vast amount of technologies and Web data sources are not included, b) lack of usage for advanced capabilities of rich Internet application technologies, and c) Web development experience required. These deficiencies can be solved by extending the capabilities of integration with other technologies and data sources, and using graphical and friendly interfaces, which are necessary in order to improve the non-expert user's experience. In the following section, an enterprise-oriented mashup classification is presented.

TABLE 1. An overview of enterprise mashup development

Proposal	Approach	Drawbacks
Chudnovsky et al [2]	Business processes integration and execution	The architecture is focused in using an specific telecommunications service (Telco)
Soriano et al. [3]	Knowledge workers as co-producers of software services and applications	Few data sources are considered and there is little support for building graphical interfaces
Maraikar et al [4]	Combine, filter and transform Web services and RSS and Atom feeds	Reo language knowledge is required, few data sources are supported
López et al. [5]	Enterprise mashup tool to build graphically mashup applications	It has support limited for building graphical and friendly interfaces
Altinel et al. [6]	Data integration service for business users	It has not support for building graphical interfaces and few data sources are considered
Huang et al. [7]	Automatic composition of mashups in response to situation changes	Few data sources are considered.
Abiteboul et al. [8]	System for mashup development based on an autocompletion mechanism	Friendly and intuitive graphical interfaces are not provided.
Hoyer et al. [9]	A reference model for Enterprise mashups from a managerial and collaborative perspective	Narrow application of the reference model in other areas and its verification in an iterative design cycle
Xu et al. [10]	Process modeling for process enterprise mashup applications	Process modeling skills and BPMN (Business Process Modeling Notation) Knowledge required

3. ENTERPRISE MASHUPS

Mashup is a new approach which has emerged in recent years and its use in enterprise environments increases every day. Vrieze et al. [11] identify four kinds of Enterprise mashups which are described next.

Business process mashups. These mashups coordinate different orchestration events, allowing a user to automate tasks that involves getting information from different data sources, aggregating, filtering, and shaping it, and then sending the result to several destinations. These kinds of mashups allow users specifying this execution himself, which will be executed as if the tasks were manually performed by the user.

Front-end mashups. These mashups aim monitoring business process in order to help technical leads in lines of business. They perform business intelligence tasks by retrieving information from different data sources, processing this, and then visualizing the results. An example of this kind of mashup is the overlay sales data on a map.

User interface mashups. Many business process tasks require the entry, editing, and approval data; this kind of mashups help the aforementioned tasks, due they function as user interfaces for data entry.

Knowledge mashups. These mashups are used on management of knowledge within an enterprise. Knowledge mashups allow users collaborating among

them. They provide social features allowing that members of different departments know what others departments are doing, the data synchronization among them, to mention a few tasks.

4. ENTERPRISE-ORIENTED MASHUP TOOLS

The enterprise-oriented tools have been primarily focused on creating an architecture that supports common processes and store data. According to Hoyer and Stanoevska-Slabeva [12], from an architectural perspective the enterprise mashup stack consists of three layers, which are described below.

Resources. It is the lowest layer and contains data or application functionality which represents the core building blocks of Enterprise Mashups. These sources are primarily accessed through APIs.

Widgets. Widgets provide applications on domain functions or data specific functions. They provide graphics, simple and efficient user interaction mechanism. They have been focused on consumption and customization in order to ensure flexibility and reusability. However, software vendors such as Microsoft®, IBM®, or Google® define their own widget model.

Mashup. The users are able to define behavior according to their needs through assembling and composing a set of widgets. Also, by aggregation and linking of widgets in a visual and intuitive way, users can solve heterogeneous business problems.

In addition to the architecture layers aforementioned, the user needs of collaborative mechanisms, since collaboration is becoming in an essential element for improving productivity and accelerating innovation at the enterprise. The emergence of Web 2.0 has opened up new ways of collaboration. An environment in which workers are considered as co-producers of software services and applications through which take competitive advantages and meet business needs without involving IT departments.

Enterprise collaboration has obtained benefits from the emergence of a new approach called Enterprise 2.0 [13]. This approach provides to enterprises with models and tools for collaboration and co-creation. Some examples of enterprise collaboration tools are social linking and tagging tools such as social networking, social bookmarking and social search. These tools offer support for comments and voting, content management platforms such as wikis and forums, and information distribution tools such as RSS/Atom feeds. Nowadays many enterprises have deployed collaboration

platforms, which provide collaborative services for knowledge workers. Besides, another fundamental element that have been provided to the enterprise employees is the chance of remixing and integrating already available services through enterprise mashups and collaborative resource repositories. A set of enterprise-oriented mashup tool that provides some of the features aforementioned are presented below.

Apatai®. It is a Java-based open source data integration software package which provides support to work with data of several data sources and formats [14]. Apatai® have a robust, scalable architecture which allows carrying out tasks such as extraction, transformation and data load for warehousing, as well as the migration, synchronization, and application integration. Apatai provides solutions form Application Integration, Salesforce and QuickBooks Synchronization, Mashup Data Integration, Data Warehousing and Analytics, Operational Integration, and Embedded Data Integration. This tool provides Enterprise 2.0-class middleware which allows developers accessing data both on a local network, an on-demand applications and Cloud systems such as Salesforce.com and SugarCRM. On Apatai, the mappings and transformation rules are graphically defined through a drag-and-drop interface, this feature allows non-developers developing data mashups. Apatai provides support for operating systems such as Windows, Linux RedHat, SUSE Linux, and MacOS, besides, it provides three different deployment options, which are: 1) desktop application, 2) server engine on JVM (Java Virtual Machine) and 3) embedded into third parties software.

IBM Mashup Center. It is an enterprise-oriented mashup platform that allows creating, sharing, and discovering reusable application building blocks such as widgets, feeds, services or mashup applications, these blocks can be assembled into new applications or leveraged within existing applications [15]. IBM Mashup Center provides the next main components: (a) Mashup Builder is a widget-based browser interface that allows creating, assembling, configuring and designing widgets, gadgets, mashup pages, and spaces, besides it allows searching widgets and add them into mashup pages; (b) Widget builder is a browser-based tool that allows both business users and Web developers creating widgets and sharing with the community; (c) Catalog is a repository that allows sharing, reusing and discovering widgets, it includes features such as ratings, tagging, and commenting; (d) Feed generator is a wizard-based interface that allows creating consumable feeds from enterprise data sources; and (e) Data mashup builder is a browser-based tool which allows mixing, merging, grouping, filtering, sorting and transforming feeds, with the aim of provide a single view of disparate sets of

information. IBM Mashup Center provides support for operating systems such as Microsoft Windows, AIX (Advanced Interactive Executive) and Linux.

JackBe Presto. It is an enterprise-oriented mashup platform that allows developing enterprise applications which integrate data from live information data sources [16]. Also, this tool enables users to visually assemble applications for business. The applications developed with Presto can be used in portals, Sharepoint, desktop, spreadsheets, Web sites and mobile devices. JackBe Presto provides a set of tools that allows carrying out the aforementioned tasks, these tools are: (a) Presto Wires allows developing data mashups, its interface is completely drag-and-drop and it provides virtual representation of data sources regardless of their type (RSS, REST, SOAP, XML, HTML); (b) Mashup Maker is a Web-based editor that allows defining applications using EMMML (Enterprise Mashup Markup Language); (c) Presto Mashup Studio is a plugin for Eclipse IDE which allows designing, testing, debugging and deploying mashups, besides, this tool allows creating mashups from scratch, enhancing mashups created by Presto Wires, and reusing applications developed by other users; (d) Presto App Maker allows analysts creating visually-rich applications from mashups without developer assistance; (e) Presto Mashboard allows business users select apps and drop them onto a WYSIWYG dashboard canvas with the aim of covering requirements related to report and data-viewing; and (f)

Presto App Depot is a self-service in-house store that allows finding, using and sharing applications across an organization.

WaveMaker. It is a development platform for Web and Cloud applications, it provides visual and drag-and-drop tools which allows building Java-based mashups [17]. WaveMaker Studio allows building applications which can be extended by expert developers using a Java IDE. WaveMaker provides security, data access and scalability through Java-based components. Furthermore, it provides support for Web application servers such as Tomcat, JBoss, GlassFish, WebSphere and WebLogic, and for databases such as MySQL, PostgreSQL, HSQLDB, Microsoft SQL Server and DB2.

The data sources supported for an enterprise-oriented mashup is a fundamental factor to consider when a tool is elected for its implementation into an enterprise. In Table 2, a comparison of the aforementioned tools is provided; this comparison is carried out through the technology and data sources supported.

The enterprise-oriented mashup tools reviewed provide a great support for a great number of data sources; also they allow integrating other technologies. Furthermore, these tools provide a great support for building the graphical interface, all this turning them in powerful tools that allows even non-expert users developing mashups that point to business problems or needs.

TABLE 2. Enterprise-oriented mashup tools comparison

Mashup tool	Technology	Data sources supported
Apatar	Java	Salesforce.com, Centric CRM, Databases (MySQL, Dbase, Oracle, PostgreSQL), Structured/Unstructured files (XML, CSV, TXT, MS Excel, EditGrid), Web destinations (Flickr, Amazon S3)
IBM Mashup Center	IBM WebSphere Application Server, Derby	Sharepoint, MW, Filenet, spreadsheets, Atom, RSS, JSON, XML, Databases, Google gadgets
JackBe Presto	Java, HSQL database, Apache's Tomcat Servlet Container, EMMML	Sharepoint, portal platforms (IBM, JBoss, LifeRay), Oracle 9i/10g/11g, Oracle Fusion SOAs, Oracle Applications, Excel
WaveMaker	Java, JavaScript	REST Web Services, SOAP Web Services, MySQL, PostgreSQL, HSQLDB, Oracle, Microsoft SQL Server, IBM DB2

5. ENTERPRISE MASHUPS TOOLS CHALLENGES

The mashup developed with any enterprise-oriented tool can contain valuable enterprise data, so these tools have provided good security mechanisms such as authentication and authorization, besides, the applications developed are shared between employees across an organization as well as external customers, partners, and suppliers, so the security must be provided regardless of where the applications is being accessed.

The popularity of mobile devices grows daily; due to this an important challenge of enterprise-oriented mashup tools is the use of lightweight Web services that allows the mobile collaboration, as well as the development of intuitive and easy mobile applications that do not represent an obstacle for non-expert users.

The support to build graphical interfaces provided by the enterprise-oriented mashup tools is still limited; so it is important include RIAs (Rich Internet Applications) features in order to provide a better user experience. Also, sometimes the applications developed require the input of data from users, so, is necessary provides validation mechanism for the input data, taking into account that such applications will be developed by non-expert users.

Other important factor to be considered by these tools is the opportunity of providing recommendation mechanism that allows users get potential mashups allowing reusing these components in order to respond to a business need in a faster way.

It is important to take into account the aforementioned challenges, because this allows enterprise get an environment where all its employees collaborate with the solution to business problem or needs, and thereby improve the enterprise productivity.

6. CONCLUSIONS

Enterprises take advantages of mashup approach because it allows employees respond to a business need in a collaborative, easier and faster way. Day-to-day new development tools, languages, methodologies and initiatives emerge with the aim of facilitating the enterprise mashup development. This works introduces the mashup approach and provides a perspective of some of the most mature enterprise-

oriented mashup tools. These tools widely vary in terms of services and features provided, and the knowledge about them helps developers and IT consultants to select a tool for its implementation into an organization in order to improve the enterprise productivity. Despite the advantages provided by these tools, they represent a great financial investment; which limits small organization to acquire such tools. Also, it is very important allow both expert and non-expert users collaborating between them with the aim of getting a feedback about the strengths and weaknesses of these tools with the aim of improving these tools allowing respond to the constant changes that can occur in the enterprise area.

ACKNOWLEDGEMENTS

This work is supported by the General Council of Superior Technological Education of Mexico (DGEST). Additionally, this work is sponsored by the National Council of Science and Technology (CONACYT) and the Public Education Secretary (SEP) through PROMEP.

REFERENCES

- [1] J. Magazinius, A. Askarov, A. Sabelfeld. "A Lattice-based Approach to Mashup Security". Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, pp. 15-23, 2010.
- [2] O. Chudnovskyy, H. Gebhardt, F. Weinhold, M. Gaedke. "Business Process Integration using Telco Mashups". The 8th International Conference on Mobile Web Information Systems, Vol. 3, pp. 677-680, 2011.
- [3] J. Soriano, D. Lizcano, M. Cañas, M. Reyes, J. Hierro. "Fostering Innovation in a Mashup-oriented Enterprise 2.0 Collaboration Environment". SIWN International Conference on Adaptive Business Systems, 2007.
- [4] Z. Maraikar, A. Lazovik, F. Arbab. "Building Mashups for the Enterprise with SABRE". Proceedings of the 6th International Conference on Service-Oriented Computing, pp. 70-83, 2008.
- [5] J. López, F. Bellas, A. Pan, P. Montoto. "A Component-Based Approach for Engineering Enterprise Mashups", Lecture Notes in Computer Science, Vol. 5648, pp. 30-44, 2009.
- [6] M. Altinel, P. Brown, S. Cline, R. Kartha, E. Louie, V. Markl, L. Mau, Y. Ng, D. Simmen, A. Singh. "Damia: A Data Mashup Fabric for

- Intranet Applications". Proceedings of the 33rd international conference on Very large databases, pp. 1370-1373, 2007.
- [7] A. Huang, S. Huang, E. Lee, S. Yang, "Improving End-User Programming with Situational Mashups in Web 2.0 Environment". Proceeding of the 2008 IEEE international Symposium on Service-Oriented System Engineering, pp. 62-67, 2008.
- [8] S. Abiteboul, O. Greenshpan, T. Milo, N. Polyzotis, "MatchUp: Autocompletion for Mashups". Proceedings of the VLDB Endowment, Vol. 2 pp. 538-549, August 2009.
- [9] V. Hoyer, K. Stanoevska-Slabeva. "Towards a Reference Model or Grassroots Enterprise Mashup Environments". Proceedings of the 17th European Conference on Information Systems, 2009.
- [10] L- Xu, P. Vrieze, K. Phalp, S. Jeary, P. Liang. "Lightweight Process Modeling for Virtual Enterprise Process Collaboration". 11th IFIP Working Conference on Virtual Enterprises, pp. 501-508, October, 2010.
- [11] P. Vrieze, L. Xu, A. Bouguettaya, J. Yang, & J. Chen. "Process-oriented Enterprise Mashups". Proceedings of the 2009 Workshops at the Grid and Pervasive Computing Conference, pp. 64-71, 2009.
- [12] V. Hoyer, K. Stanoevska-Slabeva, "Design Principles of Enterprise Mashups". Proceedings of the 5th Conference of Professional Knowledge Management, 2009.
- [13] A. MacAfee. "Enterprise 2.0 The Dawn of Emergent Collaboration", MIT Sloan Management Review, April 2006.
- [14] Apatar, Retrieved from: <http://apatar.com/>, 2011.
- [15] IBM Mashup Center, Retrieved from <http://www-01.ibm.com/software/info/mashup-center/>, 2011.
- [16] JackBe, Retrieved from: <http://www.jackbe.com/>, 2011.
- [17] WaveMaker, Retrieved from: <http://www.wavemaker.com/>, 2011.

CLOUD COMPUTING: A REVIEW OF PAAS, IAAS, SAAS SERVICES AND PROVIDERS

CLOUD COMPUTING: UNA REVISIÓN DE LOS SERVICIOS Y PROVEEDORES PAAS, IAAS, SAAS

Mg. María Salas-Zárate

Instituto Tecnológico de Orizaba, México
pilersalas@acm.org

Mg. Luis Colombo-Mendoza

Instituto Tecnológico de Orizaba, México
lcolombo@acm.org

(**Review** Article. Received el 17/10/2011. Accepted el 19/12/2011)

Abstract. Cloud computing has become an important factor for businesses, developers, workers, because it provides tools and Web applications that allows storing information on external servers. Also, Cloud computing offers advantages such as: cost reduction, information access from anywhere, to mention but a few. Nowadays, there are several Cloud computing providers such as: Google Apps, Zoho, AppEngine, Amazon E2C, among others. These providers offer Software, Infrastructure or Platform as a Service. Taking this into account, this paper presents a general review of Cloud computing providers in order to allow users, enterprises, and developers select the one that meets their needs.

Keywords: Cloud computing, IaaS, PaaS, SaaS.

Resumen. La computación en la nube se ha convertido en uno de los factores relevantes para las empresas, desarrolladores y trabajadores, porque proporciona herramientas y aplicaciones web que permite almacenar información en servidores externos. Además, la computación en la nube ofrece ventajas tales como: reducción de costos, acceso a la información desde cualquier lugar, por mencionar sólo algunos. Hoy en día hay varios proveedores de computación en la nube como: Google Apps, Zoho, AppEngine, E2C Amazon, entre otros. Estos proveedores ofrecen software, infraestructura o plataforma como un servicio. En este trabajo se presenta una revisión general de los proveedores de computación en la nube, con el fin de permitir a los usuarios, empresas y desarrolladores seleccionar el que se ajuste a sus necesidades..

Palabras clave: Computación en la nube; IaaS, PaaS, SaaS.

1. INTRODUCTION

Cloud computing is a new style of computing in which dynamically scalable and often virtualized resources are provided as a services over the Internet [1]. According to who owns and manages the Cloud services, these can be classified in: Public, Private and Hybrid Clouds. Public Cloud offers services to general public, unlike Private Cloud that are used by a single organization. Also, there are Hybrid Clouds which offer a combination of private and public Clouds. Furthermore, there are three Cloud services models: 1) SaaS (Software as a Service), 2) PaaS (Platform as a Service) and 3) IaaS (Infrastructure as a Service). IaaS offers an infrastructure of resources (storage, databases, among others), usually in terms of Virtual machine as a service. PaaS offers an environment oriented to the development, testing, deployment and hosting of applications. SaaS offers a set of applications installed and running over Internet. This paper presents a review of PaaS, SaaS and IaaS providers, with the aim of helping users to select the provider that best suits their needs and budget.

This work is structured as follows. In section 2 related works of Cloud computing are discussed. Section 3 presents the architecture and features of Cloud computing. Section 4 presents the public, private and hybrid Cloud services. In section 5 a comparison of IaaS, SaaS and PaaS providers is presented. Finally, we present our conclusions and emphasize our contribution.

2. RELATED WORK

In [2] discuss both features and benefits of Cloud computing, and the Infrastructure as a Service (IaaS). Also, they provide a means of understanding and investigating IaaS. They outline the responsibilities of IaaS providers and the facilities to IaaS consumers. In [3] present the design of a trusted Cloud computing platform (TCCP) that enables IaaS services such as Amazon EC2 to provide a closed box execution environment. TCCP guarantees confidential execution of guest Virtual Machines, and allows users attesting the IaaS provider and determining if the service is secure before they launch their Virtual Machines In [4] describe a PaaS architecture for provisioning of real-time service-oriented application in Clouds. Also, they show how the combination of methods, tools and services can be used to

improve the usability, maintainability and efficiency of services targeting Clouds with strict QoS constraints. In [5] propose a taxonomy which identifies and classifies eight important elements that characterize Cloud computing infrastructures: 1) service type, 2) resource deployment, 3) hardware, 4) runtime tuning, 5) security, 6) business model, 7) middleware, and 8) performance. The taxonomy is based on a survey conducted on an important set of seven IaaS providers, six Web hosting companies, and two mixed PaaS and Web hosting providers. In [6] present a Cloud services comparison based on aspects such as architecture, features, and application. In [7] present a survey of Cloud computing, highlighting its key concepts, architectural principles, state-of-the-art, implementation as well as research challenges. Also, they provide a better understanding of the design challenges of Cloud computing and they identify important research directions in this area. In [8] examine the available Cloud computing services and propose a tree-structured taxonomy which offers a common terminology and baseline information for easy communication. In [9] analyze public quality information of Cloud services in terms of service types, also they show that IaaS quality information is more detailed and balanced than those of PaaS and SaaS. In [10] present a detailed comparison of Cloud services according to the Cloud types, interfaces, compatibility, implementation, deployment requirement and development support.

3. CLOUD COMPUTING

According to NIST(National Institute of standards and Technology) Cloud computing is defined as *a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction* [11].

The aforementioned definition has been adopted by a great number of authors. Next, a Cloud computing general architecture is presented.

3.1 Cloud Computing Architecture

A Cloud computing architecture consists of four layers: Hardware, Infrastructure, Platforms and Application. This general architecture is shown in Fig. 1.

The layers of this Cloud computing architecture are described below.

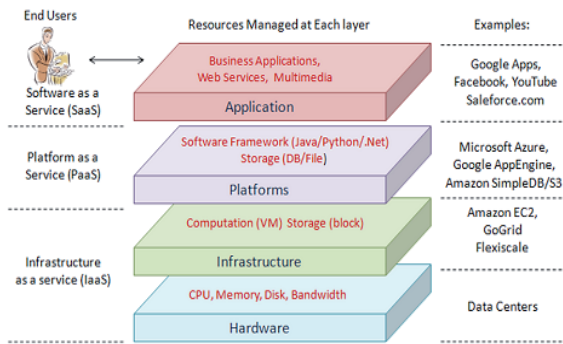


Fig. 1 Cloud computing architecture

Hardware layer: It manages the physical resources of the Cloud, including physical servers, routers, switches, power and cooling systems.

Infrastructure or Virtualization layer: It creates a pool of storing and computing resources by partitioning the physical resources using virtualization technologies such as VMware [12], Xen [13], and KVM [14].

Platform layer: It is composed of operating systems and application frameworks. Its aim is to minimize the burden of deploying applications directly in VM containers.

Application layer: It consists of the actual Cloud applications. In contrast to traditional applications, Cloud applications can leverage the automatic-scaling feature to achieve better performance, availability and lower operating cost [7].

Next, the main features of Cloud computing are described.

3.4 Cloud computing features

On-demand self-service: A consumer can be unilaterally provided of computing capabilities information such as server time and network storage, without requiring human interaction with each service provider.

Broad network access: Capabilities are available over the network and can be accessed through

standard mechanisms by heterogeneous thin or thick client platforms such as mobile phones, tablets, laptops, and workstations.

Resource pooling: The provider's computing resources are pooled in order to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to the consumer demand. There is a sense of location independence in that the customer has not control or knowledge over the exact location of the provided resources but he may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources are storage, processing, memory, and network bandwidth.

Rapid elasticity: Capabilities can be elastically provisioned and released, in some cases, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.

Measured service: Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service [11].

4. CLOUD TYPES

There are three kinds of Cloud computing services, which vary according to who owns and manages them. Next, these are described.

Private Clouds or internal Clouds: It exclusively offers Cloud services for business or organizations that prefer to keep their data in a more controlled and secure environment.

Public Clouds or external Clouds: It is the most popular type of Cloud computing that offers services that are available to the general public.

Hybrid Clouds: A hybrid Cloud is composed of several public Clouds, private Clouds and/or community Clouds. It offers more flexibility than both public and private Clouds [15]. In Hybrid clouds, part of the service infrastructure runs in private Clouds while the remaining part runs in public Clouds.

5. CLOUD SERVICES

Cloud computing services are classified in three service models: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a service (IaaS). Nowadays, there are many Cloud Computing providers of each one of the aforementioned models. Some of them are presented below.

5.1 SaaS

SaaS offers already created applications running on a Cloud infrastructure such as: Web-based email, alternatives of typical office applications as word processors, to mention but a few. This model eliminates the need to install and run the application on the customer's local computers therefore applications can be accessed through networks from various clients such as Web browsers, and mobile phones [16], [17].

Examples of SaaS providers are: Zoho [18], AtTask Inc. [19], NetSuite [20], Bloomfire Inc. [21], Smartsheet [22], Salesforce [23], and Google Apps [24], among others. Next, the aforementioned SaaS providers are described.

