

Advances in Computer Science and Artificial Intelligence

Research in Computing Science

Series Editorial Board

Editors-in-Chief:

Grigori Sidorov (Mexico)
Gerhard Ritter (USA)
Jean Serra (France)
Ulises Cortés (Spain)

Associate Editors:

Jesús Angulo (France)
Jihad El-Sana (Israel)
Alexander Gelbukh (Mexico)
Ioannis Kakadiaris (USA)
Petros Maragos (Greece)
Julian Padget (UK)
Mateo Valero (Spain)

Editorial Coordination:

Alejandra Ramos Porras

Research in Computing Science es una publicación trimestral, de circulación internacional, editada por el Centro de Investigación en Computación del IPN, para dar a conocer los avances de investigación científica y desarrollo tecnológico de la comunidad científica internacional. **Volumen 120**, noviembre 2016. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No. : 04-2005-121611550100-102, expedido por el Instituto Nacional de Derecho de Autor. *Certificado de Licitud de Título* No. 12897, *Certificado de licitud de Contenido* No. 10470, expedidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones, Tres Guerras 27, Centro Histórico, México, D.F. Distribuida por el Centro de Investigación en Computación, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, México, D.F. Tel. 57 29 60 00, ext. 56571.

Editor responsable: *Grigori Sidorov, RFC SIGR651028L69*

Research in Computing Science is published by the Center for Computing Research of IPN. **Volume 120**, November 2016. Printing 500. The authors are responsible for the contents of their articles. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of Centre for Computing Research. Printed in Mexico City, in the IPN Graphic Workshop – Publication Office.

Advances in Computer Science and Artificial Intelligence

Juan Carlos Chimal Eguía
Paula Denisse Concha Gómez (eds.)



Instituto Politécnico Nacional
"La Técnica al Servicio de la Patria"



Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2016

ISSN: 1870-4069

Copyright © Instituto Politécnico Nacional 2016

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional “Adolfo López Mateos”, Zacatenco
07738, México D.F., México

<http://www.rcs.cic.ipn.mx>

<http://www.ipn.mx>

<http://www.cic.ipn.mx>

The editors and the publisher of this journal have made their best effort in preparing this special issue, but make no warranty of any kind, expressed or implied, with regard to the information contained in this volume.

All rights reserved. No part of this publication may be reproduced, stored on a retrieval system or transmitted, in any form or by any means, including electronic, mechanical, photocopying, recording, or otherwise, without prior permission of the Instituto Politécnico Nacional, except for personal or classroom use provided that copies bear the full citation notice provided on the first page of each paper.

Indexed in LATINDEX, DBLP and Periodica

Printing: 500

Printed in Mexico

Editorial

The objective of this volume is to present the recent advances in the chosen areas of Computer Science. The works included in this volume were carefully curated by the editors, through blind review process, on the basis of originality and technical quality. This issue of the journal *Research in Computing Science* will be useful for the researchers and the students working in various areas of Computer Science, as well as for all the readers interested in the enrichment of their knowledge in this field.

This volume caters to the advancements of the following areas of computer science: Artificial Intelligence, Digital Signal Processing, Intelligent Systems for Automation, Simulation and Modeling, Natural Language Processing, Data Science and Software Technology, Networks and Data Science, Micro Technologies and Embedded systems, Intelligent Processing of Geospatial Information, Intelligent Computing, Robotics and Mechatronics, and Cyber-security.

All the submitted papers were reviewed by two or more independent members of the editorial committee. In general, the acceptance rate was 55.56%. This volume contains revised and corrected versions of the 15 accepted papers.

We would like to express our gratitude towards all the people who contributed to this volume. Firstly, the authors of the papers for their technical excellence, which guaranteed the quality of this publication. We also want to thank the members of the editorial board for their hard work in evaluation and selection of the best papers among the many submissions that we received. We sincerely express our gratitude to the *Sociedad Mexicana de Inteligencia Artificial* (SMIA) for their collaboration in the preparation of this publication. Also, we want to give special recognition to *Centro de Investigación en Computation of Instituto Politecnico Nacional* (CIC-IPN) for their support in the publication of this volume. The submission, review, and selection process were facilitated for free by the *EasyChair*, www.EasyChair.org.

Juan Carlos Chimal Eguía
Paula Denisse Concha Gómez
Guest Editors

November 2016

Table of Contents

| | Page |
|--|------|
| Estudio del análisis de componentes principales en bases de datos de calidad del aire | 9 |
| <i>Abraham Sánchez López, Viridiana Cruz-Gutiérrez, Mario Alberto Posada-Zamora, M. Teresa Torrijos M., M. Auxilio Osorio Lama</i> | |
| Un enfoque de control de movimiento de un helicóptero con cuatro rotores | 21 |
| <i>H. Yáñez-Badillo, R. Tapia-Olvera, F. Beltran-Carbajal, O. Aguilar-Mejía, I. López-García</i> | |
| Comparación cuantitativa de matrices de datos | 31 |
| <i>Angel Mandujano-García, Jesús Figueroa-Nazuno, Hiram Calvo</i> | |
| Algoritmos del muestreo/reconstrucción para realizaciones de procesos Gaussianos con espectros rectangulares | 41 |
| <i>Vladimir Kazakov, Francisco Mendoza Sánchez</i> | |
| White-donkey: búsqueda de personas con vehículos aéreos no tripulados basada en visión por computadora | 53 |
| <i>Jesús Jaime Moreno Escobar, Edgar Antonio Domínguez Ramírez, Jesús Cruz Hernández, Oswaldo Morales Matamoros, Ricardo Tejeida Padilla</i> | |
| Un algoritmo multi-objetivo NSGA-II para la programación de los pasos de laminación en caliente de aceros | 65 |
| <i>Carlos A. Hernández, Guadalupe Castilla, Alejandro López, Juana E. Mancilla</i> | |
| Análisis y comparación de diferentes métodos de reconstrucción de árboles semánticos | 81 |
| <i>Ramón Rivera Camacho, Ricardo Barrón Fernández, Hiram Calvo</i> | |
| Modelo de la dinámica de neurotransmisores por medio de caminantes aleatorios | 89 |
| <i>Jorge A. Herrera, Juan C. Chimal, Norma Sánchez Salas</i> | |
| Comparación de clasificadores para el reconocimiento de notas musicales | 103 |
| <i>Omar Velázquez López, José Luis Oropeza Rodríguez, Sergio Suárez Guerra</i> | |
| Algoritmo para el cálculo del ritmo en una señal de audio digital | 113 |
| <i>Luis Felipe Romero Morales, José Martín Flores Albino</i> | |