Zoho: It was founded by Sridhar Vembu in 1996. It offers a comprehensive suite of award-winning on-line business, productivity and collaboration applications. Customers use Zoho Applications to run their business processes, manage their information and be more productive while at the office or on the go, without having to worry about expensive or outdated hardware or software [18].

AtTask Inc: It was founded by Scott Johnson, Jason Fletcher, and Nate Bowler in 2001. AtTask is the SaaS leader in project management solutions and the only provider of Collaborative Work Management. It helps organizations empower their knowledge workers by leveraging the combined power of social collaboration, empowerment, accountability, visibility, and recognition, making it possible for people to take ownership of their work and perform at a higher level, creating an environment that fosters productivity, motivation, and results [19].

NetSuite: It was founded by Evan Goldberg in 1999. NetSuite was one of the earliest SaaS offerings on the market, and its main purpose is to provide integrated business management software to

midsize organizations. NetSuite provides a set of features such as customer relationship management (CRM), order fulfillment, inventory, accounting and finance, product assembly, Ecommerce, website management, and employee productivity [25].

Bloomfire Inc.: It was founded in 2010. It incorporates the best features of content management, file sharing, social business, and learning management software to create a collaboration tool perfect for the modern workforce. It allows capturing, displaying, saving, and sharing information, connecting people who have knowledge with those who need it, in order to improving team productivity [21].

Smartsheet: It was founded in 2005 by former executives of CRM company Onyx Software [26]. This is privately held Software as a Service (SaaS) known for its award winning online project management, collaboration, and file sharing features. It is broadly used to track and manage diverse types of work including: team projects and task lists, customer information, sales pipelines, event schedules, and business processes.

SalesForce: It was founded in 1999, and offers a suite of online programs for customer relationship management(CRM) and other business-oriented tasks [27].

Google Apps: It provides several Web applications with similar functionality to traditional office software (word processing, spreadsheets, among others), but also enables users to communicate, create and collaborate in an easy and efficient way. All applications are kept online and they are accessed through a Web browser, users can access their accounts from any internet-connected computer, and there is no need to install anything extra locally [1].

In Table 1 a comparison of providers aforementioned is presented. This comparison was carried out according to the software and price.

TABLE 1. SaaS providers

Provider	Software	Pricing
Zoho	<ul style="list-style-type: none"> - Collaboration applications: chat, commentBox, docs, discussions, mail, meeting, Projects, share, wiki. - Business applications: Assist, Marketplace, Books, People BugTracker, Recruit, Challenge, Reports, Creator, Site24X7, CRM, Sites, Invoice. - Productivity Applications: Calendar, Sheet, Notebook, Show, Planner, Writer. 	<ul style="list-style-type: none"> - Free plan available (3 apps, 2 users, 1000 records, 2 reports 200 MB storage) - Basic \$15/month (3 users, 6000 records, 5 reports, 500MB storage scheduler) - Professional \$45/month (10 users, unlimited apps, 20000 records, unlimited reports, 2GB storage, payment module, scheduler, backup) - Enterprise \$175/month (50 users, unlimited apps, 100000 records, unlimited reports, 7GB storage, payment module, scheduler, backup) - Free Trial available
AtTask, Inc.	Project Management, Project Reports & Dashboards, Resource Management Team Collaboration, Portfolio Management, Time Management. Help Desk & Issue Tracking, Product Integration, Gantt Chart, Process Improvement.	AtTask offers four license types: Full User, Team Member, View Only, and Requester. Each license type varies in terms of accessibility and functionality.
NetSuite	ERP/accounting, order management/inventory, CRM, professional services automation (PSA), and Ecommerce	\$499 base per month + \$99 per user per month. Starting from \$499.00/month Pricing model Subscription Free Trial Available
Bloomfire, Inc	Content management, file sharing, social business, and learning management software	<ul style="list-style-type: none"> - Bootstrap 10 users \$99.00/month - Startup 25 users \$199/month - small business 50 users \$299/month - Free Trial Available
Smartsheet	Project management and collaboration, file sharing.	<ul style="list-style-type: none"> - Professional a user + unlimited collaborators, 10 or 50 sheets, 3 or 15 GBfile storage \$15.95/month. - Team 3 to 200 users+ unlimited collaborators,50 sheets per user, 15 GB file storage per user \$49.95/month - Free Trial available
SalesForce	Customizable applications, mostly Customer Relationship Management (CRM) services.	<ul style="list-style-type: none"> - Contact manager \$5/user/month(contact management for up to 5 users) - Group \$15/user/month(Basic sales & marketing for up to 5 users) - Professional \$65/user/month(complete CRM for any size team) - Free trial available
Google Apps	Email, Calendar, Documents editing and creation, Mobile access, Uptime Guarantee	<ul style="list-style-type: none"> - Free plan available - Paid \$9/app/month - premier \$500/account/month

The use of the aforementioned providers depends of the user requirements. On the other hand, there are providers such as Zoho and Google Apps that are some of the most used today and they offer a free plan with a variety of available software.

5.2 PaaS

PaaS provides a development environment as a service where applications are developed using a set of programming languages and tools. These services may include development, integration, testing or resources storage to complete the life-cycle of services [28], [29]. Examples of PaaS providers are: AppEngine [30], Aneka [31], Force [32], Microsoft Azure [33], Heroku [34], Amazon Elastic MapReduce [35], CloudFoundry[36], among others. Next, the aforementioned PaaS providers are described.

AppEngine: It is a PaaS offering, which provides Python and Java support. AppEngine allows building scalable Web applications without the need for complex underlying hardware and software layers. Google abstracts those layers and lets fully concentrate on the application [37].

Aneka: It is a .NET-based application development Platform-as-a-Service (PaaS), which offers a runtime environment and a set of APIs that enable developers to build customized applications by using multiple programming models such as Task Programming, Thread Programming and MapReduce Programming. Also, Aneka provides a number of services that allow users controlling, auto-scaling, reserving, monitoring and billing the resources used by their applications. One of key characteristics of Aneka PaaS is to support provisioning of resources on public Clouds such as Windows Azure, Amazon EC2, and GoGrid [38].

Force: It is an enterprise Cloud computing platform offered by Salesforce. It allows service vendors developing and delivering stable, secure and scalable

applications. Two key technologies of Force.com are multi-tenancy and metadata. The idea of the Force.com Cloud solution is that it should take care of all common underlying requirements so that users need only focus on the design of their applications [1].

Microsoft Azure: The Azure Service platform is Microsoft's PaaS offering. Azure is based on the .Net language. It has a dedicated API to store and retrieve data called SQL Services. The underlying system for these SQL Services is Microsoft SQL Server. Although not all functionalities of Microsoft SQL Server are exposed via the API, the user can run transactions and use a restricted SQL query language [39].

Heroku: It is a very powerful Web-based platform which provides support for Ruby, JavaScript and Java Web applications. It directly deploys within a PaaS environment allowing quick migration from the development to deployment stage. It also contains a real time collaborative editor for use with up to five people [40].

Amazon Elastic MapReduce: It is an Amazon Web Service (AWS) that uses Hadoop to provide a MapReduce functionality. In the background itself relies mostly on two other AWS services: Elastic Compute Cloud (EC2) and Simple Storage Service (S3)[41].

CloudFoundry: CloudFoundry is an open-source platform-as-a-service environment offered by VMware that provides the environment to host multiple languages and frameworks in an open stack of software applications that can run on both outside and inside the firewall. The main features of CloudFoundry are: Choice of developer frameworks, Choice of application infrastructure services, and Choice of Clouds [42].

In Table 2 a comparison of the aforementioned providers is shown.

TABLE 2. PaaS providers

	<i>Programming language</i>	<i>Frameworks</i>	<i>Persistence options</i>	<i>Pricing</i>	<i>Use</i>
AppEngine	Python, Java (App Engine for Java supports many JVM languages, including BeanShell, Groovy, Scala, JRuby, Jython, and Rhino)	Restlet, Vaading, itsNat, Sweet, Struts 2, Spring MVC, Sinatra, Wicket, Lift, JSF 1.1, GWT, Tapestry, Grails, Slim3, Play Several frameworks based on Python e.g. Django, CherryPy, Pylons, web.py, and web2py.	Big Table	Maximum 500 MB storage and 10 apps free for developer	Web applications
Aneka	.NET	Spring .NET framework	RDBMS, SQL Express, MySQL and flat files.	-----	.Net enterprise applications, Web Applications
Force	Apex	----	Own object database	-----	Enterprise Applications
Microsoft Azure	NET, PHP, Java	----	Table/BLOB/queue storage, SQL Services	Maximum 10 web sites free	Enterprise applications, Web applications
Heroku	Ruby, Java, Python, Scala, Clojure, Node.js	Rails, Sinatra, ramaze, camping, Spring, Grails, Play, Tapestry, Twisted, Django	PostgreSQL, Amazon RDS MySQL, SQLite, PostgreSQL, MongoDB, CouchDB	Maximum 5 MB in disk for DB and 50MB for all files including GIT repositories	Web Applications
Amazon Elastic MapReduce	Hive and Pig, Cascading, Java, Ruby, Perl, Python, PHP, C++	mrjob	Amazon S3	Free 5 GB of Amazon S3 storage, 20,000 Get Requests, 2,000 Put Requests, and 15GB of data transfer out each month for one year	Data Processing
CloudFoundry	Java, Ruby, Scala, Python, PHP, Node.js	Spring, Rails, Sinatra, Grails Lift,	PostgreSQL, MySQL, MongoDB, Redis	service is available immediately at CloudFoundry.com in beta and is free to use. (Pricing will be announced after an extended beta period)	Web applications

AppEngine is one of the most used platforms today because it offers support for a great variety of languages and frameworks for Web applications. However, if a user requires .Net language support would have to choose Aneka or Microsoft Azure. On

the other hand, there are providers e.g. CloudFoundry that is still in its beta version, but promises to be a great provider, which is integrating a great variety of languages and frameworks.

5.3 IaaS

IaaS allows managing a large set of computing resources, such as storing and processing capacity and running applications inside virtual machines (VMs) [43].

Examples of IaaS providers are: Amazon E2C [44], Flexiscale [45], GoGrid [46], Joyent [47], Rackspace [48], Arsys [49].

Amazon E2C: It opens Amazon's large computing infrastructure to its users. The service is elastic in the sense that it enables the user to extend or shrink its infrastructure by launching or terminating new virtual machines (instances) [50].

Flexiscale: It is a complete rebuild of Europe's first Cloud computing platform using Flexiant's

revolutionary Cloud technology. The FlexiScale architecture is modular and can accommodate different implementations of its functionality. Virtual Iron is used which is built on the top of Xen Hypervisor and works as an external management layer for the virtual servers [51].

GoGrid: It hosts Linux and Windows virtual machines managed by a multi-server control panel and a RESTful API. It shares many common characteristics

with Amazon in the classic Cloud computing areas such as: support for multiple operating systems through its own image management, and load balancing and Cloud storage support [46].

Joyent: It provides hosting, infrastructure, and application services for clients looking to run collaborative applications for their users. Joyent offers compute on demand and Web application virtualization and it helps to successfully scale social network websites such as: LinkedIn and Facebook [52].

Rackspace: It provides compute instances similar to Amazon EC2 and VM role of Azure, which are referred as "Cloud Servers". Rackspace also provides a managed service level for Cloud Servers. As part of the managed service, Rackspace is responsible for applying software and security patches for operating system and middleware [53].

Arsys: It is a European provider of ICT services, including Internet presence, Managed Hosting, Cloud Computing Solutions and ICT Infrastructure. Arsys is placed among the leading European companies in technology and innovation [49].

In Table 3 a comparison of the aforementioned providers is presented.

TABLE 3. IaaS providers

<i>Provider</i>	<i>Operating system</i>	<i>Virtualization</i>	<i>Pricing</i>	<i>24/7 support</i>	<i>Hybrid hosting</i>
AMAZON E2C	Linux, Windows	Xen Xen Xen	\$0.080-\$3.580 per hour (vary for different Instance and Regions)	No	No
FLEXISCALE	Linux, Windows	Linux KVM VMware	\$17.00-\$28,866.00 (it depends of units bought 1,000-2,000,000)	Yes	Yes
GOGRID	Linux, Windows	VMware	.PAY-AS-YOU-GO \$0.06/hour(based on running server with 0.5GB of RAM) .PRE-PAID PLANS \$0.02/hour(based on Enterprise Cloud plan and servers with 0.5GB of RAM)	Yes	Yes
JOYENT	SmartOS, windows		\$0.085/hour-\$2.80/hour or \$62.05/month-\$2044/month	Yes	Yes
RACKSPACE	Linux		\$0.022/hour-\$2.64/hour	Yes	Yes
ARSYS	Linux, Windows		2 cents/hour or €10/month or €109.92/year	Yes	Yes

Some providers such as Amazon E2C, Flexiscale and GoGrid use Xen and they have support for operating systems such as Linux and Windows. GoGrid and Flexiscale unlike Amazon E2C, provides technical support included in each account and available 24 hours a day, 7 days a week (24/7 support) and hybrid hosting that allows using a combination of Cloud (virtual), dedicated (physical), and collocated servers connected via a private dedicated network; Other providers such as Rackspace and Arsys use VMware, both have 24/7 support and hybrid hosting. With regard to the price providers e.g. Rackspace, Amazon E2C and GoGrid offer pay-as-you-go.

6. CONCLUSIONS AND FUTURE WORK

Cloud computing has had a great impact in recent years. This new style of computing offers services via internet allowing users to access to programs and development platforms through different devices such as: laptops, smartphones to mention but a few. Cloud computing provides different services at different levels, Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). In recent years, a variety of IaaS, PaaS and SaaS providers have emerged, these providers aim to offer their services to small and large businesses, developers and others potential users. The analysis presented in this paper shows the current more popular SaaS, PaaS and IaaS providers. Also, a comparison with the main features of providers was presented. This comparison showed that some providers offer free plan available, however the use of these lack of features and provide limited, others offer full use of their services but a fee is required. This analysis helps users to select one cloud computing provider according to their needs and budget.

ACKNOWLEDGEMENTS

This work is supported by the General Council of Superior Technological Education of Mexico (DGEST). Additionally, this work is sponsored by the National Council of Science and Technology (CONACYT) and the Public Education Secretary (SEP) through PROMEP.

REFERENCES

- [1] B. Furht and A. Escalante, Handbook of Cloud Computing. Springer, 2010.
- [2] S. Bhardwaj, L. Jain, and S. Jain, "CLOUD COMPUTING: A STUDY OF INFRASTRUCTURE AS A SERVICE (IAAS)," vol. 2, no. 1, pp. 60–63, 2010.
- [3] N. Santos, K. P. Gummadi, and R. Rodrigues, "Towards trusted cloud computing," in Proceedings of the 2009 conference on Hot topics in cloud computing, Berkeley, CA, USA, 2009.
- [4] M. Boniface, B. Nasser, J. Papay, S. C. Phillips, A. Servin, X. Yang, Z. Zlatev, S. V. Gogouvitis, G. Katsaros, K. Konstanteli, G. Kousiouris, A. Menychtas, and D. Kyriazis, "Platform-as-a-Service Architecture for Real-Time Quality of Service Management in Clouds," in Proceedings of the 2010 Fifth International Conference on Internet and Web Applications and Services, Washington, DC, USA, 2010, pp. 155–160.
- [5] R. Prodan and S. Ostermann, "A survey and taxonomy of infrastructure as a service and web hosting cloud providers," in 2009 10th IEEE/ACM International Conference on Grid Computing, 2009, pp. 17–25.
- [6] N. K. Salih and T. Zang, "Survey and comparison for Open and closed sources in cloud computing," arXiv:1207.5480, Jul. 2012.
- [7] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," Journal of Internet Services and Applications, vol. 1, no. 1, pp. 7–18, 2010.
- [8] C. Höfer and G. Karagiannis, "Cloud computing services: taxonomy and comparison," Journal of Internet Services and Applications, vol. 2, no. 2, pp. 81–94, 2011.
- [9] J. K. Kim and J. H. Jeong, "Comparison of Cloud Service Quality Information Publication Based on Cloud Service Quality Model," in 2012 International Conference on Information and Computer Applications (ICICA 2012), Singapore, 2012, vol. 24.
- [10] J. Peng, X. Zhang, Z. Lei, B. Zhang, W. Zhang, and Q. Li, "Comparison of Several Cloud Computing Platforms," in 2009 Second International Symposium on Information Science and Engineering (ISISE), 2009, pp. 23–27.
- [11] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," Computer Security

- Division Information Technology Laboratory National Institute of Standards and Technology, Gaithersburg, Sep. 2011.
- [12] “vmware,” 2012. [Online]. Available: <http://www.vmware.com>.
- [13] “Welcome to xen.org, home of the Xen® hypervisor, the powerful open source industry standard for virtualization.” [Online]. Available: <http://xen.org/>. [Accessed: 14-Jul-2012].
- [14] “KVM.” [Online]. Available: http://www.linux-kvm.org/page/Main_Page. [Accessed: 16-Jul-2012].
- [15] L. Yan, C. Rong, and G. Zhao, “Strengthen Cloud Computing Security with Federal Identity Management Using Hierarchical Identity-Based Cryptography,” in *Cloud Computing*, vol. 5931, M. Jaatun, G. Zhao, and C. Rong, Eds. Springer Berlin / Heidelberg, 2009, pp. 167–177.
- [16] L. Wang, J. Tao, M. Kunze, A. C. Castellanos, D. Kramer, and W. Karl, “Scientific Cloud Computing: Early Definition and Experience,” in *10th IEEE International Conference on High Performance Computing and Communications*, 2008. HPCC '08, 2008, pp. 825–830.
- [17] C. N. Hoefler and G. Karagiannis, “Taxonomy of cloud computing services,” in *2010 IEEE GLOBECOM Workshops (GC Wkshps)*, 2010, pp. 1345–1350.
- [18] “Zoho.” [Online]. Available: <http://www.zoho.com/>. [Accessed: 14-Jul-2012].
- [19] “AtTask.” [Online]. Available: <http://www.attask.com/>. [Accessed: 15-Jun-2012].
- [20] “NetSuite.” [Online]. Available: <http://www.netsuite.com/>. [Accessed: 03-Jul-2012].
- [21] “Bloomfire.” [Online]. Available: <http://www.bloomfire.com/>. [Accessed: 14-Jun-2012].
- [22] “Smartsheet.” [Online]. Available: <http://www.smartsheet.com/>. [Accessed: 15-Jun-2012].
- [23] “Salesforce.” [Online]. Available: <http://www.salesforce.com/mx/>. [Accessed: 15-Jun-2012].
- [24] “Google Apps.” [Online]. Available: <http://www.google.com/intl/es/enterprise/apps/business/>. [Accessed: 15-Jun-2012].
- [25] J. Deyo, *Software as a Service (SaaS) A look at the migration of applications to the web*. 2008.
- [26] A. Goldi, “The emerging market for Web-based enterprise software,” *Sloan School of Management, Management of Technology Program*, Massachusetts, 2007.
- [27] B. Hayes, “Cloud computing,” *Commun. ACM*, vol. 51, no. 7, pp. 9–11, Jul. 2008.
- [28] S. Zhang, S. Zhang, X. Chen, and X. Huo, “Cloud Computing Research and Development Trend,” in *Second International Conference on Future Networks*, 2010. ICFN '10, 2010, pp. 93–97.
- [29] M. Alhamad, T. Dillon, C. Wu, and E. Chang, “Response time for cloud computing providers,” in *Proceedings of the 12th International Conference on Information Integration and Web-based Applications & Services*, New York, NY, USA, 2010, pp. 603–606.
- [30] “Google App Engine.” [Online]. Available: <https://developers.google.com/appengine/?hl=es>. [Accessed: 18-Jun-2012].
- [31] “Aneka.” [Online]. Available: http://www.manjrasoft.com/aneka_architecture.html. [Accessed: 18-Jun-2012].
- [32] “Force.com Cloud Computing Platform.” [Online]. Available: <http://www.salesforce.com/platform/cloud-platform/>. [Accessed: 18-Jun-2012].
- [33] “Windows Azure: Plataforma en la nube de Microsoft.” [Online]. Available: <http://www.windowsazure.com/es-es/>. [Accessed: 18-Jun-2012].
- [34] “Heroku Cloud Application Platform.” [Online]. Available: <http://www.heroku.com/>. [Accessed: 18-Jun-2012].
- [35] “Amazon Elastic MapReduce (Amazon EMR).” [Online]. Available: <http://aws.amazon.com/elasticmapreduce/>. [Accessed: 18-Jun-2012].
- [36] “Cloud Foundry.” [Online]. Available: <http://www.cloudfoundry.com/>. [Accessed: 18-Jun-2012].
- [37] K. Roche and J. Douglas, *Beginning Java Google App Engine*. Apress, 2009.
- [38] Y. Wei, K. Sukumar, C. Vecchiola, D. Karunamoorthy, and R. Buyya, “Aneka Cloud Application Platform and Its Integration with Windows Azure,” *arXiv:1103.2590*, Mar. 2011.
- [39] C. Binnig, D. Kossmann, T. Kraska, and S. Loesing, “How is the weather tomorrow?: towards a benchmark for the cloud,” in *Proceedings of the Second International Workshop on Testing Database Systems*, New York, NY, USA, 2009, pp. 9:1–9:6.
- [40] K. Grant, “Link IDE : A Real Time Collaborative Development Environment,” *San José state university*, 2012.
- [41] S. Richter and C. Thiele, “Benchmarking the Cloud - A Comparison of Lokad on Azure and Elastic MapReduce,” p. 20.
- [42] N. R. Dhanapal, “An architectural approach for the integration of wireless sensor networks with cloud computing for a secure healthcare system,” *OKLAHOMA STATE UNIVERSITY*, 2012.

- [43] A. Li, X. Yang, S. Kandula, and M. Zhang, "CloudCmp: comparing public cloud providers," in Proceedings of the 10th ACM SIGCOMM conference on Internet measurement, New York, NY, USA, 2010, pp. 1–14.
- [44] "Amazon Elastic Compute Cloud (Amazon EC2)." [Online]. Available: <http://aws.amazon.com/ec2/>. [Accessed: 20-Jun-2012].
- [45] "Flexiscale." [Online]. Available: <http://www.flexiscale.com/>. [Accessed: 20-Jun-2012].
- [46] "GoGrid." [Online]. Available: <http://www.gogrid.com/>. [Accessed: 20-Jun-2012].
- [47] "Joyent." [Online]. Available: <http://joyent.com/>. [Accessed: 21-Jun-2012].
- [48] "Rackspace." [Online]. Available: <http://www.rackspace.com/>. [Accessed: 20-Jun-2012].
- [49] "arsys." [Online]. Available: <http://www.arsys.es/>. [Accessed: 22-Jun-2012].
- [50] T. Delf, "An early performance analysis of cloud computing services for scientific computing," 2008.
- [51] A. Haji, A. Ben Letaifa, and S. Tabbane, "Cloud Computing: Several Cloud-oriented Solutions," presented at the ADVCOMP 2010, The Fourth International Conference on Advanced Engineering Computing and Applications in Sciences, 2010, pp. 137–141.
- [52] C. Vecchiola, D. Duncan, and R. Buyya, "The Structure of the New IT Frontier: Market Oriented Computing – Part II," 2009.
- [53] S. A. Baset, "Cloud SLAs: present and future," SIGOPS Oper. Syst. Rev., vol. 46, no. 2, pp. 57–66, Jul. 2012.

ENTORNO DE PRUEBAS PARA SERVICIOS INTERACTIVOS DE TELEVISIÓN MÓVIL BASADA EN EL ESTÁNDAR DVB-H

TEST ENVIRONMENT FOR INTERACTIVE MOBILE TV SERVICES BASED ON DVB-H

**Ing. Gabriel Chanchí
Golondrino**

*Universidad del Cauca, Colombia
Grupo de Ingeniería Telemática
gabrielc@unicauca.edu.co*

**Ph.D. José Arciniegas
Herrera**

*Universidad del Cauca, Colombia
Grupo de Ingeniería Telemática
jlarci@unicauca.edu.co*

Ph.D. Enrique Herrera Viedma

*Universidad de Granada, España
Grupo SCI2S
viedma@decsai.ugr.es*

(Artículo de **Investigación científica y tecnológica**. Recibido el 02-/10/2011. Aprobado el 19/12/2011)

Resumen. La televisión móvil basada en el estándar DVB-H no ha tenido una aceptación semejante a la de la televisión convencional, lo anterior debido a dificultades propias del estándar y a características de despliegue del servicio. Las problemáticas de la televisión móvil se resumen en: necesidad de una red bidireccional alterna o canal de retorno para el consumo de servicios, el tiempo de salto de un canal a otro, el bajo tiempo promedio de uso diario de la televisión móvil, los pocos dispositivos que soportan el estándar y la no existencia de un *middleware* para el desarrollo de aplicaciones interactivas de televisión móvil. El presente artículo plantea un entorno de despliegue y pruebas para servicios interactivos de televisión móvil, que busca responder a los anteriores problemas. El entorno propuesto tiene en cuenta aspectos como la convergencia de servicios en redes WLAN, el uso de economías para recomendación de contenidos y la red celular como canal de retorno o canal de consumo de servicios.

Palabras clave: Canal de retorno; DVB-H; Servicios interactivos; Televisión móvil.

Abstract. Mobile TV based on DVB-H has not had an acceptance similar to the conventional television, above due to difficulties of the standard and the characteristics of service deployment. The problems of mobile TV are summarized in: the need to have a return channel for the consumption of services, the time to switch from one channel to another, the few devices that support the standard and the absence of a middleware for the development of Mobile TV interactive applications. This paper presents a testing and deployment environment for interactive mobile TV services, which wants to respond to the above problems. The proposed environment takes into account aspects such as: the convergence of services on WLANs, the use of economies for content recommendation and the cellular network as a return channel or channel service consumption.

Keywords. Return Channel, DVB-H, Interactive Services, Mobile Television.

1. INTRODUCCIÓN

La amplia difusión de servicios para dispositivos móviles, tales como mensajería, aplicaciones multimedia, juegos, correo electrónico, conectividad inalámbrica, servicios de telecomunicaciones, acceso a redes sociales y otros servicios de internet, ha tenido un éxito considerable en el mercado de las comunicaciones móviles, lo que incrementa la masificación en el uso de estos dispositivos. Es preciso destacar que el hecho de acceder a diversos servicios de internet desde un dispositivo móvil, ha sido impulsado por el crecimiento de internet móvil y la variedad de planes ofrecidos por los operadores móviles. Un dispositivo móvil se concibe como una caja mágica en torno a la que convergen redes y servicios (redes de televisión, sistemas de radio, sistemas de posicionamiento, red celular, redes inalámbricas de área local y servicios de internet), que pueden ser combinados e integrados en distintos escenarios de aplicación y aprovechar las ventajas de la internet móvil.

Dentro de las redes y servicios de telecomunicaciones que convergen en torno a un dispositivo móvil, se destaca el servicio de televisión móvil, que se encuentra ampliamente difundido en Europa a través del estándar DVB-H (Digital Video *Broadcasting Handheld*), derivado del estándar de televisión digital DVB (Digital Video *Broadcasting*), seleccionado por la Comisión Nacional de Televisión en agosto de 2008, como el estándar de televisión digital [1]. El estudio de este estándar es importante para preparar el camino a futuros pilotos de televisión móvil tras el apagón analógico en el país [13].

La televisión móvil ha sido considerada la *killer application* del futuro, por las numerosas oportunidades y ganancias que ha generado en los países europeos, durante los últimos años [2]; sin embargo, a pesar del modelo de negocio exitoso, este servicio no cuenta aún con características similares a las ofrecidas por la televisión convencional en cuanto a facilidad de uso, respuesta rápida a los eventos, tiempo de acceso al contenido, estabilidad en la señal y facilidad de uso. De igual forma, es necesario destacar que al igual que otros estándares de televisión móvil, el estándar DVB-H no provee interactividad bidireccional [14], lo que se convierte en una limitación para vincular aplicaciones o servicios en entornos de televisión, ya que éstos obedecen al modelo petición respuesta. Así mismo, es importante resaltar que, a diferencia de la televisión digital basada en el estándar DVB, donde existe un *middleware* para el desarrollo y despliegue de aplicaciones interactivas, conocido como

MHP (Multimedia Home Platform) [18]. En televisión móvil no existe un *middleware* estándar para el despliegue de aplicaciones interactivas, puesto que cada uno de los proveedores de dispositivos apuesta por *middlewares* propietarios.

Estos aspectos se consideran un problema visible en el ámbito de la televisión móvil, que influye en el despliegue del servicio y en su aceptación por parte de los televidentes. Así, surge la necesidad de un entorno de pruebas y despliegue de servicios interactivos de televisión móvil, en el que se permita la vinculación de servicios que mejoren la experiencia usuario y contrarresten los problemas propios del estándar para aprovechar el marco de la convergencia de redes y servicios en un dispositivo móvil.

Este entorno de pruebas para servicios interactivos surge como respuesta a los problemas planteados anteriormente con las siguientes características: fácil acceso al contenido multimedia a través de un sistema de recomendaciones, vinculación de servicios remotos de apoyo al contenido mediante protocolos livianos y extensibles (REST-JSON), interactividad bidireccional mediante un canal de retorno y acceso a los servicios multimedia e interactivos a partir de un *middleware* abierto. Así, el aporte del presente artículo es la creación de un entorno para el despliegue de contenidos multimedia y servicios interactivos de televisión móvil, soportado sobre una arquitectura base de transmisión de contenidos de televisión [2] [6]. En cuanto al acceso a los contenidos multimedia se agrega una pasarela para la conversión de flujos DVB-H a flujos de *streaming* RTSP (Real Time Streaming Protocol), que solventa el problema de la no existencia de un *middleware* para las aplicaciones móviles. Sobre el acceso a servicios interactivos, se vincula un esquema flexible de televisión móvil, que se fundamenta en el protocolo REST-JSON. Esto facilita el despliegue de servicios en escenarios de televisión, así como la eficiencia en el consumo de servicios interactivos (Protocolos livianos).

El presente trabajo se llevó a cabo en el Laboratorio experimental de televisión digital de la Universidad del Cauca [3], y se encuentra enmarcado dentro del proyecto de investigación ST-CAV (Servicios de T-Learning para el soporte de comunidades académicas virtuales)[4], cuyo objetivo principal es brindar soporte a CAV (Comunidades Académicas Virtuales) en diversos escenarios de televisión, como televisión digital terrestre basada en el estándar DVB (Digital Video *Broadcasting*) y televisión móvil e IPTV.

La estructura de este artículo está definida de la siguiente manera: en la sección 2 se presentan los problemas que motivaron el estudio y la definición de la arquitectura, a partir de algunas experiencias hechas en televisión móvil. En la sección 3 se muestra la arquitectura para la implementación del entorno de pruebas. En la sección 4 se presenta el prototipo de pruebas para servicios de televisión móvil. En la sección 5 se muestra el análisis de los resultados del entorno de pruebas y, por último, en la sección 6, se hacen las conclusiones y trabajos futuros.

2. CONCEPTOS DE TELEVISIÓN DIGITAL

La televisión móvil es definida como la transmisión de contenido audiovisual a dispositivos portátiles [15], de manera independiente al estándar de televisión usado [14] y dispositivo portátil como aquel que cuenta con características de movilidad. Con esta última ventaja y la potencial convergencia de servicios en un dispositivo, la televisión móvil ha sido considerada un modelo prometedor. Sin embargo, su evolución y despliegue dependen de los acuerdos entre los actores de la cadena de televisión: proveedores de contenidos, operadores móviles, proveedores del servicio de televisión y fabricantes de dispositivos. Dentro del ámbito móvil el estándar más difundido para televisión es DVB-H (Digital Video *Broadcasting* Handheld), el cual deriva del estándar para televisión digital DVB (Digital Video Broadcast) y ha tenido acogida en la mayoría de países europeos, gracias al apoyo de los fabricantes de dispositivos, operadores de telefonía y operadores de televisión [2].

El estándar DVB-H permite el despliegue de televisión móvil a través de un proveedor de contenidos, una red de transporte o red de difusión y un cliente móvil con el soporte del estándar [2]; sin embargo, a la hora de hablar de despliegue de aplicaciones o servicios adicionales al contenido, el estándar de televisión móvil, no facilita el manejo de interactividad bidireccional, que es un aspecto fundamental para el consumo de servicios o aplicaciones basados en el modelo petición respuesta.

Si se tienen en cuenta el modelo exitoso de internet y los servicios ofrecidos por la denominada Web 2.0: *chat*, foros, *wikis*, *blogs*, la televisión móvil pretende evolucionar a una tendencia conocida como televisión móvil 2.0 [2], que plantea un modelo de *full* interactividad y acceso, en el que existen múltiples servicios vinculados a los contenidos de televisión móvil;

que, a su vez, pueden ser accedidos desde múltiples redes, en caso de no existir cobertura DVB-H (WLAN, PAN), y brindar la capacidad de participar de manera activa para enriquecer el contenido televisivo, al estilo de la Web 2.0.

El modelo de televisión móvil 2.0, plantea la necesidad de agregar un canal bidireccional o canal de retorno por medio del cual sea posible realizar el consumo de los servicios; si se tiene en cuenta que el estándar de televisión móvil DVB-H sólo permite un modo de interacción básica provista por el canal de difusión (interactividad local) y un conjunto reducido de servicios, entre los que se destaca el de guía interactiva o guía de programación electrónica (EPG). El canal de retorno propuesto en este modelo es el de la red celular (GPRS-EDGE en el caso colombiano), y está presente en el móvil y no requiere inversión adicional en cuanto a infraestructura, puesto que aprovecha la convergencia de redes en el dispositivo.

Según estudios de usabilidad y evaluación del estándar DVB-H, referidos a continuación, los problemas principales de la televisión móvil se resumen de la siguiente manera: la televisión móvil basada el estándar DVB-H provee sólo interactividad local (no bidireccional) [2]; de acuerdo con la revisión del estado del arte no existe un *middleware* estándar para el *desarrollo* de aplicaciones interactivas de televisión móvil basadas en el estándar DVB-H, sólo existe un conjunto de aplicaciones cerradas, propias de los fabricantes de dispositivos móviles, a diferencia de TV digital soportada en DVB, en la que existe el estándar MHP. Según estadísticas presentadas en [5], el uso promedio del servicio de televisión móvil por día es alrededor de 10 minutos, tiempo empleado en cambiar de un canal a otro, más conocido como *zapping*, es aproximadamente de 2 a 2.5 segundos [14][5]; el acceso al contenido multimedia a través de una guía interactiva es tediosa si se considera que se emplea gran parte del tiempo promedio de uso del dispositivo, en la búsqueda de un contenido o canal en un espacio reducido de pantalla [5]; la televisión móvil no brinda una experiencia sencilla como la televisión convencional, en cuanto al manejo de controles simples en una pantalla reducida [5]; el tamaño de la pantalla del dispositivo es pequeña; por ello es necesario distribuir servicios y contenidos de una manera estratégica [2]; la señal de televisión es inestable en regiones con relieve escabroso[5]. Las anteriores falencias tecnológicas han impedido que la televisión móvil sea totalmente un servicio de am-

plia difusión, aceptación y éxito entre los televidentes, como lo es la televisión convencional [5].

De estas argumentaciones, se infiere que es necesario modificar el modelo o adaptar las arquitecturas tradicionales de televisión móvil. Es primordial plantear una arquitectura que da respuesta a algunos de los anteriores problemas de la televisión móvil. Con el presente trabajo se busca contribuir en la solución del problema, en cada uno de los siguientes aspectos:

Garantizar *full* interactividad y convergencia de servicios adicionales. (*Chat*, votación, foros, información de apoyo al contenido) mediante un canal de retorno, que se representa por la red celular o por una red inalámbrica de área local (WLAN).

Facilitar el acceso a los contenidos multimedia para mejorar la forma en la que se hace el salto de un canal a otro (*zapping*). Lo anterior, mediante el uso de un sistema de recomendaciones de contenidos en la interfaz del sistema de televisión móvil, tal como el que se propone en [6]. Los sistemas de recomendaciones se definen como herramientas encargadas de identificar los gustos, necesidades y preferencias de un usuario, de manera que guían de forma personalizada el proceso de escogencia de un ítem u objeto a partir de muchas opciones [19], [20], [21], [22], [23] y [24]. Así, el objetivo del sistema de recomendaciones es presentar un conjunto de contenidos del historial del televidente y evitar el uso del salto de canal.

Suprimir la guía interactiva y promover la premisa Un clic para ver de la televisión convencional. Este aspecto, al igual que el anterior, se busca mejorar con el uso del sistema de recomendaciones, que, al ingresar a la plataforma, proponga en un conjunto de canales o contenidos de acuerdo con el historial del televidente [6].

Es menester destacar que uno de los aspectos fundamentales de estas características es dotar a la arquitectura base de televisión móvil de un esquema para acceder a servicios de soporte y de recomendación de contenidos y tener en cuenta las características de sincronización y las capacidades de procesamiento de un dispositivo móvil, puesto que el esquema de servicios desarrollado en el proyecto ST-CAV y planteado en [16], en el que se propone el uso del protocolo para servicios Web REST y el formato de intercambio de mensajes JSON. REST plantea una arquitectura cliente-servidor en la que

un servicio es visto como un recurso y es identificado a través de una URL, mediante la que se accede con métodos HTTP: GET y PUT. En el acceso a los recursos se intercambian mensajes en formato simple. El formato más difundido es JSON debido a su sencillez, facilidad de procesamiento y amplio uso en redes sociales.

3. ARQUITECTURA TELEVISIÓN MÓVIL

La arquitectura presentada en la Fig.1, es una adaptación y mejora de las arquitecturas base de televisión móvil 2.0 propuestas en [2], [17], [6], a partir de las cuales se han agregado algunas modificaciones en cuanto a: vinculación de un sistema de recomendaciones colaborativo basado en falcsonomías; acceso flexible a servicios interactivos de televisión móvil, a través del protocolo de servicios web REST-JSON e inclusión de una pasarela para la conversión de flujos multimedia DVB-H a flujos de *streaming* RTSP.

Esta arquitectura ha tenido en cuenta las ideas propuestas en la sección anterior, como alternativa para los problemas de la televisión móvil, con el fin de mejorar la experiencia de usuario y el modelo de despliegue de servicios interactivos. En lo que respecta a la infraestructura, el presente trabajo considera la infraestructura propuesta en [2], así como la arquitectura para el despliegue de servicios interactivos de televisión digital *desarrollada* en Laboratorio experimental de televisión digital interactiva de la Universidad del Cauca [3], que cuenta con el soporte para transmisión de televisión digital terrestre y cableada. Gracias a la compatibilidad del estándar DVB-H con el estándar DVB, para la transmisión de contenidos multimedia, basta con adaptar el formato de los flujos de transporte generados por el Servidor de Playout usado en el laboratorio (Opencaster [8], [3]).

En la Fig.1, se presentan los componentes de la Arquitectura base para el entorno de pruebas. Cabe resaltar que la transmisión del contenido multimedia de televisión se hace a través del canal de difusión o canal de *broadcast*, mediante el estándar DVB-H, mientras que el consumo de los servicios interactivos se realiza a través del canal de retorno. En este caso, como un dispositivo móvil puede acceder a internet por medio de la red celular o a través de la red inalámbrica WLAN, éstas se convierten en la mejor opción para servir como canal de retorno. El consu-

mo de servicios por red celular se hace con el protocolo REST (Representational State Transfer), que es una implementación de Servicios web y garantiza el consumo eficiente de datos, comparado con el protocolo SOAP (Simple Object Access Protocol).

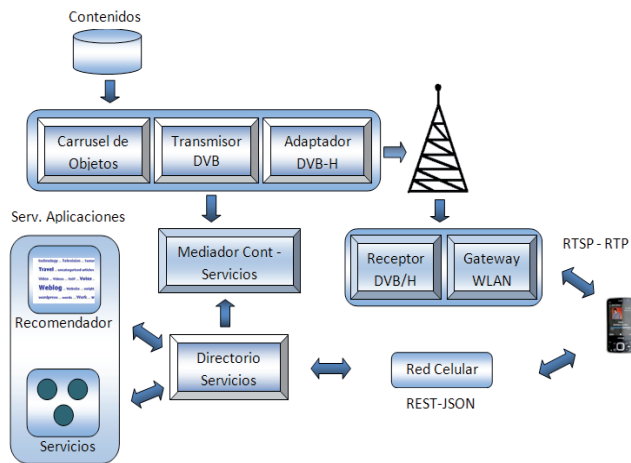


Fig. 1. Arquitectura de televisión móvil

Como no existe un *middleware* abierto y estándar para las aplicaciones interactivas de televisión móvil, esta arquitectura provee una Gateway WLAN encargada de recibir los contenidos DVB o DVB-H y redireccionarlos con el protocolo de IPTV: RSTP (Real Time Streaming Protocol), al que se accede desde una red inalámbrica WLAN o una red LAN con plan de datos en el dispositivo móvil. La Gateway WLAN hace uso de las herramientas para recepción y procesamiento de flujos DVB: DVB Tools (<http://sourceforge.net/projects/dvbtools/>) y dvbsnoop (<http://dvbsnoop.sourceforge.net/>), así como también de la herramienta de reproducción y emisión libre VLC (<http://www.videolan.org/vlc/>) para redireccionar el contenido multimedia con el protocolo RTSP.

Además, esta arquitectura provee un módulo de recomendaciones que se encarga de proponer contenidos al televidente de acuerdo con su historial y valoración o calificación dada a los contenidos visualizados. A continuación se describen los componentes de la arquitectura.

3.1 Directorio de servicios

Contiene un catálogo con los servicios que se han de desplegar en conjunto con el contenido multimedia o de difusión. Estos servicios son indexados al directorio y pueden ser ofrecidos por diversos proveedores

de servicios (servidores de aplicaciones) en internet o en la red local y permiten características de integración y composición. Los servicios indexados son de 2 tipos: asociados o no al contenido; en el primer caso, son desplegados en un determinado momento dentro de un programa de televisión, mientras que en el segundo caso son independientes al contenido y están disponibles durante todo el programa. Se considera el marco del proyecto ST-CAV y los servicios no asociados pueden ser tomados de la Web 2.0: *chat*, notificaciones, foros, *wikis* y dentro de los asociados están: *test*, encuestas, votaciones e información de refuerzo al contenido multimedia (biografías, conceptos). El directorio de servicios es invocado desde el cliente móvil a través de red celular con el protocolo de Servicios web: REST, mediante el formato JSON (Javascript Object Notation) [11].

3.2 Servidor de aplicaciones

Contiene los servicios de los proveedores dispersos a través de internet, agregados de manera individual u orquestada al Directorio de servicios. Este servidor contiene los soportes para televisión móvil y el de recomendación de contenidos, implementados de acuerdo con la arquitectura del protocolo de servicios web REST.

3.3 Recomendador de contenidos

Este módulo se encarga de recomendar contenidos multimedia de acuerdo con el historial de navegación y valoración de los contenidos visualizados del televidente. Para ello hace uso de un algoritmo colaborativo [12] y [24], que tiene en cuenta las valoraciones de los televidentes con gustos parecidos al televidente en cuestión. Los contenidos son presentados en la pantalla del dispositivo mediante una falsonomía, tal como se muestra en [25], y que se define como una clasificación colaborativa por medio de etiquetas simples en un espacio de nombres llano, sin jerarquías ni relaciones de parentesco predefinidas, también es conocida como nube de conceptos, en la que los conceptos de mayor relevancia son presentados con más profundidad o más tamaño de fuente [26]. Esta falsonomía es accedida a través del Directorio de servicios con el protocolo REST y es presentada en pantalla en una de las pestañas del cliente móvil, de tal forma que cada vez que el televidente navegue por esta pestaña se genere una petición de recomendación de contenidos.

3.4 Carrusel de objetos multimedia

Es el módulo sobre el cual viajan los contenidos multimedia adecuados para el formato de televisión móvil, se obtienen en el repositorio multimedia del proveedor de contenidos; de igual manera, este carrusel transporta la guía interactiva de televisión, opera de manera transparente al usuario final, debido a condiciones de usabilidad expresadas en secciones anteriores [3]. La guía interactiva en la presente arquitectura cumple un propósito que es llevar un registro de los contenidos y servicios que están asociados o vinculados en un programa de televisión; el registro de esta información lo realiza el Mediador de contenidos y servicios mediante entradas al documento XML en el que se basa la guía. Todo esto implica describir los momentos de interactividad (tiempos) dentro del programa de televisión. En el Laboratorio de televisión digital de la Universidad del Cauca este proceso lo lleva a cabo el Servidor de Payout, que se implementó mediante el software libre Opencaster.

3.5 Transmisor DVB

En este módulo se prepara la transmisión del contenido empaquetado en el Carrusel de objetos multimedia (contenido y guía interactiva), se dota el contenido de las características de modulación exigidas por el estándar de televisión digital: DVB. Cuando las aplicaciones son asociadas con el contenido se envían por este mismo medio. Estas aplicaciones sirven para complementar el contenido multimedia con conceptos o datos alusivos al programa transmitido.

3.6 Adaptador DVB-H

Este módulo es de suma importancia puesto que se encarga de adaptar los flujos de transmisión DVB a DVB-H, teniendo en cuenta los formatos, tablas y características de transmisión propias de los estándares; es así como el contenido queda listo para ser transmitido de acuerdo con el estándar DVB-H. Gracias a este módulo es posible que en un mismo escenario coexistan el entorno de televisión digital y televisión móvil, recomendación propuesta por la tendencia Mobile TV 2.0 para que el modelo de televisión móvil tenga mayor aceptación.

3.7 Mediador de contenidos y servicios

Este bloque es sustancial porque, cuando se dota el sistema de televisión móvil de interactividad bidirec-

cional por medio de un canal de retorno (red celular), se hace necesario sincronizar esta última con la red de televisión DVB-H. Esta sincronización consiste en vincular servicios y contenidos a través de la guía interactiva (documento XML), de tal manera que cada contenido tenga disponibles unos servicios dentro del programa de televisión. Gracias a esta información almacenada en la guía, el dispositivo móvil obtiene la información de sincronización; para ello debe contar con las librerías necesarias para procesar documentos XML.

3.8 Repositorio de contenidos

Dentro del repositorio de contenidos se almacena información multimedia generada por los proveedores de contenido; estos contenidos se encuentran en los formatos MPEG-2 y MPEG-4 para resoluciones 320x240 y 128x128, adecuados para la transmisión en entornos de televisión móvil.

3.9 Receptor DVB/H – Gateway WLAN

Estos dos módulos se alojan dentro de un solo equipo intermedio cuya función es recibir el contenido DVB o DVB-H y redireccionarlo vía *streaming con el protocolo RTSP (protocolo de streaming)*. Es preciso recordar que la arquitectura propuesta transmite contenidos multimedia con el uso de los dos estándares: DVB y DVB-H. Las diferencias básicas son las características de corrección de errores que provee el estándar DVB-H, así como la posibilidad de evitar la transcodificación.