| | |
|--|-----|
| Optimización en ubicación de estaciones base de baja potencia LTE | 123 |
| <i>Yair Ali Cerecedo Torres, Miguel Sánchez Meraz, Antonio Valdovinos Bardají, Ángela Hernández Solana</i> | |
| Modelo dinámico de las afectaciones del tiempo de ciclo kanban en el flujo de producción de una línea de ensamble | 135 |
| <i>Lisaura Walkiria Rodríguez Alvarado, Jesús Loyo Quijada, Miguel Ángel López Ontiveros</i> | |
| Propuesta de un modelo de integración de PSP y Scrum para mejorar la calidad del proceso de desarrollo en una MiPyME | 147 |
| <i>Mauricio Leonardo Urbina Delgadillo, María Antonieta Abud Figueroa, Gustavo Peláez Camarena, Giner Alor Hernández, Alma Ivonne Sánchez García</i> | |
| Predicción de oxígeno disuelto en acuicultura semi-intensiva con redes neuronales artificiales | 159 |
| <i>Carlos Julián Torres González, Víctor Manuel Landassuri-Moreno, José Juan Carbajal Hernández, José Martín Flores Albino</i> | |
| Filtro de restauración de imágenes basado en la transformada discreta del coseno y el análisis de componentes principales | 169 |
| <i>Alejandro I. Callejas Ramos, Edgardo M. Felipe-Riverón, Pablo Manrique Ramírez, Oleksiy Pogrebnyak</i> | |

Estudio del análisis de componentes principales en bases de datos de calidad del aire

Abraham Sánchez López¹, Viridiana Cruz-Gutiérrez¹,
Mario Alberto Posada-Zamora¹, M. Teresa Torrijos M.¹,
M. Auxilio Osorio Lama²

¹ Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación, México

² Benemérita Universidad Autónoma de Puebla,
Facultad de Ingeniería Química, México

asanchez@cs.buap.mx, {viricruz, mariop}@rockkruz.net,
{tere.torrijos, mariauxosorio}@gmail.com

Resumen. El cambio climático es un problema de la sociedad actual que repercute en muchos aspectos de la vida cotidiana. Como es conocido, el aumento de los gases de efecto invernadero en la atmósfera afecta la salud de millones de personas. Es de suma importancia que las autoridades cuenten con herramientas que les permitan realizar una mejor toma de decisiones ante estos eventos. El presente trabajo propone el estudio del análisis de componentes principales en los datos de las mediciones de contaminantes de la ciudad de México con la finalidad de conseguir una representación más compacta de dichos datos, para posteriormente aplicar técnicas de agrupamiento y con ello obtener factores que permitan la emisión de una alerta de pre contingencia y contingencia.

Palabras clave: Componentes principales, cambio climático, algoritmos de clustering.

A Study of Principal Components Analysis of Air Quality Databases

Abstract. The climatic change is an escalated issue of the actual society that have repercussions in so many aspects of everyday life. As it is known, the increasing levels of the greenhouse effect gases have a negative impact on the overall health of millions of people. It is greatly important that the authorities count with tools that let them to prepare and execute a better decision making related to this type of events. The present work proposes the study of Principal Components Analysis in the air pollution measurements data of the Mexico City with the main goal to get a compacted representation of the data, to later be able to apply clustering techniques and with that obtain results that let the authorities

to emit precontingency and contingency alerts to the population.

Keywords: Principal components, climatic change, clustering algorithms.

1. Introducción

El desarrollo tecnológico ha facilitado los hábitos cotidianos, los negocios, la fabricación de grandes cantidades de productos, entre otro tipo de actividades industriales; sin embargo, estos avances han provocado un deterioro ambiental que amenaza seriamente al desarrollo de la sociedad.

El aumento de los gases de efecto invernadero en la atmósfera afecta la salud de millones de personas y es el principal factor que ha modificado el clima en el planeta Tierra. Ante esta situación, es necesario llevar a cabo acciones que nos permitan adaptarnos rápidamente a este cambio y mitigar los efectos que está produciendo. Si bien esta problemática se ha estudiado a nivel mundial por el grupo de expertos sobre cambio climático, corresponde ahora a los jefes de las naciones bajar el nivel de contaminación, establecer estrategias y coordinar acciones que contribuyan al conocimiento de esta problemática, ya que se presenta en diferentes formas, dependiendo de la situación geográfica y del nivel de desarrollo de cada país. En este sentido, se propone como alternativa viable, aplicar la tecnología de la inteligencia de negocios, como una estrategia de análisis que contribuya a llevar un control de la calidad del aire y de las variables climáticas asociadas; así como para proporcionar instrumentos que ayuden a establecer políticas públicas para realizar una mitigación en el cambio climático en la Zona Metropolitana del Valle de México (ZMVM) y en otras ciudades, siempre que se cuenten con estaciones de monitoreo para poder disponer de datos útiles.

En la actualidad, diversas organizaciones y gobiernos han implementado esquemas de medición de contaminantes y así obtener los índices de calidad de aire (ICA) de las diferentes regiones del planeta. En la Ciudad de México y en la ZMVM, la contaminación del aire se mide con el Índice Metropolitano de Calidad del aire (IMECA), el cual es usado para mostrar el nivel de contaminación y el nivel de riesgo que representa a la salud humana únicamente en esta región, en un tiempo determinado y así poder tomar medidas de protección.