3.10 Dispositivo móvil

El dispositivo móvil es el encargado de presentar en pantalla los contenidos sugeridos por el recomendador a través de una falcsonomía; tal como se mencionó anteriormente, estos contenidos son presentados en forma de pestañas al estilo de un navegador de internet, ver Fig. 4. Adicionalmente, dentro de la interfaz del cliente móvil se muestran los servicios agregados al contenido multimedia, asociados o no al contenido; los servicios asociados al contenido hacen parte del programa de televisión, mientras que los servicios no asociados se ubican en una región de pantalla en la que no interfieran con el despliegue del contenido. El móvil se encarga de hacer sincronización a partir de la información consignada en la guía interactiva; para ello se hace uso de las librerías para procesar el documento XML de la guía. De igual

forma, para el consumo de los servicios en el móvil, se utilizan las librerías que realizan las siguientes funciones: hacer las peticiones HTTP REST, decodificar los mensajes JSON, realizar las peticiones RTSP a la Gateway WLAN y acceder a la red WLAN. Si se considera que no existe un *middleware* estándar abierto para aplicaciones interactivas de televisión móvil, es posible usar una implementación en cualquier plataforma que cumpla con las condiciones anteriores (Android, Java ME, Symbian, PYS60). Dentro del proyecto ST-CAV se efectuaron 2 implementaciones para acceder a los servicios de televisión móvil: una implementación basada en PYS60 (versión Python para dispositivos Nokia de la serie 60 - <http://www.pys60.org>) para consumir servicios REST-JSON y un Cliente móvil Java ME para acceder a los servicios REST-JSON y al contenido multimedia de la Gateway WLAN.

4. PROTOTIPO DE PRUEBAS O EXPERIMENTACIÓN

El entorno de pruebas que se muestra en la Fig. 2, corresponde a la implementación de la arquitectura de televisión móvil en la parte de transmisión o difusión de contenido. Este entorno tiene en cuenta la ausencia de un *middleware* para televisión móvil, que se encarga de redireccionar, vía *streaming*, el contenido multimedia transmitido para seguir el estándar DVB/DVB-H; además de esto, el entorno de pruebas agrega servicios de soporte mediante el canal de retorno (red celular o red WLAN).

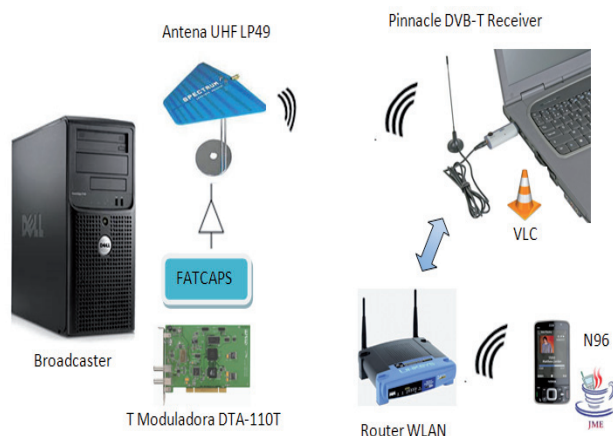


Fig. 2. Entorno de pruebas

El entorno de pruebas está formado por un servidor *Broadcaster* y una Tarjeta Moduladora DTA-110T, encargados de generar y configurar los flujos de

transmisión DVB, que, a su vez, son adaptados al estándar DVB-H por el módulo Fatcaps propuesto en [2]. El flujo de Broadcast es difundido con una antena tipo UHF (LP49), luego, este flujo es recibido en un equipo intermedio o *Gateway* a través de una antena receptora DVB (Pinnacle DVB-T Receiver); el equipo cuenta con el sistema operativo Linux Mint 10 y las herramientas DVB Tools y dvbsnoop, encargadas de recibir, procesar y redireccionar el flujo *Broadcast* en forma de flujo de *streaming* con ayuda de la herramienta de reproducción y emisión VLC. El flujo de *streaming* es puesto a disposición en una dirección IP mediante una URL de tipo RTSP; el Cliente móvil accede al flujo de *streaming* a través de la red WLAN y reproduce el contenido recibido con el API multimedia de Java ME: MMAPi o a través del cliente de *streaming Real Player*, ver Fig. 3. Para acceder a los servicios de soporte, el dispositivo usa la red inalámbrica para consumir los servicios vía REST. La respuesta obtenida de los servicios es en formato JSON y el dispositivo móvil se encarga de procesar los mensajes JSON y presentarlos en forma de pestañas en la pantalla del dispositivo.

El contenido multimedia desde el *Broadcaster* se envía directamente con el estándar DVB, de tal forma que la *Gateway* WLAN lo reciba y redireccione; sin embargo, por hacer uso del estándar DVB-H se garantiza un mayor ahorro en la batería del dispositivo y un mejor control de errores.

5. ANÁLISIS DE RESULTADOS

Si se tiene en cuenta que hay una ausencia de un *middleware* para las aplicaciones interactivas de televisión móvil, se creó un Cliente móvil Java Me para la recepción de contenidos multimedia y servicios de televisión, Ver Fig. 3 y Fig. 4; los contenidos multimedia son redireccionados desde la *Gateway* WLAN vía red inalámbrica a través del protocolo RTSP (Real Time Streaming Protocol), usado en IPTV, ver Fig. 3; por su parte, los servicios son consumidos con el uso del protocolo para servicios Web REST-JSON, ver Fig. 4. Para la recepción de contenidos y servicios, el dispositivo móvil debe contar con el soporte del protocolo RSTP, así como el soporte de conectividad WLAN y Java ME (CLDC 1.1 MIDP 2.0).

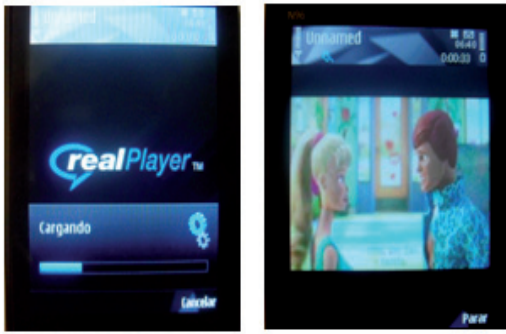


Fig. 3. Acceso a contenidos multimedia

Los servicios desplegados en el entorno de televisión móvil corresponden a los del proyecto ST-CAV, tomados de la Web 2.0 y buscan apoyar la participación de los televidentes en las comunidades académicas virtuales en entornos de televisión, ver Fig. 4. Para el entorno de pruebas y el piloto móvil se tomaron los servicios de: micro-blog y acceso a correo electrónico Gmail. El primer servicio consta de un tablón de mensajes en el que los miembros de una comunidad expresan su opinión acerca del contenido multimedia que se está transmitiendo. Los mensajes tienen ciertas restricciones en el número de caracteres al estilo Twitter; de otra parte, el servicio de acceso al correo permite acceder a los encabezados y mensajes de correo no revisados de la cuenta de Gmail del televidente; los anteriores servicios son presentados en el Cliente móvil a través de un sistema de pestañas, ver Fig. 4, considerando la necesidad de optimizar el tamaño de la pantalla del dispositivo y siguiendo la tendencia de los navegadores de Internet.



Fig. 4. Servicios REST-JSON

En cuanto a los contenidos sugeridos por la plataforma, para facilitar el acceso, en otra de las pestañas

del Cliente móvil se presenta una falsonomía de conceptos o temáticas relacionadas con los contenidos que han visualizado televidentes semejantes al televidente en cuestión. Para ello se partió del concepto de falsonomía de la Web 2.0 o nube de etiquetas, en la que los conceptos de mayor relevancia o mayor popularidad, por parte de los usuarios, son presentados con mayor nivel de profundidad o de tamaño de fuente. Como en Java ME sólo existen tres tamaños para las fuentes, se optó por encerrar el concepto en recuadros de colores proporcionales en tamaño a la importancia de la votación obtenida desde el servicio, ver Fig. 5. De igual manera, se presenta otra visión de una falsonomía en términos de un diagrama de torta, donde cada contenido o temática de interés tiene un porcentaje en la gráfica, ver Fig. 5.

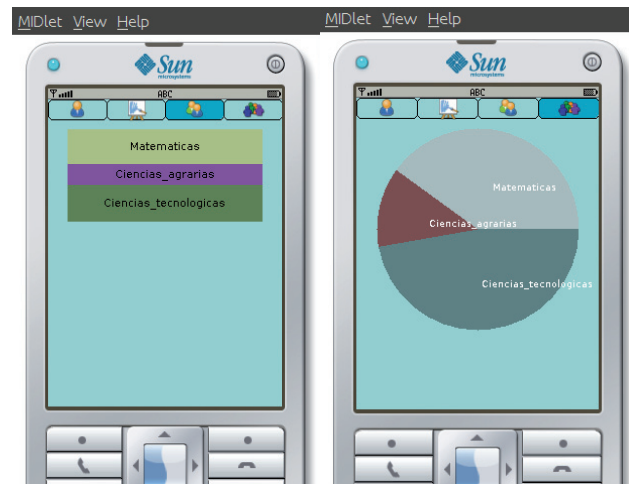


Fig. 5. Falsonomía de contenidos o temáticas

Los servicios de soporte: Micro-Blog, Acceso a correo y Recomendador han sido implementados en el Cliente móvil con el modelo de consumo de la arquitectura REST, basada en peticiones HTTP (GET y PUT) y procesamiento de mensajes JSON; las peticiones fueron desarrolladas en el Cliente móvil usando la librería para Java ME: Rest-Client (<http://www.acidum.de/2008/12/29/j2me-rest-client>). El procesamiento de los mensajes JSON se hizo mediante la librería org.json.me (<https://github.com/upiectec/org.json.me/>). Para el desarrollo de estos servicios en el lado del servidor, se hizo uso del entorno Netbeans 6.9.1 (<http://netbeans.org/community/releases/69/>) que cuenta con el soporte para la creación y despliegue de servicios web REST-JSON en Java y con el servidor de aplicaciones GlashFish V3; en el caso particular del servicio de acceso al correo se hizo uso

del API gmail4j (<http://code.google.com/p/gmail4j/>), el cual es compatible con el lenguaje JSON y permite obtener, por defecto, un conjunto de mensajes del correo de Gmail en este formato.

En la transmisión de contenidos multimedia se usaron los formatos MPEG-2 y MPEG-4 con resoluciones propias de televisión digital, se realizaron transcodificaciones en la *Gateway* WLAN; sin embargo, el contenido fue recibido de manera defectuosa e intermitente en el Cliente móvil, así que se enviaron contenidos multimedia en formatos MPEG-2 y MPEG-4 con resoluciones 320x240 y 128x128 y se hicieron procesos de transcodificación en el lado del transmisor, y se obtuvo una señal estable y de mejor calidad en el caso de la resolución 320x240, con un retraso menor a 0.5 segundos con respecto a la señal DVB-H original. Los contenidos multimedia redireccionados por la *Gateway* WLAN se recibieron en el Cliente móvil a través de la red WLAN con el API de Java ME: MMAPI y su implementación del protocolo RTSP.

6. CONCLUSIONES Y TRABAJOS FUTUROS

El presente trabajo se evaluó en el marco del proyecto ST-CAV de la Universidad del Cauca, dentro del subproyecto de televisión móvil; sin embargo, este entorno de pruebas puede ser adaptado a otros escenarios como el de TDT (Televisión Digital Terrestre) o IPTV, debido a la flexibilidad del modelo de consumo de servicios basado en el protocolo REST-JSON.

El entorno de pruebas presentado en este artículo es un aporte para el estudio y experimentación de servicios interactivos de televisión móvil basada en el estándar DVB-H y se aporta un banco de pruebas para futuros pilotos de TV Móvil.

La arquitectura de despliegue presentada en la sección 3 es una alternativa para los problemas fundamentales de la televisión móvil; principalmente en dos aspectos: las limitaciones del estándar DVB-H en cuanto al manejo de interactividad bidireccional y el fácil y rápido acceso a los canales o contenidos multimedia, si se consideran las demoras en el salto de un canal a otro.

El presente trabajo plantea un avance con respecto a la arquitectura propuesta en [2], porque permite la vinculación flexible de servicios interactivos de televisión móvil mediante el uso del protocolo REST-JSON.

Dadas las limitaciones en sincronización y procesamiento en entornos móviles, el uso de protocolos sencillos para el consumo de servicios como REST-JSON, es una buena alternativa para hacer más eficiente el despliegue de servicios en televisión móvil.

El uso de un recomendador basado en conceptos y no en ontologías ahorra tiempos de procesamiento a la hora de sugerir contenidos a los televidentes, lo cual es un aspecto fundamental para aportar en la solución de los problemas de televisión móvil.

A modo de futuro trabajo se pretende hacer un buscador semántico de contenidos, de tal forma que la selección de contenidos sea una experiencia más personalizada. De igual forma, se pretende integrar estos servicios de búsqueda semántica y recomendación de contenidos en otros entornos de televisión como: TDT (Televisión Digital Terrestre) e IPTV.

AGRADECIMIENTOS

Este trabajo ha sido realizado en la Universidad del Cauca y en la Universidad de Granada-España y ha sido financiado por el Grupo Coimbra, en el marco del programa de apoyo a estancias de investigación para investigadores latinoamericanos. De igual manera, se ha financiado por el programa de apoyo a estudios de Doctorado de Colciencias, Convocatoria 528. Adicionalmente, ha sido parcialmente financiado por los proyecto GESTV, Cod. 1103 521 28387, financiado por Colciencias, Red temática en aplicaciones y usabilidad de la televisión digital interactiva-Redauti, financiado por Cyted y ST-CAV, Cod. 1103 489 25425, financiado por Colciencias y el SENA.

REFERENCIAS

- [1] CNTV. Comisión Nacional de Televisión. Fecha de consulta: agosto de 2010. Disponible en: <http://www.cntv.org.co>.
- [2] Schatz, Raimund., Jordan, Norbert., Wagner, Siegfried. "Beyond Broadcast - A Hybrid Testbed for Mobile TV 2.0 Services".
- [3] Amaya, J., Urbano, F., Campo, W., Arciniegas, J. (2008, septiembre). "Infraestructura tecnológica para un laboratorio experimental de Televisión Digital Interactiva". Presentado en: Congreso Colombiano de Comunicaciones IEEE.
- [4] Proyecto ST-CAV. "Servicios de T-Learning para el soporte de Comunidades Académicas Virtuales". Disponible: http://renata.edu.co/index.php/descargas/doc_details/116-servicios-

- de-t-learning-para-el-soporte-de-una-comunidad-academica-virtual-st-cav.html.
- [5] Knoche, H., McCarthy, J. 2005. "Design requirements for mobile TV". In Proceedings of the 7th international Conference on Human Computer interaction with Mobile Devices & Services (Salzburg, Austria, September 19 - 22, 2005). MobileHCI '05, vol. 111. ACM, New York, NY, 69-76.
- [6] Bär, A., Berger, A., Egger, S., and Schatz, R. 2008. "A Lightweight Mobile TV Recommender". In Proceedings of the 6th European Conference on Changing Television Environments (Salzburg, Austria, July 03 - 04, 2008). M. Tscheligi, M. Obrist, and A. Lugmayr, Eds. Lecture Notes In Computer Science, vol. 5066. Springer-Verlag, Berlin, Heidelberg, 143-147.
- [7] Butkus, A. Petersen, M. "Semantic modelling using TV- Anytime Genre metadata". In Proceedings of euroTV 2007", LNCS Lecture Notes in Computer Science, 2007, vol. 4471. Springer-Verlag, Berlin, Heidelberg, 226-234.
- [8] Avalpa-OpenCaster. "The Free Digital TV Software". Disponible: <http://www.avalpa.com/the-key-values/15-free-software/33-opencaster>.
- [9] Vela, A. Cerón, H. "Plataforma móvil para redes sociales". Fecha de Consulta: agosto de 2009. Tesis Universidad del Cauca. 2009.
- [10] Limbos. "Linux setup for accessing Mobile Broadcast through Online Streaming". Fecha de Consulta: junio de 2010. Disponible: <http://amuse.ftw.at/downloads/dvbh-relay>
- [11] JSON. "Javascript Object Notation". Fecha de Consulta: agosto de 2010. Disponible: <http://www.json.org/>
- [12] Turrin, Roberto., Cremonesis, Paolo. (2010, Junio). "Recomender Systems for Interactive TV". Presentado en: EuroTV 2010 – 8th European Conference of Interactive Tv and Video. Politecnico de Milano.[PDF]. Disponible en: <http://www.euroitv2010.org>.
- [13] Apagón Analógico Colombia, http://www.tecnologiahechapalabra.com/tecnologia/glosario_tecnico/articulo.asp?i=4148
- [14] Acevedo, C., Chanchí, G., Arciniegas, J.: Surveying Mobile Television. International Journal of Communication Networks and Information Security (IJCNIS). Vol. 3, No. 1, April 2011
- [15] L. Paulson. "TV comes to the mobile phone.". IEEE Computer, vol.39, no. 4 pp. 13 -16.
- [16] Chanchí, G. Campo, W. Amaya, J. Arciniegas, J. Esquema de servicios para Televisión Digital Interactiva, basados en el protocolo REST-JSON. Proceedings of the VI Congreso Iberoamericano de Telemática - CITA 2011, ISSN 1519-132X. Gramado, Brazil, May 16-18, 2011.
- [17] Chanchí, G., Arciniegas, J.: Architecture of reference for interactive Mobile TV services supported in personalization and semantic search. Aceptado en The 5th Conference of the Euro-American Association on Telematics and Information Systems EATIS 2010, Panamá, Sep. 22 al 24 de 2010.
- [18] Tique, J., Chanchí, G., Campo, W., Arciniegas, J.: Recomendaciones para la Construcción de Juegos en Entornos de TDi Basados en el Estándar MHP. Memorias del evento Tecnologías Emergentes en Telecomunicaciones y Telemática – TET2010 . Popayán, junio 10 al 12 de 2010.
- [19] C. Porcel, A.G. López-Herrera, E. Herrera-Viedma, "A Recommender System for Research Resources based on Fuzzy Linguistic Modeling. Expert Systems with Applications", 36 (2009) 5173-5183
- [20] C. Porcel, A. Tejada-Lorente, M.A. Martínez, E. Herrera-Viedma, "A hybrid recommender system for the selective dissemination of research resources in a technology transfer office". Information Sciences 184:1 (2012) 1-19
- [21] C. Porcel, J.M. Moreno and E. Herrera-Viedma, "A multi-disciplinary recommender system to advice research resources in University Digital Libraries". Expert Systems with Applications 36:10 (2009) 12520-12528
- [22] A.Felfernig and Gerhard Friedrich, University of Klagenfurt Lars Schmidt-Thieme. "Recommender System". 2007. University of Hildesheim.
- [23] M.Zanker, D.Jannach, and S.Gordea, "Comparing Recommendation Strategies in a Commercial Context". 2007. University Klagenfurt
- [24] P. Melville, J. R. J. Mooney and RamadassNagarajan. "Content-Boosted Collaborative Filtering for Improved Recommendations". 2002. Universidad de Texas.
- [25] G. E. Chanchí, J. L. Arciniegas, E.H. Viedma. "Recommender based on folksonomy for virtual academic communities in Mobile TV". Presented at Colcom 2012. Cali, Colombia, 2012.
- [26] C. Damme, M.Hepp and K.Sioapaes, "FolksOntology: An Integrated Approach for Turning Falcsonomies into Ontologies". Digital Enterprise Research Institute. Universidad de Innsbruck. In Bridging the Gep between Semantic Web and Web 2.0 (SemNet 2007) (2007), pp. 57-70.

SISTEMA HÍBRIDO DE ENERGÍA UTILIZANDO ENERGÍA SOLAR Y RED ELÉCTRICA

HYBRID POWER SYSTEM USING SOLAR ENERGY AND POWER LINE SUPPLY

Mg. Jorge Díaz Rodríguez
Universidad de Pamplona, Colombia
jdiazcu@gmail.com

Ing. Luis Pabón Fernández
Universidad de Pamplona, Colombia
davidpabon@hotmail.es

Ph.D. Aldo Pardo García
Universidad de Pamplona, Colombia
apardo13@hotmail.com

(Artículo de **Investigación científica y tecnológica**. Recibido el 03/10/2011. Aprobado el 19/12/2011)

Resumen. En este trabajo se realiza el diseño y cálculo de los dispositivos de un sistema híbrido de energía, integrado por un generador fotovoltaico (arreglo de paneles solares), una fuente de energía convencional (red eléctrica), un banco de baterías para el almacenamiento de la energía proveniente del arreglo fotovoltaico y los respectivos dispositivos de control de las etapas y la interacción de las mismas. El sistema híbrido se constituye en una solución económica para un suministro confiable de electricidad. En el trabajo se enfatiza el diseño del inversor de potencia, se aportan datos necesarios para la selección o construcción de las partes eléctricas involucradas en la implementación de un convertidor multinivel de 81 escalones. Con el uso de este inversor se garantiza un mejoramiento ostensible de la calidad de la energía eléctrica suministrada por el sistema híbrido.

Palabras clave: Sistema híbrido; Energía fotovoltaica; Inversor; Inversores multinivel; Convertidor *Buck*.

Abstract. This paper presents the design and the devices calculation of a hybrid power system integrated by a photovoltaic (PV) source (solar array), a conventional power source (mains), a battery array to store energy from the PV generator and the stage control devices and its interaction. The hybrid system is an economical solution for a reliable power supply. The paper emphasizes in the design of the power inverter, providing all data required for the selection and construction (all electrical parts involved) of a 81 steps multilevel inverter. With the use of this inverter the energy quality provide by the hybrid system is guarantee.

Keywords: Hybrid system, photovoltaic energy, power inverter, multilevel inverter, buck converter.

1. INTRODUCCIÓN

Los sistemas de energía híbridos utilizan fuentes de energía primaria como la energía solar, eólica, hidroeléctrica, geotérmica, mareomotriz y biomasa, entre otras. Un sistema que utiliza alguna combinación de estas fuentes tiene la ventaja de presentar un mayor equilibrio y estabilidad, disfruta de una mejor calidad de la energía y proporciona salidas estables a partir de sus fuentes primarias y se reduciendo la dependencia de la salida de los cambios externos como las variaciones climáticas y la escasez de combustibles. Se optimiza la utilización de las fuentes de energías alternativas disponibles.

Hacer una revisión del estado del arte sobre sistemas híbridos implica revisar trabajos en la región como [1], [2], [3], [4] en los que se realizan el diseño y cálculo, inclusive con software especializado. En dichos trabajos se enfatiza en determinar las capacidades de los dispositivos; sin embargo, usan el inversor de potencia convencional, que genera una distorsión armónica (THD) considerable. En [5] se mencionan topologías de sistemas híbridos pero no se establece una selección óptima de la fuente de energía primaria. En [6] y [7] se optimiza la selección de la fuente primaria de energía mediante el modelado de los componentes de un sistema híbrido, pero no existe una optimización en la selección de los dispositivos que integran al sistema híbrido. Tampoco se tiene en cuenta la calidad de la energía suministrada. En [8] se hace una revisión del estado del arte de los sistemas híbridos de energía que utilizan como energía primaria la energía solar, eólica, convencional (red) y diesel. Se hace énfasis en los desarrollos futuros con el potencial económico de cada sistema. En este trabajo se pone de manifiesto la capacidad de avance en los sistemas híbridos que usan energía solar. Finalmente, en [9], se describe una metodología para diseñar y evaluar los sistemas híbridos de energía renovable y se visiona la proyección para su futuro.

2. SISTEMA HÍBRIDO DE ENERGÍA

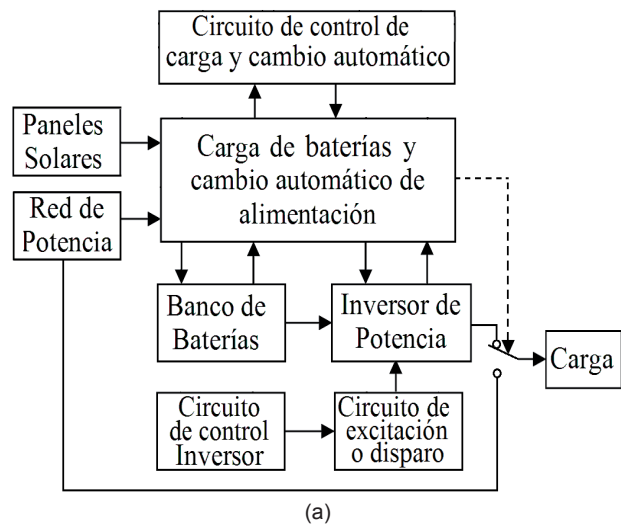
Cualquier sistema de energía híbrido, por lo general, implica la conversión de la energía porque la energía primaria generada es de CD, o, por defecto, convertida a CD, para permitir el uso de las baterías que se utilizan para almacenar energía eléctrica. Luego, para su uso final, se hace necesario convertir la CD en CA, puesto que la mayoría de los electrodomés-

ticos usan esta corriente de 120 V o 220 V y una frecuencia de 50 Hz o 60 Hz. Para esta aplicación se debe usar un dispositivo inversor de potencia que garantice la conversión adecuada. Los inversores de potencia tipo puente son los más utilizados por sus características particulares tanto en servicio monofásico como en trifásico. Las nuevas topologías han permitido la aplicación de los inversores multinivel, con el objetivo principal de aumentar los niveles de voltaje a la salida y obtener ondas de voltaje y corriente de CA casi ideales en función de la cantidad de niveles. Además, se eliminan los armónicos de menor orden que son los que mayor distorsión aportan.

El sistema híbrido diseñado estaría integrado de forma general por las siguientes etapas [10]:

1. Arreglo de paneles solares.
2. Banco de baterías.
3. Circuito de control de carga y cambio automático.
4. Carga de baterías y cambio automático de alimentación.
5. Circuito de control del inversor.
6. Circuito de excitación o de disparo.
7. Inversor de potencia.

Estas etapas están relacionadas, de acuerdo con el diagrama en bloque de la Fig. 1(a) y el esquema funcional del sistema resultante se muestra en la siguiente Fig. 1(b).



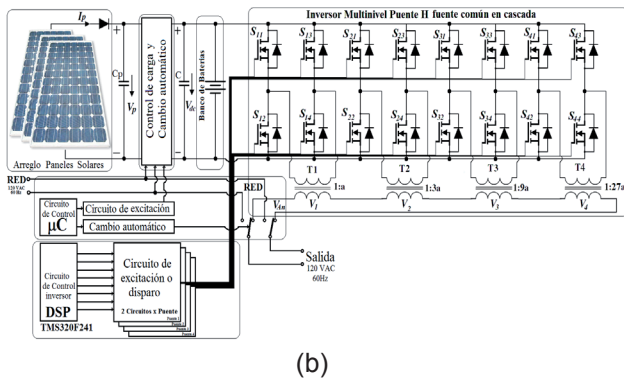


Fig. 1. Sistema híbrido de energía. (a) Diagrama de bloques, (b) Esquema funcional

2.1 Arreglo de paneles solares

Los paneles solares, mediante el efecto fotovoltaico, convierten los fotones en corriente eléctrica, la luz solar causa excitación en los electrones del panel solar y crean un campo electromagnético interno que produce una corriente. Esta electricidad de corriente continua se puede utilizar para producir energía eléctrica limpia puesto que no involucra el uso de minerales fósiles y reduce la contaminación ambiental y el efecto invernadero.

3. METODOLOGÍA DE CÁLCULO DE PANELES SOLARES

Un procedimiento sencillo para dimensionar los paneles solares es [11]:

A. Consumo diario

Puede resultar un cálculo simple, basta buscar la placa de identificación de potencia en los aparatos electrodomésticos para determinar su consumo de potencia en vatios. Si no aparece el dato, se mide con la multiplicación de la corriente y el voltaje para obtener la potencia (se considera el factor de potencia unitario). Luego, se multiplica por el número de horas que se espera que operen en un día normal para determinar el consumo máximo en vatios-hora. Los televisores, computadores, equipos de sonido, relojes y otros aparatos, a veces, utilizan energía incluso cuando están apagados. Por esta razón, y para tener un margen adicional de potencia, se multiplican los vatios-hora por un factor del 1.5, con esto se tiene en cuenta la pérdida de potencia por concepto de caída de voltaje en el cableado de conexión y los ciclos de carga y descarga de la batería.

Otra aproximación, más directa y efectiva, para calcular la energía para el hogar, está representada en las facturas mensuales para obtener un resultado preciso del consumo mensual de energía.

B. Disponibilidad de la luz solar

Este dato es suministrado por Ideam en el Atlas de radiación solar de Colombia, con el que se determina la disponibilidad de la luz solar en la región donde se instalará el sistema. Para Colombia, obtenemos de 4.5-6 horas de sol por día, es suficiente si consideramos el peor mes, el de diciembre, que es de 4.5. En la tabla 1 se relaciona el potencial de la energía solar por regiones para Colombia, basado en el Atlas de radiación colombiano del Ideam y del UPME. Se muestran los datos insolación y radiación solar por regiones [12].

El diseño de esta etapa se fundamenta en el peor dato de radiación establecido para la región andina, según un modelo hecho en el Matlab por los autores, con datos geográficos de *Google Earth* con el que se obtiene la radiación mínima $Rad_{min} = 463.97 \text{ W/m}^2$.

El método utilizado es el equivalente solar que consiste en definir la energía total de un día y hallar el tiempo equivalente si se estableciera una radiación promedio $Rad_{prom} = 1000 \text{ W/m}^2$, con el fin de medir la potencia que se ha de instalar, con la potencia nominal por panel que entrega el fabricante. Las horas de sol diaria (Tsd) aproximada para Colombia son $Tsd=10$ horas.

TABLA 1: Potencial de la energía solar en Colombia por regiones

Región	Insolación (Kwh/m2/año)	Radiación (W/m2)
Guajira	2000 - 2100	684.93 - 719.18
Costa Atlántica	1730 - 2000	592.46 - 684.93
Orinoquia	1550 - 1900	530.82 - 650.68
Andina	1550 - 1750	530.82 - 599.15
Costa Pacífica	1445 - 1550	494.86 - 530.82

La insolación (I) es la cantidad de energía en forma de radiación solar que llega a un lugar de la Tierra en un día concreto (insolación diaria) o en un año (insolación anual), para nuestro caso:

$$I = Rad_{min} \cdot T_{sd} = 463.97 \text{ W/m}^2 \cdot 10 \text{ hr} = 4639.7 \text{ Whr/m}^2 \quad (1)$$

El equivalente día solar (D_s) será:

$$D_s = I / Rad_{prom} = 4693.7 \text{ Whr/m}^2 / 1000 \text{ W/m}^2 = 4.639 \text{ hr} \quad (2)$$

Por tanto, la potencia para instalar (P_{instalar}) está determinada por la razón de la energía nominal (E_n) entre el equivalente del día solar (D_s). La energía nominal se calcula como la relación de la potencia nominal P_n (se considera el factor de potencia unitario) por el tiempo de autonomía de la batería ($t_{\text{autonomía}}$).

$$E_n = P_n \cdot t_{\text{autonomía}} = 2000 \text{ W} \cdot 8 \text{ hr} = 16000 \text{ Whr} \quad (3)$$

$$P_{\text{instalar}} = E_n / D_s = 16000 \text{ Whr} / 4.639 \text{ hr} = 3449 \text{ W} \quad (4)$$

El panel seleccionado sería el KD210GH-2 de la compañía coreana Kyocera, con una potencia nominal de $W_n = 210 \text{ W/m}^2$ a 1000 W/m^2 , con un voltaje nominal de $V_n = 26.6 \text{ V}$ y una corriente nominal $I_n = 7.9 \text{ A}$.

C. Potencia del arreglo de paneles solares.

La potencia necesaria para instalar es de 2000 W (a factor de potencia unitario) y cada panel aporta independiente una potencia de 210 W , por tanto, el número total de paneles necesarios es:

$$N_{\text{paneles}} = P_{\text{instalar}} / W_n = 3449 \text{ W} / 210 \text{ W} = 16.42 \approx 18 \text{ paneles.} \quad (5)$$

Se escoge un número entero par superior al obtenido (16.42 paneles) porque, para conseguir el voltaje de la batería, se requiere de la conexión en serie de 2 paneles.

Si se asume que el sistema operará en condición máxima de carga todo el tiempo, y que el objetivo es que utilice lo menos posible la energía convencional de la red, el número de paneles se aproxima a los citados anteriormente (18 paneles).

El voltaje nominal de la batería seleccionada será de 48 V , por consiguiente, se requieren dos paneles conectados en serie para alcanzar este nivel. Esto hace que los paneles se agrupen en disposiciones de dos, con seis módulos en paralelo. Así, se forma un generador fotovoltaico de 2×9 con los 18 paneles KD210GH-2, y quedarían dispuestos como se muestra en la Fig. 2.

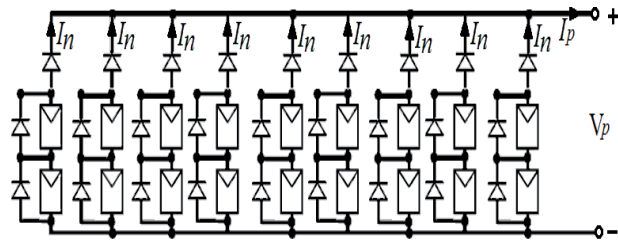


Fig. 2. Arreglo de paneles solares.

El voltaje nominal será de 48 V , aunque el arreglo entregará un voltaje a máxima potencia de:

$$V_{\text{carga max}} = 2V_p = 2 \cdot 26.6 \text{ V} = 53.2 \text{ V} \quad (6)$$

El voltaje máximo de circuito abierto (vacío) es:

$$V_{\text{CA}} = 2V = 2 \cdot 33.2 \text{ V} = 66.4 \text{ V} \quad (7)$$

La corriente máxima que entregará el arreglo de paneles será de:

$$I_{\text{pmax}} = 9I_{\text{cc}} = 9 \cdot 8.58 \text{ A} = 77.22 \text{ A} \quad (8)$$

La corriente de operación a potencia nominal tendrá un valor de:

$$I_p = 9I_n = 9 \cdot 7.9 = 71.1 \text{ A} \quad (9)$$

Según el fabricante, la pérdida de eficiencia entre una incidencia de radiación de 1000 W/m^2 y 200 W/m^2 es de aproximadamente 6% . Lo que se ve balanceado por el sobredimensionamiento de los paneles solares.

Si se tienen en cuenta los parámetros calculados en el dimensionamiento de la batería, el voltaje a plena carga cumple con el máximo establecido por el fabricante de la batería, quien recomienda cargar cada celda con un valor pico máximo de 2.25 V , lo que da un voltaje total de 54 V . Este voltaje está por encima de los 53.2 V nominales del arreglo de los paneles. De igual forma, la máxima corriente que entrega la disposición de paneles no sobrepasa la calculada para el cargador de la batería. Por esta razón, no es necesario implementar un circuito que controle el flujo de corriente porque la optimización del punto de operación de la batería la realizará el control de carga.

D. Determinar la capacidad de la batería

Una posible estimación de la capacidad de la batería es que se acerque a cinco veces el consumo diario del hogar. Se recomienda que el sistema sea capaz de proporcionar energía de forma continua durante

cinco días sin recarga, en el caso de que la aplicación sea totalmente dependiente del sistema de paneles solares. Multiplicar el consumo diario en amperes por 5, y luego dividir el resultado por el voltaje de la batería (12, 24 ó 48 voltios). El resultado es la capacidad recomendada en amperios-hora (A.h) de la batería.

2.2 Banco de baterías

Considerando que la carga que debe soportar el sistema es de 2000 VA, la batería fue dimensionada para proveer $P_n=2000W$, durante una autonomía de $T_{autonomía} = 8$ horas con el fin de garantizar la energía necesaria para alimentar la carga total en un corto tiempo. Se tienen pérdidas en el sistema tales como: regulación en conductores, disipación en dispositivos de control y de potencia, pérdidas en transformadores y otras, es decir, la potencia, más un 5% de seguridad, será la siguiente:

$$P_{batería} = 1.05 \cdot P_n = 1.05 \cdot 2000 = 2100 \text{ W} \quad (10)$$

La autonomía del sistema es de 8 horas, por consiguiente la energía para almacenar será:

$$E_{almacenada} = P_{batería} \cdot T_{autonomía} = 2100 \cdot 8 = 16800 \text{ W} \cdot h \quad (11)$$

La tensión nominal de la batería es 48V, lo que demanda una capacidad nominal de:

$$I_{Ah} = E_{almacenada} / V_n = 16800 \text{ W} \cdot h / 48 \text{ V} = 350 \text{ A} \cdot h \quad (12)$$

Como la conexión de baterías en paralelo no es recomendable, debido a diferencias de voltajes entre las mismas, la conexión de celdas en serie es la más utilizada, aunque exige la adquisición de celdas de la misma capacidad. Para este diseño se podrían seleccionar 24 celdas 5 OPzS 350, de voltaje nominal 2 V y de capacidad 350 A.h a una C10, con un voltaje final por celda de 1.8 V. La selección de esta batería entrega un margen de 10 horas de autonomía a máxima carga y con un máximo de ciclos de vida útil, ya que la operación del sistema exige una tasa de descarga menor a la dada por el fabricante y el voltaje final de descarga está por encima de 1.75V (valor típico de descarga total para una batería de Pb-ácido).

2.3 Circuito de control de carga y cambio automático de alimentación

El circuito de control de carga y cambio automático es un convertidor CD-CD o fuente de alimentación

conmutada, convierte un nivel de voltaje de CD de manera eficiente a otro nivel de voltaje de CD, a través de una etapa intermedia de almacenamiento de energía (inductor), de tal manera que la corriente de carga circule de forma continua y lo más cercano a un valor constante. Las aplicaciones de este convertidor se encuentran en fuentes de poder de baja potencia, alrededor de unos pocos kilovatios, como es nuestra aplicación, en particular como cargador de batería y regulador del voltaje en la carga.

Los convertidores conmutados se clasifican en convertidor reductor o tipo *Buck*, convertidor elevador o tipo *Boost* y convertidor reductor - elevador (*Buck-Boost*) o tipo *Flyback*. El funcionamiento de estos convertidores CD-CD está dado por el almacenamiento temporal de energía en el campo magnético de la inductancia y su posterior entrega. El ajuste del período de tiempo determina la cantidad de energía entregada a la carga [13].

En nuestra aplicación se escogió un conversor tipo *Buck*. Éste transforma el voltaje no regulado en la entrada en un voltaje de salida regulado, siempre inferior al de entrada (reductor).

El voltaje del arreglo de los paneles solares, que se constituye en la entrada del conversor, es de $V_i = 53.2 \text{ V}$, mientras que el voltaje deseado a la salida (voltaje nominal de la batería) es de $V_o = 48 \text{ V}$. El circuito de control utiliza una frecuencia de trabajo fija y genera una modulación por anchura de pulso (PWM) al comparar un nivel de referencia de CD con una señal diente de sierra, como se muestra en la Fig. 3.

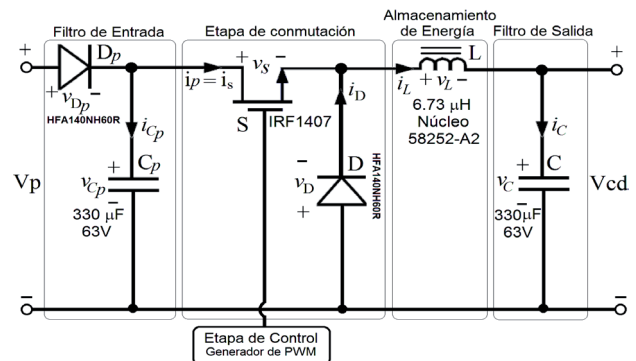


Fig. 3. Convertidor DC-DC tipo reductor (Buck)

La modulación PWM estará generada por un microcontrolador que se basa en los algoritmos programados con este fin, con ciclos de trabajo variables (10, 50, 70 y 100%) de acuerdo con el estado de la carga de la batería. Usualmente, en esta etapa se

programa el algoritmo de determinación del punto de máxima transferencia de potencia (MPPT) a partir de los datos de sensado de voltaje y de corriente.

La polaridad del voltaje en la salida se mantiene igual a la del voltaje en la entrada, característica de este convertidor y que contribuye a minimizar el ruido generado, en lo que influye el filtro LC resultante por la combinación del inductor y el condensador en la salida (filtro pasa bajos). Sin embargo, el voltaje en la entrada recibe los pulsos del transistor de potencia lo que hace que el convertidor produzca un ruido elevado en la entrada. Para disminuir este ruido se pone un condensador que actúa como filtro pasa bajos.

2.4 Carga de baterías y cambio automático de alimentación

Debido a que el sistema que se pretende diseñar es de carácter híbrido, éste debe contar con un circuito capaz de seleccionar la fuente de mayor conveniencia, para este caso en particular, debe cambiar a alimentación convencional sólo cuando el sistema fotovoltaico no posea las condiciones necesarias para operar. Además, debe proveer una carga eficaz y disponer de un circuito que garantice las condiciones óptimas del flujo de energía de los paneles hacia la batería. Se integraron los dos circuitos con el fin de reducir costos y asegurar que las acciones sean realizadas desde un único punto, con el propósito de garantizar la coordinación. El diseño del control está basado en el manejo de tres dispositivos interruptores potencia que conectan los puntos clave del sistema. En la siguiente tabla se presenta la descripción de los interruptores.

TABLA 2:
Dispositivos interruptores de potencia

Interruptor	Conexión	Función	Tipo
S1	Inversor - Carga	Conectar o desconectar la fuente de energía renovable.	Relé
S2	Red CA - Carga	Conectar o desconectar la fuente de energía convencional.	Relé
S3	Paneles - Batería	Desconectar paneles cuando la batería esté en sobrecarga y permitir PWM de alta frecuencia.	Mosfet

Los interruptores S1 y S2 son relés ya que funcionan simplemente con acciones de apagado encendido (on-off), para el interruptor S3 se escogió un mosfet debido a la optimización del flujo de energía: cuando

la batería alcanza un nivel alto de carga (75%), se debe implementar en S3 una modulación por ancho de pulso (PWM) de CD.

Con el fin de evitar gasificación en el electrolito [14] la modulación PWM debe disminuir el ciclo de trabajo a medida que el voltaje de la batería aumente. El circuito diseñado se muestra en la Fig. 4.

El diseño está basado en el uso del microcontrolador de gama media PIC16F876 que, mediante su módulo interno de conversión análogo digital (ADC) de 10 bits, transforma una señal de voltaje proveniente de un sensor de tensión conectado a la batería en una señal lógica que se compara con las referencias establecidas para realizar las respectivas acciones y visualice en un *display* de cristal líquido (LCD) el voltaje actual de carga de la batería y los estados en los que se encuentra el sistema.

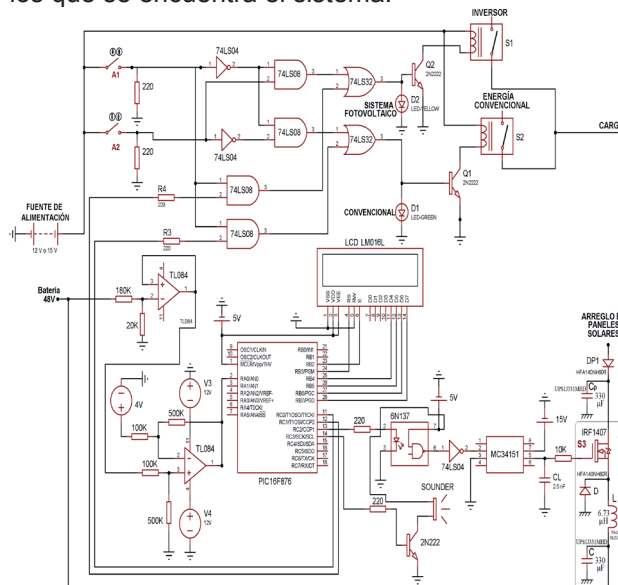


Fig. 4. Circuito control de carga y cambio de alimentación.

Referencias de voltaje

En la tabla 3 se resumen los estados de los interruptores y los rangos de voltajes. En la Fig. 5 se presenta el diagrama de flujo que atiende a las necesidades del microcontrolador frente a este diseño [14].

TABLA 3:

Intervalos de operación

Intervalo	S1	S2	S3 ciclo PWM	Alarma	
$V \leq 43.2$	0	1	100%	0	
$43.2 < V \leq 45$	Descargándose	1	0	100%	0
	Cargándose	0	1	100%	0
$45 < V \leq 47$	1	0	100%	0	
$47 < V \leq 47.5$	1	0	70%	0	
$47.5 < V \leq 48$	1	0	50%	0	
$48 < V \leq 48.5V$	1	0	10%	0	
$48.5 < V \leq 49$	1	0	5%	0	
$49 < V \leq 50$	1	0	0%	0	
$V > 50$	1	0	0%	1	

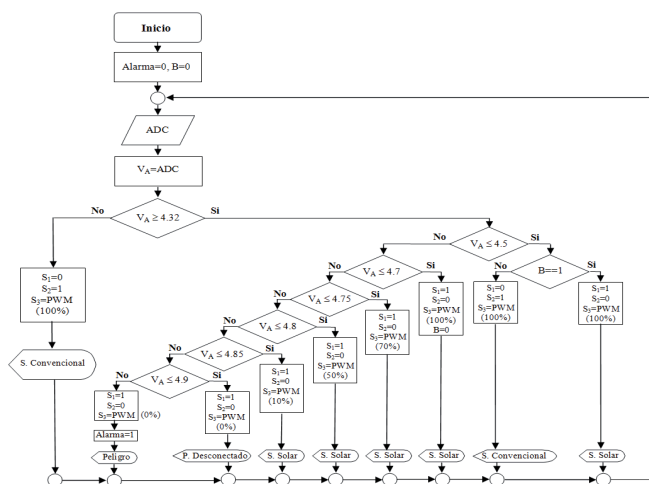
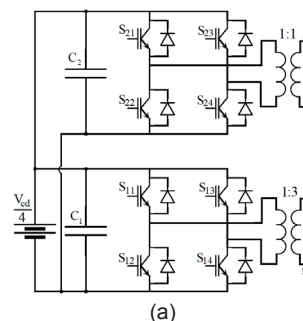


Fig. 5. Diagrama de flujo de la programación

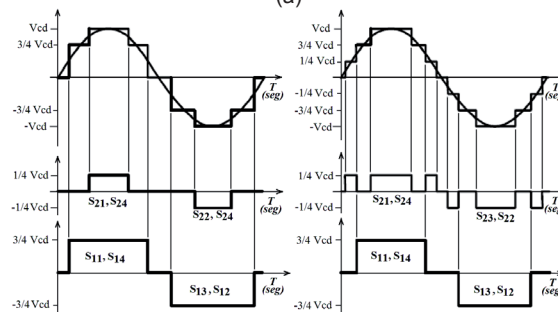
2.5 Inversor multinivel en cascada tipo puente h fuente común asimétrico

El inversor multinivel es una topología reciente de convertidor electrónico de potencia que sintetiza un voltaje deseado a partir de varios niveles de voltajes de CD. Si se utiliza un número suficiente de fuentes de CD o transformadores a la salida, se puede obtener una forma de onda de voltaje casi sinusoidal [15].

El inversor multinivel en cascada tipo puente H fuente común simétrico es similar al del inversor multinivel en cascada tipo puente H multifuente, pero colocando transformadores cuyos devanados primarios están conectados a la salida del inversor y los devanados secundarios están conectados en serie entre sí, como se muestra en la Fig. 6 (a) [16], [17] y [18].



(a)



(b)

Fig. 6: (a) Inversor puente H fuente común asimétrico en cascada de dos etapas (b) operando con cinco niveles (c) operando con siete niveles.

Se puede construir un inversor multinivel en cascada tipo puente H fuente común con la proporción adecuada de voltaje la cual es ajustada con la relación de transformación. Como puede observarse, a diferencia del inversor multifuente, ambos puentes están alimentados desde una misma fuente CD y además los transformadores utilizados poseen diferentes relaciones de transformación. Este inversor de potencia es equivalente al inversor multinivel puente H en cascada con fuente común simétrico; por lo tanto es capaz de generar cinco niveles de voltajes (dos positivos, dos negativos y el cero), como se muestra en la Fig. 6 (b).