En el presente trabajo se elabora una propuesta de aplicar la técnica conocida como análisis de componentes principales (ACP) en las mediciones de los contaminantes para establecer un patrón; los atributos de cada patrón, son los valores de cada contaminante y de esa forma su agrupamiento se podrá comparar con los datos a los que no se le haya aplicado el ACP. En [2,4] se describen técnicas de agrupamiento básicas utilizadas comúnmente como el método K-means y K-medoids para la agrupación, pero en este trabajo se exploran algoritmos más complejos como Fuzzy c-Means, Possibilistic c-Means, Competitive Leaky Learning y Valey Seeking. En la *Sección 2* se abordan definiciones conceptuales de clustering y de los aspectos generales del ACP, posteriormente, en la *Sección 3* se presenta el desarrollo del ACP, después en la *Sección 4* se dan a conocer

los resultados y las comparativas pertinentes de los algoritmos, finalmente, en la *Sección 5* se muestran las conclusiones y trabajo futuro de este trabajo.

2. Marco teórico

El clima es una descripción estadística de las condiciones de tiempo y sus variaciones, incluyendo condiciones promedio y extremas. Los gases de efecto invernadero juegan un rol importante en el deterioro del clima y provocan el cambio climático. Los gases de efecto invernadero incluyen vapor de agua, dióxido de carbono (CO_2), metano (CH_4), óxido nitroso (N_2O) y algunos gases industriales tales como cloro fluorocarbonos ($CFCs$). Estos gases actúan como una manta aislante, manteniendo la superficie de la tierra más caliente de lo que debería estar, esto es debido a que no se reflejan los rayos del sol al espacio. Una vez que estos gases son liberados a la atmósfera, muchos de ellos permanecerán ahí por un largo periodo de tiempo [1,3,4].

La primera fase de la estrategia consiste en preparar los datos para su agrupamiento, y dado que el conjunto inicial de datos tiene su origen en un grupo de hojas de cálculo, es necesario aplicar algunas técnicas como la reducción de dimensiones. Este proceso consiste en encontrar un espacio adecuado que es menor en la dimensión en el cual se representan los datos originales. Se espera que la representación de los datos ayude a

- explorar datos de gran dimensión con el objetivo de explorar estructuras o patrones que dirija a la formación de una hipótesis estadística,
- visualizar los datos con diagramas de dispersión, donde la dimensión es reducida a 2 o 3 dimensiones,
- analizar los datos en métodos estadísticos, como clustering, estimación de la densidad de probabilidad o de clasificación.

Un posible método para la reducción de dimensiones sería únicamente elegir subconjuntos de las variables para procesar y analizar estos grupos, sin embargo, en algunos casos, eso significaría deshacerse de mucha información útil. Una alternativa podría ser crear nuevas variables a partir de las variables originales en forma de combinaciones lineales [2,5].

2.1. Análisis de componentes principales

El propósito de análisis de componentes principales es reducir un espacio de dimensión p a un nuevo espacio de dimensión d , donde d es mucho menor que p , mientras que al mismo tiempo representa la variación de datos como sea posible. Con el ACP, se transforman los datos en un nuevo conjunto de coordenadas o variables que son una combinación lineal de las variables originales. Además, las observaciones en el nuevo espacio de componentes principales no están correlacionadas. Se espera obtener información y comprensión de los datos al analizar las observaciones en el nuevo espacio [2].

2.2. Técnicas de agrupamiento

Entre las técnicas de agrupamiento, se encuentra la de tipo no jerárquico [2], que consiste en dividir los datos en k particiones o grupos donde cada partición representa un grupo. El funcionamiento básicos de dicho método son

1. seleccionar K centroides iniciales, siendo K el número de grupos deseados,
2. asignar cada patrón al grupo que le sea más cercano,
3. reasignar o relocalizar cada observación a uno de los K grupos de acuerdo con algún criterio de paro,
4. termina si no hay reasignaciones de los puntos o si la reasignación satisface la regla de paro. En otro caso regresa al paso dos.

Entre estos métodos existen dos que son los más utilizados que es el K-means y K-PAM (Partitioning Around Medoids), estos métodos son de utilidad para encontrar grupos (clusters). Para poder unir variables o individuos es necesario tener algunas medidas numéricas que caractericen las relaciones entre las variables o los individuos. La medida de asociación puede ser una distancia o una similaridad; a continuación se describen brevemente dichas medidas:

- Cuando se elige una distancia como medida de asociación (por ejemplo la distancia euclídeana) los grupos formados contendrán individuos parecidos de forma que la distancia entre ellos ha de ser pequeña.
- Cuando se elige una medida de similaridad (por ejemplo el coeficiente de correlación) los grupos formados contendrán individuos con una similaridad alta entre ellos.

Entre los algoritmos más utilizados en el análisis de clusters se encuentran: el algoritmo K-means y el algoritmo K-medoids.

2.3. Validación de clusters, índice de la silueta

El índice de silueta es una métrica para evaluar el buen funcionamiento de los algoritmos de aprendizaje no supervisado. El objetivo de este índice es identificar el número óptimo de agrupamientos. En los algoritmos de aprendizaje no supervisado, el número de clústeres puede ser un parámetro de entrada del algoritmo (K-means). La determinación del número óptimo de clústeres tiene que ser realizado mediante alguna medida externa al algoritmo. El índice silueta es indicador del número ideal de clústeres. El valor de la silueta para cada punto es una medida de que tan similar es ese punto hacia los puntos que están en su propio clúster, cuando este es comparado con puntos en otros clusters.