TABLA 4:

Transformadores en topología multinivel fuente común.

Característica	Número de niveles				
	3	9	27	81	3k
Número de transformadores	1	2	3	4	k
Relación de transformación	1:a	1:3a	1:9a	1:27a	1:3k-1a

La ventaja de esta configuración frente a la que no utiliza transformadores es que con una sola fuente CD se pueden alimentar todos los puentes del inversor. Esta ventaja es más evidente si se aumenta el número de etapas del inversor. Por ejemplo, en un inversor de potencia multifuente de 4 etapas necesita 4 fuentes independientes de CD mientras que uno

de fuente común de las mismas características sólo utiliza una fuente CD. No obstante hay que mencionar que esta topología no es de uso práctico para aplicaciones de frecuencia variable.

2.6 Circuito de excitación

Para activar o desactivar un *mosfet* es necesario diseñar un circuito que propicie las condiciones necesarias para que se garantice la transición de estados y se consiga en un tiempo corto. Los tiempos de conmutación deben ser pequeños en el orden de los microsegundos [13].

El circuito excitador se obtiene cuando se conectan etapas de acoplamiento óptico, se usa el circuito integrado 6N137 y se adicionan 2 diodos *Zener* 1N4744A de 15 V (D_1 , D_2) que garantizan voltaje constante y regulado en los *mosfet*, como se muestra en la Fig. 7 [19].

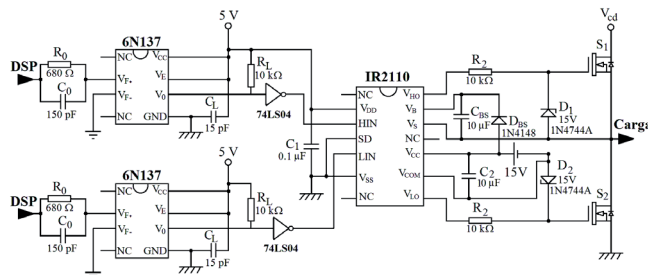


Fig. 7. Circuito excitador para una rama del inversor.

Las señales de control de los dos *mosfet* de una misma rama, deben contar con un tiempo muerto, con el fin de que, en ninguno de los casos, ambos *mosfet* conduzcan en el mismo intervalo de tiempo, ya que esto generaría un cortocircuito. Para la asignación de este tiempo muerto se dispone de un arreglo RC con la siguiente característica:

$$\tau = R_0 \cdot C_0 = 680 \Omega \cdot 150 \text{ pF} = 102 \text{ nSeg} \quad (13)$$

Este circuito asigna un tiempo muerto inferior a los microsegundos para garantizar que no exista la posibilidad de cortocircuito en rama alguna. Para reducir inductancias parásitas y tener un óptimo funcionamiento del *driver* el fabricante provee las siguientes consideraciones para las capacitancias. Estas consideraciones se encuentran en las recomendaciones del fabricante, la compañía *International Rectifier* (*IR*).

Teniendo en cuenta la hoja de característica del *driver* y del *mosfet* de mayor potencia:

$$C \geq 0.629 \mu\text{F}. \quad (14)$$

La recomendación para el condensador C_1 es que este condensador debe ser como mínimo 10 veces mayor al calculado para C , por tanto, los dos condensadores C y el condensador C_1 serán de $10 \mu\text{F}$.

En la entrada del *driver* se debe ubicar un condensador entre los pines V_{DD} y V_{SS} del *driver*, estos pines son la alimentación del nivel lógico y la tierra con el fin de eliminar rizado en la fuente de alimentación, el condensador de conexión típico para el puente H es de $4.77 \mu\text{F}$. El condensador empleado para la realización de este diseño es de $C_{BS} = 10 \text{ mF}/50\text{V}$.

La hoja de características del circuito optoacoplador 6N137 de *Hewlett Packard* (*HP*) en sus curvas de características sugiere que para conseguir un tiempo de demora de $T_{\text{d optocoplada}} = 30 \text{ nS}$ los valores de los componentes son: $CI = 15 \text{ pF}$ y $RL = 350 \Omega$. Esta disposición asegura un tiempo de respuesta de:

$$T_{\text{resp max}} = T_{\text{d optocoplada}} + T_{\text{d NOT}} + T_{\text{d driver}} = 30 \text{ nS} + 15 \text{ nS} + 150 \text{ nS} = 195 \text{ nS} \quad (15)$$

4. CONCLUSIONES

La calidad de la energía de los sistemas fotovoltaicos depende de la onda de tensión de la señal de voltaje de salida del inversor, es decir, si se disminuye el contenido armónico de esta señal se logrará mejorar sustancialmente la calidad de la energía entregada por estos sistemas. En este trabajo la estrategia empleada para optimizar la calidad de energía de los sistemas fotovoltaicos, es utilizar un inversor multinivel que tenga un contenido armónico muy bajo, para lo que se seleccionó el caso de un convertidor multinivel de 81 escalones.

La batería es el bloque del sistema híbrido de energía que presenta menor tiempo de vida útil y se convierte en un componente crítico. Cuando se diseña un sistema que utilice baterías estacionarias, no se debe escatimar en otorgar todas las garantías para que opere con un flujo de carga óptimo y que su descarga no supere los niveles ni los tiempos recomendados por el fabricante. De allí la aparente complejidad del circuito diseñado para la carga de la batería. El uso de microcontroladores en el sistema de control de carga, ofrece variantes como la modulación PWM de ciclo de trabajo variable, que permite optimizar el flujo de energía que proviene de los paneles solares y facilita una mejor visualización de los estados en los que el sistema está operando.

La utilización de un sistema híbrido como el diseñado es importante si se aplica en el desplazamiento de cargas constantes tales como equipos de refrigeración, ventiladores y aparatos que deben operar en lugares donde la radiación solar es alta. Así, se disminuiría la carga alimentada por la red convencional y se aporta en la solución de la problemática medio ambiental.

REFERENCIAS

- [1] N. Sogari. "Diseño de un sistema híbrido solar-eólico para proveer de energía a una comunidad". Comunicaciones Científicas y Tecnológicas, Dpto. de Física - Facultad de Ciencias Exactas y Naturales y Agrimensura - UNNE, Universidad Nacional del Nordeste, Argentina, 2003.
- [2] P. Tagle S. y J. Duque R. "Diseño preliminar de un sistema híbrido de suministro de energía doméstica para una comunidad aislada en base al modelado en el software HOMER". Facultad en Ingeniería Mecánica y Ciencias de la Producción, Repositorio de la Escuela Superior Politécnica del Litoral, Guayaquil, Ecuador, 2003.
- [3] J. González Espinosa, J. I. Véliz Alonso y Rincón Toledo. "Diseño de un sistema híbrido con base en energías renovables para garantizar 5 kW a un consumidor". Facultad de Ingeniería Química-Mecánica, Universidad de Matanzas, Cuba, 2009.
- [4] E. Gutiérrez, D. Aguirre, J. Carpio y J. L. Jaramillo. "Sistemas mixtos fotovoltaicos - energía eléctrica convencional para iluminación: bloque de storage de energía, regulador de carga e inversor". Universidad Técnica Particular de Loja, Loja, Ecuador, 2010.
- [5] J. Rocabert Delgado, S. Busquets Monge y J. Bordonau Farrerons. "Sistema Autónomo de Generación de Energía Renovable". Memorias de las XVII Jornadas de Conferencias de Ingeniería Electrónica, Universidad Politécnica de Cataluña, Terrassa, España, 2011.
- [6] M. Muselli, G. Notton and A. Louche. "Design of hybrid-photovoltaic power generator, with optimization of energy management". *Solar Energy*, Volume 65, Issue 3, 1 February 1999, pp. 143-157.
- [7] R. Dufo López. "Dimensionamiento y control óptimos de sistemas híbridos aplicando algoritmos evolutivos". *Tesis doctoral*, Universidad de Zaragoza, Zaragoza, 2007.
- [8] P. Nema, R. K. Nema and S. Rangnekara. "A current and future state of art development of hybrid energy system using wind and PV-solar: A review". *Renewable and Sustainable Energy Reviews*, Volume 13, Issue 8, October 2009, pp. 2096-2103.
- [9] M. K. Deshmukh and S. S. Deshmukh. "Modeling of hybrid renewable energy systems". *Renewable and Sustainable Energy Reviews*, Volume 12, Issue 1, January 2008, pp. 235-249.
- [10] H. L. Gasquet. "Sistemas fotovoltaicos mixtos e híbridos". Asociación Solar de El Paso EPSEA, Texas, 2008.
- [11] H. L. Gasquet. "Manual sobre los sistemas fotovoltaicos". Asociación Solar de El Paso EPSEA, Texas, 2007.
- [12] H. Rodríguez Murcia. "Desarrollo de la energía solar en Colombia y sus perspectivas". *Revista de Ingeniería*, Facultad de Ingeniería, Universidad de los Andes, Bogotá, Colombia, No. 28, July/Dec., 2008.
- [13] N. Mohan, M. Undeland y W. Robbins. "Electrónica de potencia convertidores, aplicaciones y diseño". 3^{ra} Edición, México D.F., 2009.
- [14] H. Wayne Beaty. "Handbook of electric power calculations". 3rd Edition, New York: McGraw-Hill, USA, 2001.
- [15] A. Pardo y J. L. Díaz. "Aplicaciones de los convertidores de frecuencia. Estrategias PWM". Universidad de Pamplona, Editorial JAVA E.U., Colombia, 2004.
- [16] J. Rodríguez, J.-S. Lai. and F. Z. Peng. "Multilevel Inverters: A Survey of Topologies, Controls, and Applications". *IEEE Transactions on Industrial Electronics*, Vol. 49, No. 4, August 2002.
- [17] A. A. Breton Schuwirth. "Diseño y construcción de un inversor trifásico multinivel de cuatro etapas para compensación armónica y de reactivos". Pontificia Universidad Católica de Chile, Trabajo de grado, Santiago de Chile, 2003.
- [18] C. M. Elgueta Díaz. "Aplicación de un inversor multinivel como variador de frecuencia de un motor de inducción trifásico". Pontificia Universidad Católica de Chile, Tesis de Maestría, Santiago de Chile, Julio 2005.
- [19] J. L. Díaz y L. D. Pabón. "Diseño y análisis de un inversor de potencia multinivel para la aplicación en un sistema híbrido de energía". Trabajo de Categoría Profesor Asociado. Universidad de Pamplona, Colombia, 2011.

CRITERIOS PARA LA EVALUACIÓN DE SUPERFICIES MULTITÁCTILES Y SUS APLICACIONES

EVALUATION CRITERIA FOR MULTI-TOUCH SURFACES AND THEIR APPLICATIONS

Mg. Yenny A. Méndez

*Universidad de San Buenaventura, Colombia
yamendez@usbcali.edu.co*

Ph.D. César A. Collazos

*Universidad del Cauca, Colombia
ccollazo@unicauca.edu.co*

Ph.D. J. Alfredo Sánchez

*Universidad de las Américas Puebla, México
alfredo.sanchez@udlap.mx*

Ph.D. Luis Merchán Paredes

*Universidad de San Buenaventura, Colombia
lmerchan@usbcali.edu.co*

Mg. Yazmín Magallanes

*Universidad de las Américas Puebla, México
yazmin.magallanesvz@udlap.mx*

Mg. Ariel Molina

*Universidad de las Américas Puebla, México
ariel.molinara@udlap.mx*

(Artículo de **Investigación científica y tecnológica**. Recibido el 21/10/2011. Aprobado el 19/12/2011)

Resumen. Las superficies multitáctiles son tecnologías que empiezan a tener un elevado uso. Se toman como referentes aspectos utilizados para la evaluación de usabilidad de sistemas interactivos, así como las características que los investigadores y desarrolladores sugieren que deberían tener las superficies multitáctiles y las aplicaciones desarrolladas. Se presenta una propuesta inicial de criterios que deberían tenerse en cuenta para incluir aspectos de ergonomía, usabilidad y colaboración en el momento de realizar evaluaciones para estas tecnologías. Se ilustra cómo se pueden aplicar los criterios propuestos a través de un caso de estudio que cuenta con resultados promisorios.

Palabras clave: Criterios; Ergonomía; Usabilidad; Colaboración.

Abstract. This paper proposes a set of evaluation criteria that can be applied to assess ergonomic, usability and collaboration aspects of multitouch surfaces and their applications. Multitouch surfaces are technologies that are becoming popular among users. Our proposal takes into account elements that are used for usability evaluation of interactive systems as well as features that have been suggested by researchers and developers for multi-touch surfaces and their applications. We illustrate how our proposed criteria can be applied by means of a case study, which has produced encouraging results.

Keywords: criteria, ergonomics, usability collaboration.

1. INTRODUCCIÓN

El uso de las superficies táctiles se ha incrementado considerablemente en los últimos años y en diversos dominios. La gran mayoría de estas superficies se aplican en sistemas altamente visuales y controladas por contactos y gestos sobre la superficie del sistema, para habilitar a los usuarios a que interactúen directamente con los objetivos con sus propias manos [1].

Gracias a que la evaluación de usabilidad se ha enfocado en evaluar aspectos relacionados con la efectividad, eficiencia y satisfacción del usuario de sistemas interactivos, generalmente relacionados con aplicaciones para equipos de escritorio, dispositivos móviles y, en gran medida, para entornos web, se hace necesario enfocar las investigaciones en temas atinentes a pruebas de evaluación de usabilidad en las aplicaciones desarrolladas para superficies multitáctiles.

A partir de las opiniones de desarrolladores de superficies multitáctiles y de varios proyectos de investigación alrededor de la temática, se presentan los resultados iniciales de una investigación cuyo propósito es identificar y agrupar criterios para tener en cuenta en el momento de evaluar aspectos de ergonomía en superficies multitáctiles, además de la usabilidad y los factores colaborativos en aplicaciones para superficies multitáctiles. En la siguiente sección se dan a conocer algunos de los trabajos relacionados con el proyecto; posteriormente, se presenta información detallada de los criterios y su respectiva clasificación, así como un caso de estudio en el que se usan los criterios. Finalmente, se presenta el trabajo futuro y las conclusiones.

2. TRABAJOS RELACIONADOS

Actualmente no se cuenta con una amplia cantidad de directrices para ayudar a los diseñadores en el proceso de construcción de aplicaciones para mesas interactivas [2]. Como contribución a esta necesidad, se estableció un lenguaje de patrones [2] que sirve como referente para la creación de superficies. Los patrones propuestos se clasifican en las siguientes categorías [2]: ergonómicos, de interfaz, de usabilidad y colaboración específica. Adicionalmente, se propusieron patrones de extensión de entradas y patrones de escenarios especiales. Cada uno de los patrones tiene información sobre el diseño de super-

ficies multitáctiles y ofrecen solución para problemas recurrentes.

En la industria, Microsoft ha publicado un conjunto de las mejores prácticas para la industria, relacionadas con el desarrollo de superficies multitáctiles [3], las que se refieren a los principios básicos para una óptima interacción con el usuario en las aplicaciones multitáctiles, desarrolladas para Microsoft Surface.

En [4] se incluye una propuesta de un conjunto de lineamientos para el diseño de aplicaciones de aprendizaje colaborativo para superficies. Los lineamientos se relacionan con el diseño del entorno físico y las características específicas de la interfaz del software, teniendo en cuenta las necesidades especiales que se requieren para recoger información contextual. Parte de esta investigación se basó en [5], en el que se establecieron las directrices para sistemas basados en superficies interactivas, orientadas particularmente a apoyar la colaboración.

3. PROPUESTA DE CRITERIOS

De estos trabajos citados, se identificaron criterios que se sugieren durante la evaluación de superficies multitáctiles y aplicaciones desarrolladas para estas tecnologías.

A manera de ejemplo, en [2] se menciona que uno de los inconvenientes de la interacción por contacto directo en las mesas, es la fatiga que se puede presentar en los brazos de los usuarios, asunto que está relacionado con la ergonomía de la mesa. De esta información se proponen dos criterios:

Los usuarios se sienten cómodos cuando interactúan con los objetos en la superficie.

La altura de la superficie es adecuada para interactuar con los objetos.

Los criterios propuestos se han de agrupar en categorías: ergonomía, usabilidad y colaboración. (Ver Fig. 1). Se presenta a continuación la descripción detallada de cada categoría.

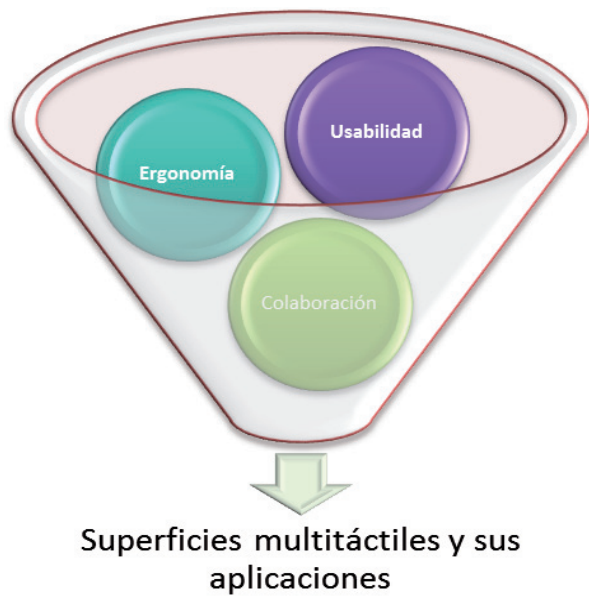


Fig. 1 Categorías propuestas para la evaluación de superficies multitáctiles

3.1 Categorías de clasificación

Ergonomía

Se incluyen criterios mediante los cuales sea posible identificar aspectos relacionados con la ubicación y diseño de la superficie multitáctil, de tal manera que su uso no genere incomodidad o problemas físicos para los usuarios, mientras interactúan con las aplicaciones.

Usabilidad

En esta categoría se agrupan criterios que permitan evaluar que los elementos en las interfaces de las aplicaciones sean agradables para el usuario, produzcan satisfacción de uso y las respuestas sean efectivas y eficientes.

Colaboración

Se incluyen en esta categoría los criterios para evaluar si las aplicaciones potencian un trabajo colaborativo entre los participantes.

3.2 Criterios

Estos son los criterios sugeridos en los casos en los que se requiera evaluar algún aspecto relacionado con las categorías mencionadas. Se presentan a continuación algunos de los criterios para las diferentes categorías:

Categoría: Ergonomía

La inclinación de la mesa es adecuada para interactuar con la aplicación.

Se cuenta con un espacio cómodo para interactuar con los objetos.

Si participan varios usuarios, cada uno puede interactuar con sus objetos tranquilamente.

La altura de la superficie es adecuada para interactuar con los objetos.

El tamaño de la mesa es coherente para interactuar con los objetos.

Los usuarios se sienten cómodos los objetos en la superficie.

Categoría: Usabilidad

Es posible ampliar o reducir el tamaño de los objetos en la interfaz.

Si se amplía el tamaño de los objetos, no se entorpece el trabajo de los demás participantes.

El tamaño del texto en los objetos es cómodo para su lectura.

El tamaño de los objetos es adecuado para interactuar con ellos.

Es posible identificar al usuario que realiza una acción específica.

Los usuarios pueden orientar manualmente los elementos en la interface, de tal manera que tengan buena visibilidad y legibilidad.

La ubicación de los elementos genera comodidad en los usuarios.

Cada uno de los usuarios puede trabajar con sus propios elementos.

La distancia de los objetos con los que interactúa el usuario es adecuada.

Al tocar un objeto se obtiene una respuesta inmediata por parte de la aplicación.

La interacción con los objetos en la superficie es precisa.

Los ademanes utilizados son similares a los empleados en otras aplicaciones.

Las acciones que realizan los usuarios en la aplicación son similares a las ejecutadas en el mundo real.

Si se presentan nuevos ademanes, éstos no producen cargas cognitivas adicionales en los usuarios.

Los ademanes “no comunes”, presentan algún tipo de ayuda para los usuarios, cuando interactúan por primera vez.

La interpretación de las indicaciones sobre ademanes necesarios para realizar funciones en el sistema es intuitiva.

Los ademanes propuestos no implican esfuerzo físico innecesario para los usuarios.

Los objetos en la aplicación son similares a los del mundo real.

Se presenta una respuesta rápida cuando el usuario lo solicita (interactúa con algún objeto de la superficie).

Las tareas más relevantes a realizar, son fáciles de encontrar en la aplicación.

Las metáforas de interacción utilizadas en la aplicación son consistentes.

El primer contacto de interacción entre los usuarios y los objetos se realiza de manera intuitiva.

En todo momento el usuario sabe lo que tiene que hacer con los objetos.

No se presentan controles adicionales que reemplacen la manipulación directa.

Las tareas son claras y las opciones disponibles se enfocan en su buen desempeño.

Las acciones ocurren por solicitud explícita del usuario.

Se presenta en la aplicación un agradable y adecuado diseño visual.

Si se presenta retroalimentación para alguna acción del usuario, ésta no genera ambigüedad.

Categoría: Colaboración

Es posible manejar un “espacio privado” para que los usuarios interactúen con sus propios objetos.

Los elementos requeridos por el usuario se despliegan en zonas cercanas y no entorpecen el trabajo de los demás.

Los elementos potencian un trabajo colaborativo entre los participantes.

Con los elementos disponibles se puede realizar

trabajo colaborativo entre los participantes.

Se puede tener una participación equitativa entre los usuarios, a partir del uso de los objetos disponibles.

Los objetos son suficientes para que los participantes alcancen un objetivo común.

En caso de que se tenga un objeto compartido por los usuarios, éste se puede ubicar en el centro, de tal manera que les permita un acceso fácil a los participantes.

Si el objeto es “propiedad” del usuario, es posible ubicarlo lo más cercano a él.

Se puede restringir la modificación de los objetos, en caso de que el usuario propietario lo requiera.

Los evaluadores de aplicaciones para superficies multitáctiles podrían hacer uso de los criterios listados anteriormente para apoyar sus procesos de evaluación y encontrar problemas relacionados específicamente con estas superficies.

4. CASO DE ESTUDIO: EVALUACIÓN DE APLICACIONES PARA SUPERFICIES MULTITÁCTILES

Los criterios identificados y clasificados durante la investigación, fueron utilizados para evaluar la aplicación InnovIMMS [6], que apoya la producción e innovación de ideas y la colaboración porque interactúa con superficies multitáctiles. La aplicación permite la manipulación de tres clases de objetos: texto, imágenes y lienzos para pintar. Se incluye en la aplicación un menú de abanico y otros elementos gráficos para acceder a los recursos y realizar tareas sobre los objetos [7].

Las interfaces evaluadas en la aplicación fueron: interfaz inicial, menú, submenú, insertar imagen, insertar texto, insertar dibujo a mano alzada, barra de herramientas y ayuda ademanes.

Durante la evaluación, un grupo de seis evaluadores, cuatro de ellos investigadores experimentados en temas de usabilidad y dos desarrolladores de sistemas interactivos (con habilidad en el uso de aplicaciones para superficies multitáctiles), interactuaron con la aplicación. Posteriormente, y de manera individual, los evaluadores asignaron a cada uno de los criterios un valor cuantitativo para estimar la severidad con

que aparecen los problemas (ver Tabla 1). Después se hicieron comentarios que justifican su valoración.

TABLA 1
Escala grado de severidad

Escala	Concepto
0	No es un problema presente en la aplicación.
1	Problema “cosmético”: no necesita ser resuelto a menos que se disponga de tiempo extra en el proyecto.
2	Problema menor: darle pronta solución tiene baja prioridad.
3	Problema mayor: es importante arreglarlo, se le debe dar alta prioridad.
4	Problema catastrófico: es imperativo arreglarlo antes de que el producto sea liberado.