El valor de la silueta para el i -ésimo punto, si está definido como

$$S_i = \frac{(b_i - a_i)}{\max(a_i, b_i)}, \quad (1)$$

donde a_i es la distancia promedio desde i -ésimo punto hacia los otros puntos en el mismo cluster, y b_i es la distancia mínima promedio desde el i -ésimo punto

hacia los puntos en un clúster diferente. El valor de la silueta tiene rangos de -1 a +1. Un alto valor de silueta indica que i está bien asignado a su propio clúster, y pobremente mal asignado a clústers vecinos. Si muchos puntos tienen un alto valor de silueta, entonces la solución del clúster es apropiada. Por el contrario, si muchos puntos tienen un valor bajo o negativo, entonces la solución puede tener muchas o pocas particiones. El criterio de evaluación de clúster con siluetas puede ser utilizada con cualquier métrica de asociación.

2.4. Otros algoritmos de agrupamiento

Los algoritmos Fuzzy c -Means son algunos de los principales algoritmos utilizados en el agrupamiento difuso y pertenecen a una clase de algoritmos basados en funciones objetivo [7]. Definen un criterio de agrupamiento en la forma de una función objetivo que depende de la partición difusa.

Los algoritmos Possibilistic c -Means aparecen con el objetivo de resolver el mal comportamiento de los algoritmos Fuzzy c -Means, al ser utilizados en conjuntos de datos con mucho ruido [8]. Estos algoritmos se caracterizan por interpretar los valores como grados de compatibilidad con los grupos, en lugar de probabilidades de pertenencia. Para esto, se relaja la restricción de las particiones difusas que obliga a que la suma de los grados de pertenencia de un elemento hacia todos los grupos sea uno, exigiendo solamente que al menos uno de los grados de pertenencia sea positivo.

3. Propuesta del estudio

Dada la insuficiencia de datos de calidad del aire, disponibles de la ciudad de Puebla, se optó por utilizar los datos para la Ciudad de México. Se obtuvieron los datos históricos de los contaminantes criterio considerados en el estudio, los cuales son ozono (O_3), dióxido de nitrógeno (NO_2), monóxido de nitrógeno (NO), dióxido de azufre (SO_2), monóxido de carbono (CO), partículas menores a 10 micrómetros (PM_{10}), partículas menores a 2.5 micrómetros ($PM_{2,5}$) y partículas fracción gruesa o “coarse” (PM_{CO}), se cuenta con una base de datos desde 1986. Debido a que en algunos años las medidas y el número de contaminantes criterio no eran consistentes, se decidió comenzar desde el año 1995 hasta 2014, con las siguientes consideraciones:

- De 1995 a 2003 se toman en cuenta los contaminantes (CO), (NO_2), (NO), (O_3), (PM_{10}) y (SO_2).
- De 2004 a 2011 se agrega el contaminante criterio ($PM_{2,5}$).
- De 2013 en adelante, se menciona en [6] que comenzó la medición de (PM_{CO}).

Concluida la limpieza de los datos, se lleva a cabo el ACP:

- **Análisis de la matriz de correlaciones:** Un análisis de componentes principales adquiere todo su sentido, si existen altas correlaciones entre las variables (esto indica que existe información redundante y por lo tanto, pocos factores explicarán una gran parte de la variabilidad total).

- **Selección de los factores:** Se realiza de forma que el primer factor recolecte la mayor proporción posible de la variabilidad original, el segundo factor debe por lo tanto recolectar la máxima variabilidad no recolectada por el primero, etc. De los factores se elegirán aquellos que recolecten un porcentaje de variabilidad considerado como suficiente (componentes principales).
- **Análisis de la matriz factorial:** Una vez que se han seleccionado los componentes principales, estos se representan en forma matricial. Cada elemento representa por lo tanto los coeficientes factoriales de las variables (las correlaciones entre las variables y los componentes principales).
- **Interpretación de los factores:** Para que un factor sea interpretado con facilidad, este tiene que exhibir las siguientes características:
 - Los coeficientes factoriales deben ser próximos a 1.
 - Una variable debe tener coeficientes elevados sólo con un factor.
 - No debe haber factores con coeficientes cercanos.
- **Cálculo de las puntuaciones factoriales:** Son las puntuaciones que tienen las componentes principales en cada caso y esto permite graficarlos.

4. Resultados

Al analizar todos los conjuntos de datos, se observa un total de 8760 registros por contaminante considerado (lo que corresponde a la medición por hora), por cada una de las estaciones. Se decidió tomar los datos de una estación de monitoreo para poder así crear un modelo inicial. La estación a ser elegida deberá medir todos los contaminantes ya que también existen estaciones que no proporcionan registros de algunas partículas. La estación elegida fue la de la delegación Tlalpan. En la Figura 1, se muestran los resultados obtenidos.

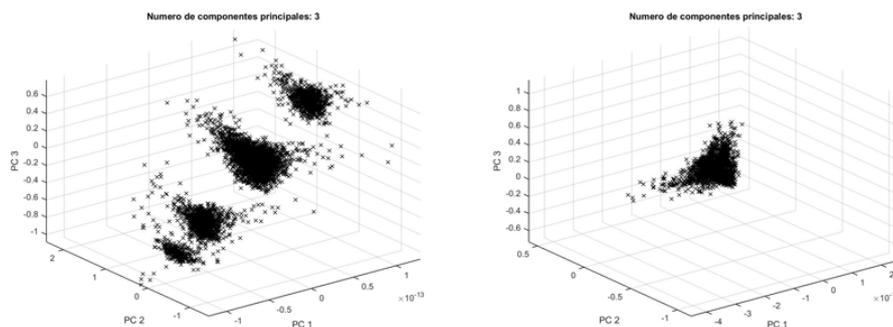


Fig. 1. Resultados del ACP para la estación Tlalpan en los años 2013 y 2014.

Al final del análisis se esperaba que cuando se realizaban las mediciones de PM_{10} , $PM_{2.5}$ y PM_{CO} existiera un agrupamiento, al observar los datos se puede apreciar que ya no hay una estabilidad de los contaminantes, así que se debe estudiar cuales son factores que hacen que la distribución se dispare.