Finalmente, el equipo de evaluadores, con el análisis de resultados, estableció los problemas identificados en la aplicación y generó recomendaciones para su solución.

De las calificaciones dadas por los evaluadores, se identificaron los criterios en los que había mayores problemas con la aplicación. En la Tabla 2 se presentan algunos de los criterios ordenados de manera descendente sobre su promedio.

TABLA 2
Promedio de los criterios evaluados

Criterio	Valor
Es intuitivo para los usuarios saber qué acción ejecutar con los objetos.	2.83
El tamaño de la mesa es adecuado para interactuar con los objetos.	2.67
Los ademanes utilizados son similares a los empleados en otras aplicaciones.	2.50
Los usuarios se sienten cómodos al interactuar con los objetos en la superficie.	2.33
En todo momento el usuario sabe lo que tiene que hacer con los objetos.	2.33

A partir de los criterios que tuvieron un mayor promedio y de los comentarios expresados por los evaluadores, se identificaron los problemas a los que debería dárseles pronta solución. Entre los problemas

identificados en la aplicación se encuentran:

Movimiento de los objetos con el uso de dos dedos.

Incomodidad por el tamaño del teclado.

Difícil ubicación de ayuda brindada para el usuario.

Difícil encontrar las opciones asociadas con los objetos.

Dificultad en el uso de la navegación del menú.

Falta de simplicidad para navegar entre nodos.

No hay claridad del ícono para creación de enlaces.

Menús incompletos o inaccesibles.

5. TRABAJO FUTURO

De la investigación inicial presentada se obtuvo un conjunto de criterios que requieren validez en un trabajo futuro. Esta validación es preciso hacerla en términos de pertinencia y de importancia para evaluaciones de superficies multitáctiles y sus aplicaciones.

Una vez se obtenga un conjunto más refinado de criterios es menester proponer su priorización para establecer aquellos que son indispensables considerar en el momento de hacer evaluaciones.

6. COMENTARIOS FINALES

Los criterios propuestos son un referente para realizar evaluaciones en las que se requiera identificar aspectos relacionados con la ergonomía, la usabilidad y la colaboración en superficies multitáctiles y en sus aplicaciones. A partir de los criterios y de las justificaciones respecto a la valoración dada para tales criterios, se pueden identificar problemas relacionados directamente con superficies multitáctiles y las aplicaciones desarrolladas para éstas.

AGRADECIMIENTOS

Este trabajo ha sido financiado parcialmente con los recursos de *Latin American and Caribbean Collaborative ICT Research* mediante el apoyo para Short Stays Program 2011, el Programa de Aplicaciones Avanzadas CUDI-Conacyt a través del proyecto “Superficies Interactivas como Apoyo a Procesos de Innovación”, el Ministerio de Educación Nacional de Colombia (MEN) y CINTEL

en el marco del proyecto “Mecanismo para la interpretación de emociones en la evaluación de usabilidad de entornos virtuales de aprendizaje” y el proyecto Lineamientos de usabilidad para el diseño de aplicaciones de Televisión Digital Interactiva, código 3583 de la Vicerrectoría de Investigaciones de la Universidad del Cauca (Colombia).

REFERENCIAS

- [1] T. Hesselmann, “SCIVA - Designing Applications for Surface Computers”. *Engineering interactive computing*, pp. 191–196, 2011.
- [2] C. Remy, “A Pattern Language for Interactive Tabletops in Collaborative Workspaces”. *Building*, vol. 10, pp. 1–47, 2001.
- [3] Microsoft, “Microsoft Surface User Experience Guidelines”. *Text*. Microsoft, pp. 1–77, 2009.
- [4] A. Collins and J. Kay, “Who did what? Who said that?: Collaid: an environment for capturing traces of collaborative learning at the tabletop”. *Proceedings of the ACM*, 2011.
- [5] S. D. Scott, K. D. Grant, and R. L. Mandryk, “System Guidelines for Co-located, Collaborative Work on a Tabletop Display”. *Computer*, vol. p24, no. September, pp. 14–18, 2003.
- [6] Y. Magallanes, “InnovIMM: Modelo de Interfaces para Innovación usando Superficies Multitáctiles”. Universidad de las Américas Puebla, 2011.
- [7] Y. Magallanes, A. Molina y J.A. Sánchez, “Towards an optimal combination of gestures and graphical elements for collaboration using multitouch surfaces”. *Proceedings of the CONIELECOMP*, 2012.

SISTEMA DE MANIPULACIÓN REMOTA DE ROBOTS ARDUINO POR MEDIO DE SERVICIOS WEB SEMÁNTICOS

REMOTE MANIPULATION SYSTEM OF ARDUINO ROBOTS THROUGH SEMANTIC WEB SERVICES

Ph.D. Jaime Guzmán-Luna

*Universidad Nacional de Colombia
Sede Medellín, Colombia
jaguzman@unal.edu.co*

Ing. Juan Meza Gonzáles

*Universidad Nacional de Colombia
Sede Medellín, Colombia
judmeza@unal.edu.co*

Ing. Eimer Castro Hincapié

*Universidad Nacional de Colombia
Sede Medellín, Colombia
ewcastroh@unal.edu.co*

(Artículo de **Investigación científica y tecnológica**. Recibido el 31/10/2011. Aprobado el 19/12/2011)

Resumen. Este artículo presenta la implementación de un sistema basado en servicios web semánticos, orientados hacia la manipulación remota de robots capaces de ejecutar jugadas básicas de fútbol, diseñados y desarrollados con la plataforma de hardware libre Arduino. Dichos servicios fueron descritos semánticamente con el lenguaje ontológico OWL-S y bajo una ontología de dominio común que describe un robot capaz de jugar fútbol.

Palabras clave: Robótica; Robótica en la nube; Servicios web; Web semántica.

Abstract. This paper presents an implementation of a system based on different semantics web services, aimed at manipulating robots capable of executing basic plays soccer, designed and developed using the open hardware platform Arduino. These services were described semantically using ontology language OWL-S and under a common domain ontology to describe a robot that can play soccer.

Keywords: Robotics, cloud robotics, web services, semantic web

1. INTRODUCCIÓN

Hoy se advierte una tendencia mundial para conseguir que los robots, sin importar el tipo, la plataforma o el sistema operativo que usen, estén en capacidad de comunicarse entre sí por medio de mensajes remotos intermediados por servidores [1], que estarían ejecutando servicios web [2] y que permitan desarrollar cualquier tarea propuesta de forma remota.

Uno de los principales aportes del manejo remoto de un robot es que, como existe un servidor o servidores remotos que intervienen en el proceso, es posible disminuir la carga computacional que el robot debe realizar [3], es decir, ya no sería el robot el encargado de los cálculos, análisis y ejecución de algoritmos, sino que parte de estas tareas serían asignadas a los servidores remotos como parte integrada del sistema robótico. Ello facilita el aumento de la capacidad de cómputo del sistema asociado con el robot para incrementar la

complejidad de los procesos y la capacidad de procesamiento por parte de los servidores asociados. Esto genera un nuevo tipo de robot, más económico, capaz de realizar innumerables tareas y para diferentes propósitos, haciéndolos más accesibles para el uso personal [4].

Este artículo se enfoca en mostrar que los servicios web semánticos [5] pueden ser aplicados para manipular remotamente robots construidos con la plataforma Arduino [6] para ejecutar jugadas básicas de fútbol [7]. Dichos servicios fueron descritos semánticamente con el lenguaje OWL-S [8] sobre una ontología de dominio [9] que describe un robot capaz de jugar fútbol. De este modo, se hace posible la invocación de estos servicios con el uso de la API de OWL-S [10].

Este documento está organizado de la siguiente manera: en la sección II se detalla la arquitectura propuesta para el sistema y se muestran las capas que la conforma. En la sección III se explica cómo se realizó la construcción de un robot futbolista con Arduino y sus principales características para ejecutar jugadas básicas de fútbol. En la sección IV se describe la implementación de los servicios web asociados con el sistema robótico, que facilitan la ejecución de jugadas de fútbol de manera remota. En la sección V se caracteriza la ontología de robots asociada con

el sistema robótico. Esta ontología describe un robot con sus características y componentes para jugar fútbol y se hace explícita la forma en la que se realizó la descripción semántica de los servicios. La sección VI presenta cómo se efectúa la invocación de los servicios a partir de su descripción semántica. Finalmente, en la sección VII, se presentan las conclusiones y trabajos futuros de este proyecto.

2. METODOLOGÍA PROPUESTA

Para soportar este trabajo se propone una metodología compuesta por los siguientes pasos: la implementación física del robot, la implementación de los servicios web que manejan remotamente el robot, la creación del marcado semántico de los servicios y la implementación del módulo para la invocación de los servicios. (Ver Fig. 1).



Fig. 1. Metodología propuesta

Como resultado de seguir esta metodología se obtuvo un sistema robótico manejado remotamente en el que el robot se comunica con el servicio web que, a su vez, tiene una descripción semántica que hace uso de una ontología del dominio en asociación con una ontología para la descripción de servicios y, finalmente, con el uso de las anteriores ontologías, se invoca cada servicio por medio de una aplicación orientada hacia el manejo de descripciones semánticas de servicios web.

3. CONSTRUCCIÓN FÍSICA DEL ROBOT FUTBOLISTA CON ARDUINO

El objetivo del robot es que sea capaz de realizar jugadas básicas de fútbol tales como atacar y defender con el uso de la plataforma de Arduino.

El robot fue construido con el ensamble de los siguientes componentes:

- **Placa Arduino mega ADK:** Contiene lo necesario para hacer funcionar el micro controlador; simplemente se conecta al computador con el cable USB o con un transformador o batería para empezar [11].
- **Bases en acrílico:** Soportes necesarios para poner los componentes electrónicos del robot. Se utilizaron 2 bases, una para soportar las ruedas y otra para las tarjetas controladoras del robot.
- **Ruedas omnidireccionales:** Cuando el eje gira, se comportan como una rueda normal. Sin embargo, su arquitectura permite que la rueda también se desplace hacia los lados.
- **Motoreductor con encoder:** Este motoreductor funciona con 12V y tiene una relación de reducción metálica de 50:1. Incluye un *encoder* de cuadratura que provee 64 pulsos por vuelta en el eje del motor y equivale a 3200 pulsos por vueltas en la salida de la caja de reducción.
- **Wireless Shield Proto:** Permite a la placa Arduino comunicarse de forma inalámbrica mediante un módulo inalámbrico.
- **Arduino Motor Shield:** Para manejar dos motores de corriente continua con la placa Arduino, el control de la velocidad y dirección de cada uno de forma independiente.
- **Brújula (compás) HMC6352:** Es un módulo para brújula completamente integrado que combina sensores magnético-resistivos de dos ejes con los circuitos análogos y digitales necesarios y los algoritmos para computación.
- **Dagu Mini Pan and Tilt Kit:** Este kit fue diseñado para hacer montajes de sensores y es adaptable a casi cualquier posición de los sensores que se requiera.
- **Sensor de color HDJD-S822:** Con un filtro de color rojo (R), verde (G) y azul (B) recubiertas por la matriz de fotodiodos, el sensor convierte la luz RGB para salidas analógicas de tensión.
- **Batería lipo KingMax:** La batería tiene dos celdas y la salida es de 7.4V 2200mAh de almacenamiento de carga.
- **Sensores infrarrojo:** Foto-transistores infrarrojos de 940nm NPN 5mm oscuros, ampliados con electrónica (para que su capacidad de detención de la pelota sea mejor y mayor).
- **Polulo DC Motor Driver:** Compatible con ultrasonidos (hasta 20 kHz) de ancho de pulso (PWM) de la tensión de salida del motor que elimina la conmutación audible causada por el control de velocidad.

Con el uso de estos componentes se realizó el ensamblaje del robot y se llevaron a cabo los siguientes pasos:

- **Ensamble de la estructura de soporte de las ruedas:** Se ensambló la base de acrílico que sirve de soporte a las ruedas omnidireccionales y sus respectivos motores, como se detalla en la Fig. 2.



Fig. 2. Ruedas y motores sujetos a la base de acrílico

- **Ensamble de la base para el soporte de los dispositivos de control del robot:** En la segunda base de acrílico se ensambló la tarjeta controladora Arduino mega ADK y, luego, esta base se acopló a la base de las ruedas mediante un juego de tornillos dispuestos para este fin. (Ver Fig. 3).



Fig. 3. Arduino Mega ADK sujeta a la segunda base

- **Montaje en torre de las tarjetería de control Arduino:** Se montaron verticalmente las diferentes tarjetas controladoras asociadas con los periféricos del robot (*Wireless Shield Proto*, *Arduino Motor Shield*, *Brújula (compás) HMC6352*, *Dagu Mini Pan and Tilt Kit*, sensor de color HDJD-S822,

sensores infrarrojo), y se establecieron las conexiones respectivas con la tarjeta controladora mega ADK, previamente instalada. (Ver Fig. 4).

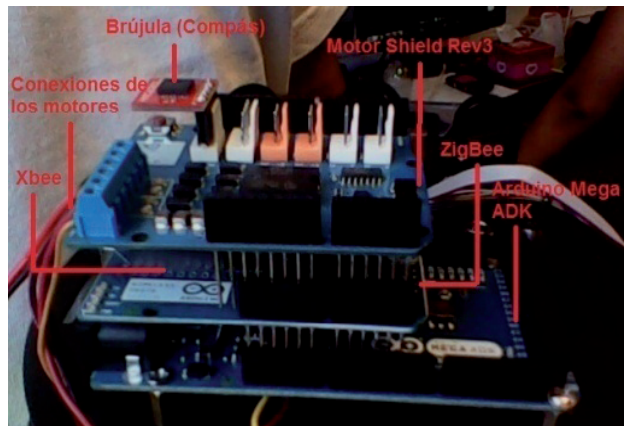


Fig. 4. Montaje en torre de la tarjetería restante

- **Ensamble del Dagu Mini Pan and Tilt Kit:** Se realizó el ensamble del manipulador que facilita la implementación del “brazo” para patear el balón de fútbol, que está montado en la parte delantera de la base que soporta las ruedas, como se muestra en la Fig. 5.



Fig. 5. Ensamble de la Dagu Mini en la parte delantera

Cuando esté listo el ensamblaje y la conexión de estos componentes, se obtiene el robot omnidireccional construido con la plataforma Arduino, capaz de ejecutar jugadas básicas de fútbol, como se advierte en la Fig. 6.

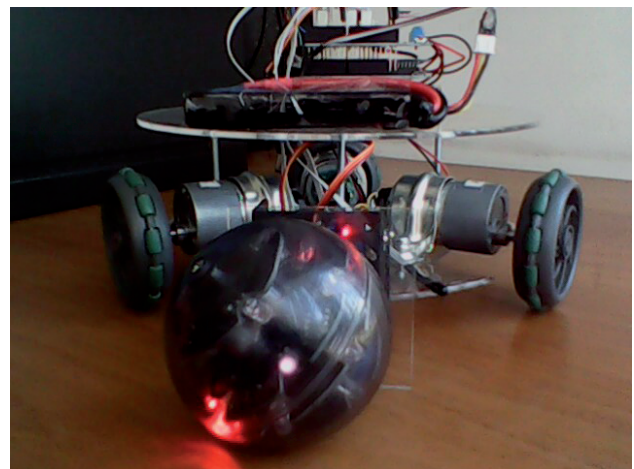


Fig. 6. Robot Arduino construido

4. CREACIÓN DE SERVICIOS WEB

Un servicio web es una aplicación ejecutable en un servidor y es accesible por medio de protocolos de comunicación, independientes de la plataforma o sistema operativo de quien realiza la comunicación ni del servidor en el que se encuentra alojado el servicio [15].

Si un servicio web es capaz de establecer una comunicación con un robot y enviarle a éste una acción para ejecutar, es posible conseguir que un robot sea controlado de manera remota. Los servicios web son un soporte básico de un área actual de investigación llamada robótica en la nube [16], en la que se desarrollan sistemas computacionales que permiten tener robots accesibles desde la web.

Para ver cómo apoyan los servicios web a la robótica y tener un primer acercamiento al concepto de la robótica en la nube, se decidió implementar tres servicios web para que hicieran tres acciones específicas de un robot jugador de fútbol: buscar la pelota, patear la pelota y pasar la pelota. Para estas acciones se implementaron los servicios de tal manera que reciben como parámetro la descripción de un robot y se basa en las habilidades, dispositivos y acciones del robot. El servicio determina si está en capacidad de llevar a cabo la acción solicitada. En caso de ser aprobada la solicitud, se realizan los cálculos pertinentes en el propio servidor y éste interactúa remotamente con el robot para llevar a cabo físicamente la acción. A continuación se describen los tres servicios relacionados:

- **El servicio BuscarPelota:** Este servicio determina si entre la descripción asociada con el robot se especifica que éste contiene una tarjeta de comunicación y si entre sus habilidades se encuentra la de buscar la pelota. En la figura 7 se muestra parte del código que realiza la esta tarea.

```

OWLKnowledgeBase kb = OWLFactory.createKB();
String[] reg = robot.split("<");
int inicio = reg[2].indexOf("\n") + 1;
int fin = reg[2].lastIndexOf("\n");
String uri = reg[2].substring(inicio, fin);
URI url = URI.create("http://localhost:8080/Trabajo 2 2 IW/buscarPelota.owl");
Service aService = kb.readService(url);
OWLIndividual individual = kb.getIndividual(URI.create(uri));
Object[] owl = individual.getProperties().values().toArray();
String salida = "";
for(int i = 0; i < owl.length; i++)
{
    salida += owl[i].toString().toLowerCase();
}
String respuesta = "El robot no es capaz de jugar futbol o buscar la pelota";
if(salida.contains("ruedas") && salida.contains("futbol") && salida.contains("x-bee") &
&& salida.contains("luz") && salida.contains("ultrasonido") && salida.contains("buscar")) {
    respuesta = "El robot puede buscar la pelota";
}
return respuesta;

```

Fig. 7. Servicio web, BuscarPelota

- **El servicio PatearPelota:** Mediante este servicio se determina si en la descripción asociada con el robot se especifica que éste contiene una tarjeta de comunicación y si por lo menos una de sus habilidades es la de patear. En la Fig. 8 se muestran apartes del código que realiza la tarea.

```

OWLKnowledgeBase kb = OWLFactory.createKB();
String[] reg = robot.split("<");
int inicio = reg[2].indexOf("\n") + 1;
int fin = reg[2].lastIndexOf("\n");
String uri = reg[2].substring(inicio, fin);
URI url = URI.create("http://localhost:8080/Trabajo 2 2 IW/patearPelota.owl");
Service aService = kb.readService(url);
OWLIndividual individual = kb.getIndividual(URI.create(uri));
Object[] owl = individual.getProperties().values().toArray();
String salida = "";
for(int i = 0; i < owl.length; i++)
{
    salida += owl[i].toString().toLowerCase();
}
String respuesta = "El robot no es capaz de jugar futbol o patear la pelota";
if(salida.contains("ruedas") && salida.contains("futbol") && salida.contains("x-bee") &
&& salida.contains("luz") && salida.contains("ultrasonido") && salida.contains("ataque")) {
    respuesta = "El robot puede patear la pelota";
}
return respuesta;

```

Fig. 8. Servicio web, PatearPelota

- **El servicio PasarPelota:** Este servicio define si el robot está en capacidad de pasar la pelota, por medio de un análisis de la descripción recibida, se verifica que entre sus habilidades se encuentre pasar la pelota y que, además, sea capaz de comunicarse por medio de una tarjeta. (Ver Fig. 9).

```

OWLKnowledgeBase kb = OWLFactory.createKB();
String[] reg = robot.split("<");
int inicio = reg[2].indexOf("\n") + 1;
int fin = reg[2].lastIndexOf("\n");
String uri = reg[2].substring(inicio, fin);
URI url = URI.create("http://localhost:8080/Trabajo 2 2 IW/pasarPelota.owl");
Service aService = kb.readService(url);
OWLIndividual individual = kb.getIndividual(URI.create(uri));
Object[] owl = individual.getProperties().values().toArray();
String salida = "";
for(int i = 0; i < owl.length; i++)
{
    salida += owl[i].toString().toLowerCase();
}
OWLIndividual individual2 = kb.getIndividual(URI.create("http://localhost:8080/Trabajo
2 2 IW/ dominio.owl#Arduino"));
String respuesta = "El robot no es capaz de jugar futbol o pasar la pelota";
if(salida.contains("ruedas") && salida.contains("futbol") && salida.contains("x-bee") &
&& salida.contains("luz") && salida.contains("ultrasonido") && salida.contains("pase")) {
    respuesta = individual2.toRDF(true, true);
}
return respuesta;

```

Fig. 9. Servicio web, PasarPelota

La implementación de estos servicios dejó entrever que es posible manejar remotamente un robot. Para ello, a cada servicio se le implementó inicialmente una comunicación mediante cable y luego, con una comunicación inalámbrica, se hizo uso de tarjetas XBee [18] entre el servidor web y el robot. Con ello, los procesos principales de cada jugada se realizan en el servidor y la programación básica asociada con el sistema operativo relacionada con movimientos y sentido, se efectúa en el robot.

5. DESCRIPCIÓN SEMÁNTICA DE LOS SERVICIOS

Con el fin de que las máquinas entendieran inequívocamente lo que hace cada servicio, se optó que para cada uno de los servicios implementados en la sección anterior se realizara su respectiva descripción semántica con el uso de ontologías. “Una ontología es una especificación explícita de una conceptualización”, según Thomas Gruber [12]. A continuación se describen los principales elementos que componen esta descripción semántica.

5.1 Creación de la ontología del dominio

Con el uso de lo que es una ontología, en esta sección se muestra que en el marco de este trabajo se realizó una representación del conocimiento asociado con el robot y el dominio de jugar fútbol. Como marco de referencia en la construcción de la ontología se asumió que un robot puede verse como una entidad a la que se le puede asociar un conjunto de habilidades para que ejecute tareas específicas como jugar fútbol al igual que se le asocia un conjunto de componentes mecánicos y electrónicos para que el robot lleve a cabo acciones de sentido, movimiento y planeación, entre otras. [13].

Para la construcción de la ontología se usó el editor de ontologías Protégé [14]. En la ontología se definió un robot como un objeto conformado por dispositivos, capaz de ejecutar diversas acciones, caracterizado por ciertas habilidades y de movilizarse según un sistema de locomoción. Los principales conceptos y las relaciones que conforman esta ontología son:

- **Los dispositivos:** Dentro de la ontología se definieron tres principales dispositivos (comunicación, energía y sentido). Cada uno de estos dispositivos contiene, a su vez, componentes más específicos para un robot, como se aprecia en la Fig.10.

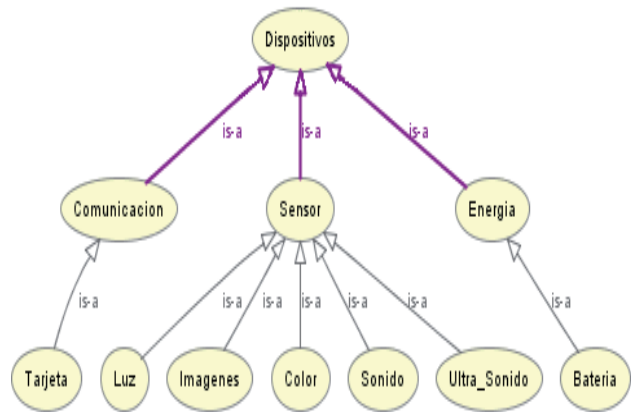


Fig. 10. Los dispositivos en la ontología

- **Las acciones:** Se definieron dos conjuntos básicos de acciones relacionadas con un robot capaz de jugar fútbol, como son las acciones de sentido y las acciones propias de las jugadas de fútbol que puede realizar el robot. Estas acciones se muestran en la Fig. 11.