También, se esperaba que, al incrementar las variables con datos dispersos, se obtuvieran más componentes, lo cual resultó cierto; pero efectuando un análisis visual, se observan algunos conglomerados de grupos con los que es posible hacer un análisis y agruparlos como se menciona en [3] para poder llegar a realizar los experimentos y ver qué resultados arrojan.

Con los resultados obtenidos del proceso anterior se prosiguió a realizar un estudio de clústers para observar si la consistencia de los datos persiste. Para el análisis se ocuparon los métodos K-means y K-medoids con los datos sin procesar y posteriormente con los datos obtenidos después del ACP, para poder hacer el estudio comparativo de ambas y así determinar si el ACP mantiene la consistencia de la información, y con ello poder probar la hipótesis que con la reducción de la dimensión de cada instancia se pueda llegar al mismo resultado o tener una aproximación aceptable.

Por la naturaleza de los métodos y dado que la inicialización de los centroides es de manera aleatoria se realizó la ejecución del método varias veces y después se evaluaron los resultados con la media de los valores con el índice de silueta para poder determinar el número óptimo de clústers. En la Tabla 1, se muestran las medias de los resultados de 20 ejecuciones con el método K-means del año 2003 en adelante, ya que en los conjuntos de datos anteriores no se muestra un gran cambio con el número de clúster óptimo que arrojó esta medición; se destacan los mejores valores en cada clúster, que representan el óptimo. En la Tabla 2 se muestra el número óptimo de cada conjunto de datos con el método K-medoids.

Se realizaron nuevamente las pruebas de clústers, con los datos obtenidos del ACP, con la finalidad de determinar el número óptimo de particiones y efectuar una comparativa. Las pruebas se realizaron con las mismas iteraciones límite, ejecuciones, mecanismo de medición e índice de silueta; con ello se trata de determinar si se mantuvieron los resultados realizando la reducción de dimensión.

Tabla 1. Índices de siluetas en K-means.

| Clústers | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|------------|---------------|---------------|--------|--------|--------|--------|--------|--------|
| Año | | | | | | | | |
| 2003 | 0.6496 | 0.5596 | 0.5329 | 0.6327 | 0.5682 | 0.5917 | 0.5975 | 0.5842 |
| 2004 | 0.5382 | 0.5598 | 0.5347 | 0.5332 | 0.5143 | 0.5008 | 0.4952 | 0.4967 |
| 2005 | 0.5791 | 0.5831 | 0.5406 | 0.5255 | 0.5259 | 0.5177 | 0.5108 | 0.5025 |
| 2006 | 0.5933 | 0.5566 | 0.5822 | 0.5543 | 0.5774 | 0.5137 | 0.5164 | 0.5059 |
| 2007 | 0.5496 | 0.4979 | 0.4988 | 0.4844 | 0.4950 | 0.4859 | 0.4914 | 0.4884 |
| 2008 | 0.5560 | 0.4920 | 0.5172 | 0.5073 | 0.5242 | 0.5144 | 0.5165 | 0.5041 |
| 2009 | 0.6095 | 0.5407 | 0.5505 | 0.5780 | 0.5427 | 0.5354 | 0.5161 | 0.5235 |
| 2010 | 0.5782 | 0.5249 | 0.5215 | 0.5062 | 0.5188 | 0.5080 | 0.5007 | 0.4856 |
| 2011 | 0.4265 | 0.3591 | 0.3623 | 0.3569 | 0.3557 | 0.3474 | 0.3347 | 0.3380 |
| 2012 | 0.4794 | 0.4175 | 0.3919 | 0.3910 | 0.3761 | 0.3446 | 0.3596 | 0.3800 |
| 2013 | 0.4308 | 0.4134 | 0.4186 | 0.4116 | 0.3820 | 0.3716 | 0.3619 | 0.3165 |
| 2014 | 0.4916 | 0.4137 | 0.4211 | 0.4047 | 0.3894 | 0.3720 | 0.3695 | 0.3609 |

También se utilizaron otros algoritmos con distintos comportamientos: Fuzzy c-Means (FCM), Possibilistic c-Means (PCM), Competitive Leaky Learning y Valey Seeking. Estos algoritmos tienen fortalezas y debilidades que se confirmaron con las pruebas ejecutadas sobre los datos de los contaminantes, utilizando el ACP y los datos sin el uso de la reducción de la dimensionalidad. Estos algoritmos no son óptimos para grandes cantidades de datos, debido a su comportamiento iterativo y aunque tengan una condición de paro externa, estos suelen ser tardados en tiempo de ejecución sin realizar el ACP, sin embargo, para crear un comparativo, se realizarán algunas pruebas.

Tabla 2. Resultados finales de los clústers con K-medoids.

| Año | Clúster | Índice |
|------|---------|--------|
| 2003 | 4 | 0.7321 |
| 2004 | 4 | 0.6872 |
| 2005 | 4 | 0.6870 |
| 2006 | 5 | 0.6951 |
| 2007 | 4 | 0.6849 |
| 2008 | 4 | 0.6936 |
| 2009 | 4 | 0.6999 |
| 2010 | 4 | 0.7066 |
| 2011 | 4 | 0.4563 |
| 2012 | 4 | 0.4711 |
| 2013 | 4 | 0.4553 |
| 2014 | 4 | 0.4475 |

Para el algoritmo Fuzzy c-Means (FcM), se usa el grado de compatibilidad del vector de una función objetivo con cierto clúster (pertenencia), el algoritmo es sensible a los ‘outliers’ o datos fuera de rango. También es sensible al grado de ‘defuzzificación’ que el valor debe de estar en un rango dado de pruebas. El algoritmo considera poder separar los clústers según un grado difuso de pertenencia, se hizo antes una prueba para poder buscar el número de clústers óptimo, con los métodos ya existentes, el cual con ciertas medidas arroja un número adecuado de división en las que se podría representar los datos.

Cuando se usa el ACP hace la separación casi igual a K-means y a K-medoids pero estos métodos hacen más separación. El número de clusters oscila entre 5-10 grupos, y en la visualización se nota la misma separación con los métodos simples de K-medoids y K-means.

El algoritmo Possibilistic c-Means (PcM), es ideal para revelar clústers compactos, igual que FcM se tiene un grado de pertenencia a cada clúster que se defina, pero es menos sensible al número exacto de clústers, este algoritmo es iterativo y tiene un costo computacional no tan grande por su comportamiento (Figuras 2 y 3).

El algoritmo Leaky Learning (LLA), es un algoritmo apropiado para revelar clústers compactos. Se asume el número de clústers, por lo cual hay que hacer

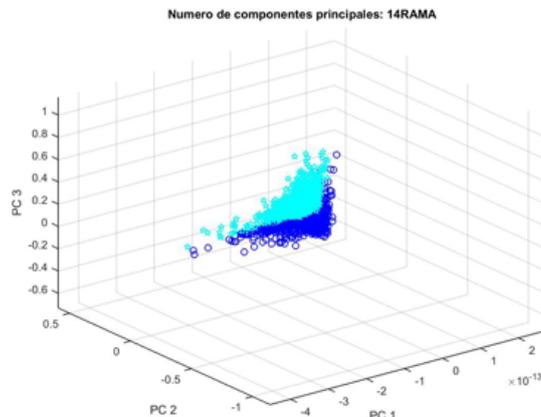


Fig. 2. Dos grupos encontrados, utilizando el algoritmo PcM para el año 2014.

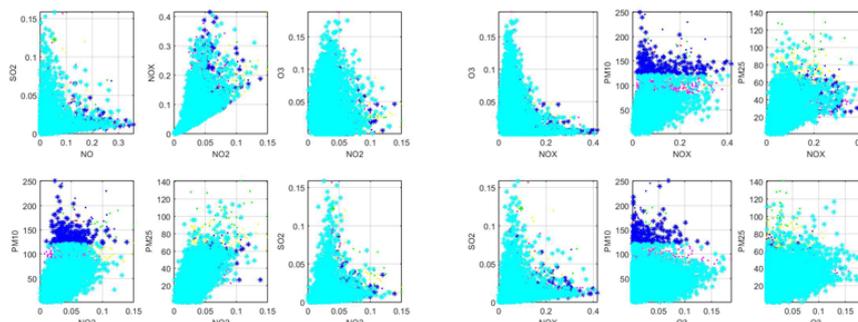


Fig. 3. Resultados del año 2007 sin el ACP con el algoritmo PcM.

pruebas para encontrar el número adecuado de clústers. Este algoritmo usa el término *densidad*, el cual requiere conocer en qué región se usa la estrategia de competición. Tiene parámetros de paro, los cuales deben de estar en un rango. El comportamiento del algoritmo muestra que con un número de 4 clústers, llega a ser obvia la separación de los datos y poder así interpretarlos. Este algoritmo es similar al algoritmo K-means, pero mucho más estable. Otro factor a considerar es que es mucho más rápido y las condiciones de paro son más claras. En los datos sin usar el ACP, se nota que se traslapan los clústers, esto indica que es indispensable hacer un tratamiento previo de los datos (Figuras 4 y 5).

En el algoritmo Valey Seeking Clustering (VS), los clusters son considerados picos de datos descritos por los individuos, y estos son separados por valles. El algoritmo es muy sensible con las variables con las que se inicializa, ya que se podría efectuar un refinamiento erróneo, esto se notó en todos los años, excepto en el año 2012, ya que se tiene un parámetro que se compara con las distancias, se tiene que encontrar el ideal, dado que, en estos datos con el ACP, las distancias entre los puntos es muy pequeña y si se excede comienza a absorber

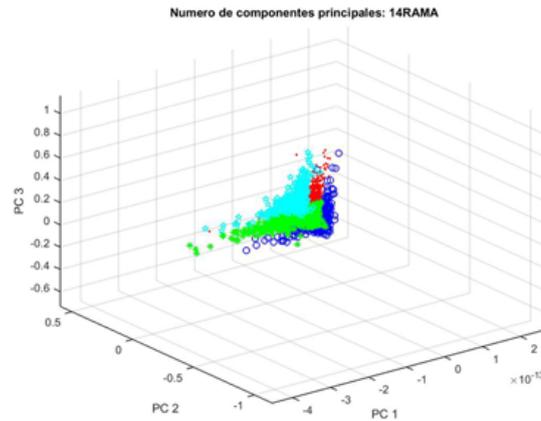


Fig. 4. Resultado del año 2014 sin el ACP con el algoritmo LLA.

a los demás grupos. Para poder tener resultados óptimos con este algoritmo hay que encontrar el paso anterior de la asignación para así se pueda hacer un refinamiento, y el valor de *desimilitud* correcto para el que no desaparezca los grupos. El proceso de pruebas de VS puede ser muy tardado, al encontrar el valor de *desimilitud*, y hacer bastantes pruebas.

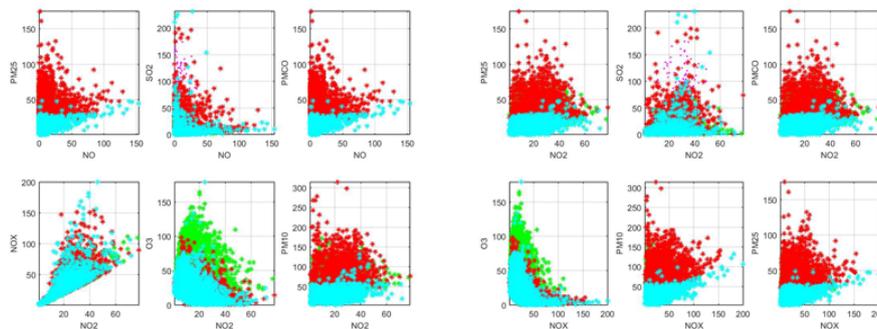


Fig. 5. Los grupos encontrados con el algoritmo LLA para el año 2014.