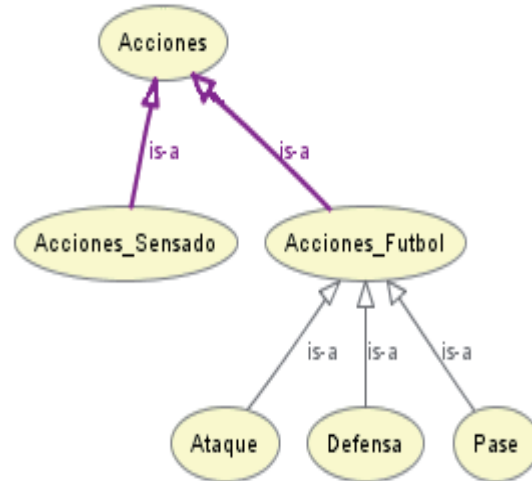


Fig. 11. Las acciones en la ontología

- **Las habilidades:** En la ontología asociada con el robot jugador de fútbol se definió el concepto de habilidades con el que se busca definir el concepto que reúne a aquellos elementos que ayuden al robot a llevar a cabo, con facilidad, cierta tarea. Para el robot futbolista se consideraron dos habilidades: la de juego y la de interacción con otros robots. En la Fig. 12 se muestran los conceptos y sus relaciones asociados con las habilidades del robot en la ontología.

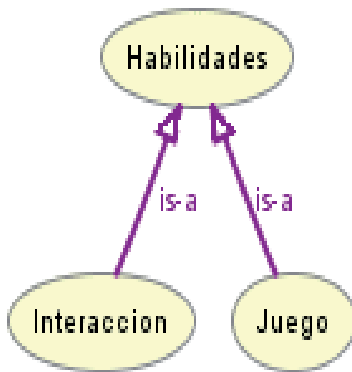


Fig. 12. Las habilidades en la ontología

Finalmente, una visión general de la ontología se detalla en la Fig. 13.



Fig. 13. Ontología de un robot jugador de fútbol

5.2 Creación de la descripción semántica de los servicios

Para la descripción semántica de los servicios web se usó la ontología OWL-S [8]. Su implementación se hizo con el Plugin de OWL-S para Protégé [17]. Así, para cada servicio se definió su respectiva instancia OWL-S que hace uso de la ontología del do-

minio descrita en la sub-sección anterior con el fin de manejar los parámetros de entrada y salida de cada servicio. En la Fig. 14 se muestran apartes de la descripción semántica del servicio BuscarPelota en el que se observa la integración del OWL-S y la ontología del dominio asociada con las entradas de este servicio. (Ver el concepto *hasInput* resaltado en la Fig. 14). Tal como se aprecia en la figura, el servicio recibe como entrada el tipo de dato complejo Robot, que corresponde a la clase Robot_Móvil definida en la ontología del dominio.

```

<service:presents>
  <profile:Profile rdf:ID="BuscarPelotaProfile">
    <profile:hasInput>
      <process:Input rdf:ID="Robots">
        <process:parameterType rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI"
          >http://localhost:8080/Trabajo_2_2_IW/dominio.owl#Robot_Movil</process:parameterType>
      </process:Input>
    </profile:hasInput>
    <service:presentedBy rdf:resource="#BuscarPelotaService"/>
    <profile:textDescription rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
      >Busca la pelota</profile:textDescription>
    <profile:hasOutput rdf:resource="#Salida"/>
    <profile:serviceName rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
      >Buscar Pelota</profile:serviceName>
  </profile:Profile>
</service:presents>
  
```

Fig. 14. Descripción semántica de los servicios

6. INVOCACIÓN DE LOS SERVICIOS SEMÁNTICOS

Por tener una descripción semántica de los servicios, su invocación se puede realizar por medio de la API de OWL-S [10]. Esta API, permite crear una base de conocimiento soportada por las instancias de las ontologías de los servicios web y del dominio de fútbol que manejan. Esta API, basada en la descripción semántica de cada servicio web, permite obtener una instancia del servicio y del proceso asociado con éste, se definen las entradas (complejas o primitivas), las salidas requeridas y la ejecución del proceso asociado con cada servicio. En la Fig. 15 se muestran apartes del código en Java que, cuando se ejecuta mediante la API OWL-S, hace la invocación del servicio BuscarPelota.

```

public class BuscarPelotaInvocacion
{
    public static void main(String[] args) throws IOException, ExecutionException
    {
        ProcessExecutionEngine exec = OWLFactory.createExecutionEngine();
        OWLKnowledgeBase kb = OWLFactory.createKB();
        URI url = URI.create("http://localhost:8080/Trabajo 2 2 IW/buscarPelota.owl");
        Service aService = kb.readService(url);
        Process aProcess = aService.getProcess();
        ValueMap<Input, OWLValue> inputs = new ValueMap<Input, OWLValue>();
        inputs.setValue(aProcess.getInput(), kb.getIndividual(URI.create("http://localhost:8080/Trabajo 2 2 IW/dominio.owl#Arduino")));
        ValueMap<Output, OWLValue> outputs = exec.execute(aProcess, inputs, kb);
        OWLDataValue out = outputs.getDataValue(aProcess.getOutput());
        System.out.println(out.getValue());
    }
}

```

Fig. 15. Invocación del servicio con OWL-S API

8. CONCLUSIONES Y TRABAJO FUTURO

Con la implementación del prototipo del sistema de manipulación remota de robots por medio de servicios web semánticos, los usuarios, tanto humanos como las máquinas, pueden tener acceso al robot desde cualquier parte, siempre y cuando se tenga un punto de acceso a la web. Además, como se contempla una descripción semántica de los servicios que manipulan el robot, es posible realizar tareas de manera automática ya que las máquinas entienden la funcionalidad de los servicios que abren opciones interesantes como el descubrimiento automático de tareas que hace el robot al igual que componer automáticamente nuevas tareas que combinen los servicios asociados con las tareas que hace el robot.

Se encontró que existe la API OWL-S, que es una herramienta adecuada para invocar fácilmente servicios web marcados semánticamente. Esta API presenta un acoplamiento coherente con las descripciones de un dominio hechas en OWL y con las descripciones de servicios web con la ontología OWL-S. Su programación se hace en Java y se puede editar, invocar y crear servicios web desde cero.

El uso de servicios web para implementar en ellos los procesos complejos asociados con las tareas que debe realizar un sistema robótico y deja la implemen-

tación de pequeñas tareas básicas en el propio hardware del robot, permite que se disminuya el costo de los robots porque no se requeriría un hardware complejo en el propio robot.

Por último, la metodología propuesta en este trabajo es una muestra de cómo podría ser el proceso de construcción basado en una secuencia de capas que involucre la manipulación remota de un robot cualquiera.

Como trabajo futuro se propone hacer, de manera experimental, pruebas que permitan evaluar la complejidad y la eficiencia de éstos a clase de sistemas robóticos basados en servicios web semánticos orientados hacia el control remoto de los robots.

Otro trabajo futuro que se tiene proyectado es la implementación de un sistema capaz de componer tareas robóticas complejas a partir de tareas descritas mediante los servicios web semánticos. Para ello, se piensa utilizar algunas herramientas basadas en técnicas de planificación de inteligencia artificial adaptadas en la composición de servicios web.

Como última línea de trabajo futuro se plantea trabajar en mejorar nuevas técnicas de comunicación remota entre el servidor y el propio robot. Para ello, se considera estudiar nuevas tecnologías de comunicación inalámbrica como WI-FI y hacer uso de redes de telefonía celular para tener un mayor alcance entre el servidor y el robot.

AGRADECIMIENTOS

Este trabajo es apoyado por el proyecto “Programa de Fortalecimiento del Grupo de Investigación Sistemas Inteligentes Web – Sintelweb, convocatoria nacional 2010-2012, Modalidad 3”, con código 20201009532.

REFERENCIAS

- [1] Lafuente, A. Larrea, M, “Sistemas Distribuidos. Introducción”, Dpto. ATC UP/EHU, Jul. 10. 2009, Universidad del País Vasco. [en línea], Disponible en: <http://www.sc.ehu.es/acwlaalm/sdi/introduccioi-slides.pdf>. 2009.
- [2] Cavanaugh, E, “Web services: Benefits, challenges, and a unique, visual development solution”, white paper, Feb. 10. 2006, Sitio web de Altova. [en línea], Disponible en: <http://www.altova.com/whitepapers/webservice.pdf>. 2006.

- [3] Paul d. Hestand, a service oriented architecture for robotic platforms, university of massachusetts lowell, 2011
- [4] Willow Garage, "Personal Robot 2 (PR2)," [en línea], Disponible en: www.willowgarage.com.
- [5] Vaculin, R. Sycara, K, "Semantic Web Services Monitoring: AN OWL-S based Approach", 2007.
- [6] Margolis A, Arduino Cookbook, O'REILLY, 2011.
- [7] Matamoros, J M. López, J I, "Comportamiento adaptable de un equipo de fútbol de Robots Humanoides", 2011.
- [8] Martin D, Paolucci M, McIlraith S, Burstein M, McDenitt D, McGuinness D, Parsia B, Payne T, Sabou M, Solanki M, Srinivasan N & Sycara K. "Bringing Semantic to Web Services: The OLW-S Approach", 2004.
- [9] Boyce, S., & Pahl, C. Developing Domain Ontologies for Course Content. *Educational Technology & Society*, 10 (3), 275-288. 2007.
- [10] Sirin E & Parsia B. "The OWL-S Java API", 2004.
- [11] Arduino Mega ADK, Sitio web de Arduino [en línea], Disponible en: <http://arduino.cc/en/Main/ArduinoBoardADK>
- [12] Gruber, T, "What is an Ontology", 2007.
- [13] Schlenoff, C. Messina, E, "A robot Ontology for Urban Search and Rescue", 2006.
- [14] Noy. N F, McGuinness, D L, "Ontology Development 101: A Guide to Creating Your First Ontology", 2002.
- [15] Gunzer H, "Introduction to Web Services", 2002
- [16] Hu, G. Tay, W P. Wen Y, "Cloud Robotics: Architecture, Challenges and Applications", 2011.
- [17] Elenius D, Denker G, Martin D, Gilham F, Khouri J, Sadaati S & Senanayake R. "The OWL-S Editor A Development Tool for Semantic Web Services", 2005.
- [18] Mayalarp V, Limpaswadpaisarn N, Poombansao T & Kittipiyakul S. "Wireless Mesh Networking with XBee", 2010.
- [19] A. Elçi and B. Rahnama, "Human-Robot Interactive Communication Using Semantic Web Tech. in Design and Implementation of Collaboratively Working Robots", in Proc. ROMAN, pp.273-278. 2007.
- [20] Mortiz, T. Ulrich, K. Dejan, P & Micheale, B., "Web-enabled Robots that use the Web as an Information Resource", 2011.
- [21] Arkin, R., "Motor schema-based mobile robot navigation", In Proceedings of the IEEE International Conference on Robotics and Automation, pp.264-271. 2007.
- [22] Hogan, A. Harth, A. y Polleres, A., "Scalable authoritative OWL reasoning for the web", *Information Systems*, pp.49-90. 2008.
- [23] Gennari, J. Musen, M. Ferguson, R. Grosso, W. Crubézy, M. Eriksson, H. Noy, N. y Tu, S., "The evolution of Protégé: an environment for knowledge-based systems development", *International Journal of Human-Computer Studies*, pp.89 - 123. 2003.

GUÍA PARA AUTORES

Lámpsakos

La revista recibirá aportes teóricos en las áreas de Ingeniería. Para ello dispone de las siguientes normas de presentación:

1. Tipos de artículos que se reciben:

- Artículos de investigación: presenta de manera detallada los resultados originales de investigaciones terminadas. La estructura utilizada contiene cuatro apartes: Introducción, metodología, resultados y conclusiones.
- Artículo de reflexión: presenta resultados de investigación terminada desde una perspectiva analítica, interpretativa o crítica del autor, sobre un tema específico, recurriendo a fuentes originales.
- Artículo de revisión: sistematiza e integra resultados de una investigación terminada, publicada o no publicada, sobre un campo o ciencia, para dar cuenta de avances y tendencias de desarrollo.

Presenta una cuidadosa revisión bibliográfica de por lo menos cincuenta referencias.

- Artículo corto: presenta resultados preliminares o parciales de una investigación.
- Revisión de tema: presenta la revisión crítica de la literatura sobre un tema en particular.

2. Formato general del texto, disponible en (<http://www.funlam.edu.co/lampsakos/paginas/normas.html>):

- El texto deberá ser presentado en Microsoft Word, en letra Trebuchet MS, tamaño 10. La configuración de página tendrá un tamaño A4 de 21.0 x 29.7cm, con márgenes superior, inferior, izquierdo y derecho de 2cm. Deberá ser a dos columnas con separación central de 0.5cm. El interlineado: sencillo.
- Extensión: el número de páginas que debe contener el artículo es de mínimo 5 y máximo 8, incluyendo cuadros, gráficas, bibliografía y notas.
- El título del texto debe estar en idioma español e inglés.
- Reseña del autor: quien sea el autor del texto deberá redactar, en un párrafo que no supere las 10 líneas, un breve resumen de su hoja de vida, que incluya: su nombre y apellidos como los registra para efectos de autoría; el último grado de esco-

laridad y la Universidad en la que lo obtuvo; filiación institucional y cargo que desempeña; si hace parte de algún grupo de investigación. Deberá también indicar el país donde se asocia su filiación y el correo electrónico (institucional).

- Resumen analítico: Los artículos deberán contener un resumen o abstract en los idiomas español e inglés, con una extensión de máximo 200 palabras, que sintetice la idea central, los objetivos, la metodología y las conclusiones.
- Palabras clave: deberán relacionarse cinco palabras clave en los idiomas español e inglés.
- Secuencia del artículo: la estructura del artículo deberá ser:

- Título.
- Nombre del autor.
- Grado académico: indicar, con abreviatura, el último título académico obtenido (Ph.D., Mg., Esp.)
- Resumen en el idioma original del artículo.
- Palabras en el idioma original del artículo.
- Resumen en otro idioma (inglés o español).
- Palabras clave en otro idioma (inglés o español).
- Introducción: incluirá la importancia y los alcances del tema tratado, así como los objetivos e hipótesis del mismo.
- Desarrollo y resultados: se presenta el desarrollo del tema. Los resultados obtenidos deberán exponerse en forma clara y concisa (conclusiones).
- Referencias: todas las obras citadas en el texto deberán estar en las referencias y viceversa, siguiendo las normas APA. No se recibirán las obras que no estén en este tipo de citación.

3. Criterios de citación.

- La citación deberá hacerse conforme a lo indicado por el sistema el formato habitual para las referencias empleado en las revistas del IEEE en inglés con objeto de facilitar las referencias cruzadas con el Xplore.

EJEMPLOS:

- [1] J. F. Fuller, E. F. Fuchs, & K. J. Roesler. "Influence of harmonics on power distribution system protection". IEEE Trans. Power Delivery, Vol. 3, No. 2, pp. 549-557, Apr. 1988.
- [2] E. Clarke. "Circuit Analysis of AC Power Systems" Vol. I. New York: Wiley, 81 p. 1950.
- [3] IEEE Guide for Application of Power Apparatus Bushings, IEEE Standard C57.19.100-1995, Aug. 1995.
- Las citas deben transcribirse en el idioma original en el que fueron escritas.
 - Las notas de pie de página se emplearán sólo para hacer aclaraciones o aportar datos adicionales. No deben emplearse para referencias bibliográficas.
- ### 4. Lista de referencias
- Se ubican en un título sin numeración y en continuidad con el texto.
 - Las referencias deben estar numeradas en el orden en que aparecen en el documento.
 - Cada número de referencia debe estar entre corchetes [] en la línea de texto.
 - Las obras de un mismo autor se ordenan cronológicamente comenzando por la más antigua.
- ### 5. Proceso de evaluación de las publicaciones
- El autor(es) del texto deberá enviar, vía correo electrónico y junto al artículo que desea someter a la revista, dos cartas dirigidas al Director de la revista (formatos disponibles (<http://www.funlam.edu.co/lampsakos/paginas/normas.html>)):
 - Una carta de sesión de derechos, en la que garantice la originalidad de su artículo e indique que éste no ha sido publicado en ningún otro medio de difusión y que no se encuentra postulado simultáneamente para otra u otras revistas.
 - Una carta de presentación. En esta carta se presenta el artículo al Comité Editorial para comenzar el proceso de evaluación.
 - El Director de la revista dará respuesta de confirmación de la recepción del texto en un máximo de tres días y remitirá los documentos al Consejo Editorial. Este ente colegiado realizará una evaluación preliminar del texto recibido, y determinará la pertinencia de la publicación. Una vez verificada, se someterá a evaluación por parte de dos árbitros anónimos, quienes establecerán si el material enviado es publicable.
 - En el evento en que un árbitro apruebe y el otro rechace el artículo, se decidirá enviar el artículo al autor con los resultados de ambos evalua-

dores, a fin de que el propio autor atienda todas las observaciones citadas en el documento de evaluación. Una vez hecho esto, el autor puede remitir nuevamente a la revista su artículo para presentarlo en la siguiente convocatoria (si mantiene su interés en la revista).

- El autor o autores a los cuales les hayan sido aceptados sus artículos, se obligarán a transferir de manera exclusiva todos los derechos sobre la reproducción digital del material (incluidas las fotografías y otros soportes) en favor de la revista Lámpsakos. Así mismo, se comprometerán a que una vez enviada la carta de aprobación del artículo, no lo retirarán.
- Una vez aprobado el artículo, este deberá pasar por un proceso de revisión gramatical a cargo de un experto en el oficio. El objetivo de esta etapa, es mejorar la calidad de la publicación; razón por la cual se remitirá nuevamente el artículo al autor, con las observaciones y sugerencias, formuladas por el corrector; el autor contará con un plazo de tres días para emitir un pronunciamiento relacionado con las sugerencias emitidas por el corrector. En caso de no recibir una respuesta en el plazo convenido, se atenderán las sugere-

ncias como aceptadas por el autor.

- La revista Lámpsakos notificará a los autores sobre el estado de los artículos. Advierte además que en ningún caso estará obligada a devolver los documentos que haya recibido impresos o en medios digitales.

6. Información para el envío

Los artículos y demás documentos se recibirán por correo electrónico en:

Correo electrónico: lampsakos@funlam.edu.co

7. Datos de contacto:

Fundación Universitaria Luis Amigó
 Director Revista Ingrid-Durley Torres
 Transversal 51A #67 B 90. Bloque 1, oficina 310
 Medellín - Colombia

Teléfonos: (57) (4) + 448 76 66 – ext. 9602

Fax (57) (4) + 384 97 97

Nombre del Director de la revista: Ingrid-Durley Torres.

GUIDE FOR AUTHORS

Lámpsakos

The journal will admit theoretical work from all engineering areas. All submitted material must follow these guidelines:

1. Types of articles:

- a. Research articles that include information regarding original results of research projects that have concluded. The article must be developed in terms of: introduction, methodology, results and conclusions.
- b. Reflexive articles that present research results of projects that have already concluded from an analytic, interpretative or critical perspective on a specific topic based on original sources.
- c. Literature review article that systematizes and includes results from a concluded, published or unpublished, research project on a specific field or science in order to show advances and development tendencies.
- d. Short articles that present preliminary or partial results of a research project.
- e. Topic review articles that present a critical revision of the literature for a specific topic.

2. General text format (available at: <http://www.funlam.edu.co/lampsakos/paginas/normas.html>):

- The text should be presented in Microsoft Word, A4 size 21.0x29.7, using proportional margins of 2cm. Spacing must be single and presented in Trebuchet MS 10. It must be presented in two columns with a central separation of 0.5cm.
- The minimum length is 5 pages and the maximum 8 pages per article, including tables, bibliography and notes.
- The title must be presented in both English and Spanish
- Author information: the author must write a 10-line paragraph about his/her CV including name, last name for authorship, highest degree and institution, institutional affiliation and position; research group. Also include country where the institutional affiliation is and (institutional) e-mail.
- Analytical summary: articles must include a summary or abstract with a maximum of 200 words in English and in Spanish pointing out the main idea, objectives, methodology and conclusions.
- Key words: five key words must be established in English and in Spanish.
- Order of the article: the structure of the article must be:
 - Title
 - Author
 - Degree: use the appropriate abbreviation of the highest academic degree (Ph.D., Mg., Esp.)
 - Summary in the article's original language.

- Summary in other languages (English or Spanish).
- okey words in other languages (English or Spanish)
- Introduction: includes the importance and impact of the topic, as well as the objectives and hypotheses.
- Development and results: it deals with the development of the topic. Results should be presented clearly and precisely (conclusions).
- References: all work cited must be in the references and vice versa, using the APA system. Work that does not use this reference system will not be accepted.

3. Citation criteria.

- Citation must be done following the reference format of the IEEE journals in English in order to facilitate the cross reference with Xplore.

EXAMPLES:

- [1] J. F. Fuller, E. F. Fuchs, & K. J. Roesler. "Influence of harmonics on power distribution system protection". IEEE Trans. Power Delivery, Vol. 3, No. 2, pp. 549-557, Apr. 1988.
- [2] E. Clarke. "Circuit Analysis of AC Power Systems" Vol. I. New York: Wiley, 81 p. 1950.
- [3] IEEE Guide for Application of Power Apparatus Bushings, IEEE Standard C57.19.100-1995, Aug. 1995.

- Citation must be transcribed in the original language.
- Footnotes should only be used for further explanations or additional data; must not be used for bibliographic references.

4. Reference list

- The reference list must be located under a title without numeration and continuously in the text.
- References must be numbered in the order that they appear in the text.
- Each reference number must use brackets [] in the text.
- Work by the same author should follow a chronological order.

5. Review process

- The author must submit via e-mail two letters directed to the Editor (formats available at: <http://www.funlam.edu.co/lampsakos/paginas/normas>).

html)

- A letter regarding authorship rights and confirming the originality of the material and that it is not currently under review by another Journal.
- A letter of presentation. This letter presents the article to the Editorial committee to start the process of evaluation.

- The Editor will acknowledge receipt of three days and send it to editorial board members for blind review. This board will make an initial assessment in order to determine the relevance of the article. Once it has been selected it will be submitted to two experts for a blind review. They will decide if the material should be published.
- In case the two experts have different stands regarding the material, the material will be returned to the author(s) with the observation for the analysis of the assessment. The author can once again present the article for the next issue.
- The articles that are accepted will undergo a grammatical review process by an expert. The purpose of this process is to improve the quality of the publication; the corrections and suggestion will then be sent to the author; the author will have three days to make a statement about it. In case the editor does not receive such statement, it will be taken as an acceptance on behalf of the author.
- The Journal will be in constant communication with the authors regarding the process of the article. The original material will not be returned under any circumstances.

6. Submitting information

Material can be submitted electronically to Email: lampsakos@funlam.edu.co

7. Contact information

Fundación Universitaria Luis Amigó
Ingrid-Durley Torres, Editor
Transversal 51A #67 B 90. Bloque 1, oficina 310
Medellín - Colombia
Phone number: (57) (4)+ 4487666 ext. 9602
Fax (57) (4) + 3849797

Editor: Ingrid-Durley Torres

ÉTICA DE LA PUBLICACIÓN

- El Comité Editorial en virtud de la transparencia en los procesos, velará por la confidencialidad de la información que se recibe y la calidad académica de la revista.
- Se consideran causales de rechazo: el plagio, adulteración, invento o falsificación de datos del contenido y del autor, que no sean inéditos y originales.
- En ningún caso, la dirección de la revista exigirá al autor la citación de la misma ni publicará artículos con conflicto de intereses.
- Si una vez publicado el artículo: 1) el autor descubre errores de fondo que atenten contra la calidad o científicidad, podrá solicitar su retiro o corrección. 2) Si un tercero detecta el error, es obligación del autor retractarse de inmediato y se procederá al retiro o corrección pública.