5. Conclusiones y trabajo futuro

El interés por realizar la presente investigación es para contestar las siguientes preguntas: ¿existe un patrón en los registros de cada año?, ¿sólo un contaminante criterio se dispara por medición?, ¿cuáles son los contaminantes que se disparan con mayor frecuencia?. Estas preguntas no pueden ser contestadas con solo tener el registro de la calidad del aire en un momento determinado, si no que se

requiere del análisis de las mediciones de calidad del aire para observar cual es el comportamiento de los datos y obtener las conclusiones pertinentes. Se considera que el análisis de clúster es una buena técnica para descubrir muchos patrones ocultos en los datos. Es claro que el análisis de componentes principales aporta una disminución importante en la dimensión. Con este tipo de estudios se pueden conocer cuáles son los contaminantes que están agrupados en el momento de declarar una contingencia, por ejemplo.

Como trabajo futuro inmediato queda por analizar si se obtienen los mismos resultados que en los experimentos propuestos por [1] y [3] y hacer un comparativo con el enfoque presentado, para hacer una destacada aproximación de un modelo para la toma de decisiones en la mejora de la “calidad de aire”. También se puede hacer un análisis de la “descomposición de valores singulares” como otro método lineal de reducción de dimensión o aplicar un método no lineal y hacer experimentos comparativos.

Referencias

1. Camarillo Ramírez, P., Sánchez López, A., Calva Rosales, L. J., Pérez Vásquez, I.: Análisis de datos de calidad del aire de la Zona Metropolitana del Valle de México mediante técnicas de agrupamiento. *Journal Research in Computing Science* 72, pp. 137–150 (2014)
2. Theodoridis, S., Koutroumbas, K.: *An Introduction to Pattern Recognition: A MATLAB Approach*. Elsevier Inc., USA (2010)
3. Sanchez, A., Reyes, J.: Analysis of air quality data in Mexico city with clustering techniques based on genetic algorithms. *Electronics, Communications and Computing CONIELECOMP*, pp. 27–31 (2013)
4. Everitt, B. S., Landau, S., Leese, M., Stahl, D.: *Cluster analysis*. Wiley 5th Edition, England (2011)
5. Ding, C., He, X.: K-means clustering via principal component analysis. In: *Proceedings of the 20th International Conference on Machine Learning* (2004)
6. Secretaría del Medio Ambiente, Ciudad de México, <http://www.aire.cdmx.gob.mx/>
7. Yang, M.S.: A Survey of Fuzzy Clustering. *Mathl. Comput. Modelling* 18 (11), pp. 1–16 (1993)
8. Krishnapuram, R., Keller, J.M.: A Possibilistic Approach to Clustering. *IEEE Transactions on Fuzzy Systems* 1, pp. 98–110 (1993)

Un enfoque de control de movimiento de un helicóptero con cuatro rotores

H. Yáñez-Badillo¹, R. Tapia-Olvera², F. Beltran-Carbajal³, O. Aguilar-Mejía¹,
I. López-García³

¹ Universidad Politécnica de Tulancingo,
Departamento de Posgrado, Tulancingo, Hidalgo,
México

² Universidad Nacional Autónoma de México,
Departamento de Ingeniería Eléctrica, Ciudad de México,
México

³ Universidad Autónoma Metropolitana,
Unidad Azcapotzalco, Departamento de Energía, Ciudad de México,
México

{dophugo.yanez,omar.aguilar}@upt.edu.mx, rtapia@fi-b.unam.mx,
{fbeltran,ilg}@azc.uam.mx

Resumen. En este artículo se describe un esquema de control para tareas de regulación y seguimiento de trayectorias planificadas para el movimiento de un helicóptero con cuatro rotores, el cual también se conoce como *Quadrotor*. El esquema de control consiste de cuatro controladores clásicos y una etapa de generación de trayectorias para el movimiento de traslación deseado en el plano $X - Y$. Se incluyen resultados de simulación donde se demuestra que el desempeño del esquema de control de movimiento especificado para el *Quadrotor* es aceptable.

Palabras clave: Quadrotor, control de movimiento, seguimiento de trayectorias, sistemas mimo.

A Motion Control Approach of a Helicopter with Four Rotors

Abstract. In this paper a control scheme is described for regulation and trajectory tracking tasks planned for the motion of a helicopter with four rotors which is also known as *Quadrotor*. The control scheme consists of four classical controllers and a generation stage of trajectories for the desired translational movement in the $X - Y$ plane. Simulations results show the acceptable performance of the control scheme specified for the *Quadrotor*.

Keywords: Quadrotor, motion control, trajectory tracking, mimo systems.

1. Introducción

El interés en el estudio de los vehículos aéreos no tripulados (UAV's, por sus siglas en inglés) se ha incrementado en los últimos años. Una de las principales plataformas en la que se han enfocado diversas investigaciones es el helicóptero con cuatro rotores comúnmente llamado *Quadrotor*. Éste tiene la capacidad de aterrizar y despegar verticalmente (VTOL, por sus siglas en inglés) característica que permite su operación en interiores de manera segura, tarea complicada para otro tipo de UAV's como los de ala fija [1].

En la literatura se han realizado algunas consideraciones fundamentales para el análisis de esta plataforma. En la mayoría de las contribuciones el *Quadrotor* se considera como un cuerpo rígido, por lo que se deben emplear dos marcos de referencia para describir su comportamiento dinámico. El primer marco de referencia se establece fijo respecto a la superficie terrestre O_e y el segundo solidario al cuerpo del robot O_b [2]. Existen diversas representaciones matemáticas para describir la dinámica del robot, las citadas en la literatura son los formalismos: Euler-Lagrange [3] y Newton-Euler [4], esta última es la adoptada para el presente estudio.

Las técnicas para regular el seguimiento de trayectorias son diversas, entre las principales se tienen a los controladores convencionales PD, PI y PID [5-9], controladores no lineales del tipo backstepping [10,11], modos deslizantes [12,13], controladores robustos H_∞ [14], difusos [15], redes neuronales [16], algoritmos híbridos [17,18] y control por planitud diferencial [19,20]. El presente artículo hace énfasis en los métodos de control clásicos, principalmente para remarcar sus versatilidad y facilidad de implementación, así como la funcionalidad para fundamentar métodos de control efectivos.

En [5] se presentan controladores PD y PID para la regulación del movimiento de rotación del robot, donde no se han considerado los acoples dinámicos provocados por los efectos giroscópicos del sistema. Los parámetros del controlador fueron sintonizados mediante simulaciones previas a la implementación, donde se concluye que las aproximaciones clásicas de control son suficientes para estabilizar el *Quadrotor* en condiciones de vuelo estacionario. Sin embargo, ante la presencia de perturbaciones de gran magnitud estos algoritmos no son capaces de estabilizar el sistema. En [6] se trabaja con controladores PI en cascada, donde la regulación de los principales movimientos del helicóptero se llevan a cabo a través del ajuste de las velocidades angulares de los cuatro rotores que constituyen el mecanismo. Por otro lado, en [7] se desarrolla un controlador PD mejorado, que se enfoca en la respuesta transitoria de un micro helicóptero de cuatro rotores ya que debido a sus características dinámicas es sensible incluso a pequeñas perturbaciones. En el trabajo se emplean los cuaternios para evitar singularidades en el modelo y el error de esta representación es minimizado con las ganancias del controlador, de tal forma que no afecten el desempeño del vuelo del *Quadrotor*. En [8] se establecen dos controladores PD para controlar los desplazamientos en las direcciones X y Y , mediante los ángulos θ y ϕ respectivamente. Una propuesta de un controlador de altitud PID restringido mediante técnicas de regulación cuadráticas (LQR) se propone en

[9]; el algoritmo propuesto se presenta de esta manera para que las ganancias del controlador PID sean determinadas dentro de las posibilidades físicas del sistema, sin degradar la acción del controlador mediante una función de costo.

A pesar de ser sensible a perturbaciones, los controladores clásicos presentan una estructura compacta y funcional con resultados satisfactorios, lo que ha permitido su implementación para modelos comerciales [18]. En la literatura es posible encontrar la estructura general de estos algoritmos aunado a un grado de robustez mediante técnicas como el análisis de backstepping [21], redes neuronales [22], por mencionar algunos, donde las ganancias del controlador se actualizan de forma dinámica, sin afectar la estabilidad del robot.

En este artículo se describe un esquema de control para tareas de regulación y seguimiento de trayectorias planificadas para el movimiento de un helicóptero con cuatro rotores. El esquema de control consiste de cuatro controladores clásicos y de una etapa de generación de trayectorias para el movimiento de traslación deseado en el plano $X - Y$. Se incluyen algunos resultados de simulación para mostrar el desempeño aceptable del esquema de control de movimiento especificado para el *Quadrotor*.

2. Dinámica de un helicóptero con cuatro rotores

Esta clase de vehículo aéreo presenta seis grados de libertad, mediante cuatro entradas de control principales que dependen de las velocidades angulares de cada rotor. El movimiento en el plano cartesiano se logra mediante movimientos de alabeo (*roll*), cabeceo (*pitch*) y giñada (*yaw*) que relacionan los ángulos de Tait-Bryan ϕ , θ y ψ (utilizados en el área de aeronáutica debido a su configuración), con los desplazamientos en las direcciones X , Y y Z , respectivamente. El sentido de la rotación de las aspas para cada rotor ω_i , $i = 1, \dots, 4$, deben de establecerse mediante pares, es decir ω_1 y ω_3 en el sentido conforme a las manecillas del reloj y ω_2 y ω_4 en sentido contrario al de las manecillas del reloj; la rotaciones de las hélices en contacto con el aire, producen fuerzas y pares que le permiten el movimiento a esta clase de vehículos aéreos (véase la Fig. 1).

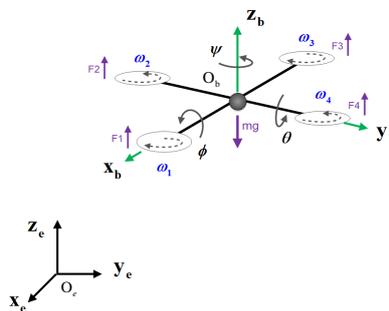


Fig. 1. Diagrama de cuerpo libre de un Quadrotor.

El modelo matemático que describe la dinámica del helicóptero con cuatro rotores está dado por [4]

$$\ddot{x} = \frac{1}{m} (\sin \theta \cos \phi \cos \psi + \sin \phi \sin \psi) u_1 - \frac{c_x}{m} \dot{x}, \quad (1)$$

$$\ddot{y} = \frac{1}{m} (\sin \theta \sin \psi \cos \phi - \sin \phi \cos \psi) u_1 - \frac{c_y}{m} \dot{y}, \quad (2)$$

$$\ddot{z} = \frac{1}{m} (\cos \phi \cos \theta) u_1 - g - \frac{c_z}{m} \dot{z}, \quad (3)$$

$$\ddot{\phi} = \frac{J_y - J_z}{J_x} \dot{\theta} \dot{\psi} - \frac{J_r}{J_x} \dot{\theta} \Omega_n - \frac{b_\phi}{J_x} \dot{\phi} + \frac{1}{J_x} u_2, \quad (4)$$

$$\ddot{\theta} = \frac{J_z - J_x}{J_y} \dot{\phi} \dot{\psi} + \frac{J_r}{J_y} \dot{\phi} \Omega_n - \frac{b_\theta}{J_y} \dot{\theta} + \frac{1}{J_y} u_3, \quad (5)$$

$$\ddot{\psi} = \frac{J_x - J_y}{J_z} \dot{\phi} \dot{\theta} - \frac{b_\psi}{J_z} \dot{\psi} + \frac{1}{J_z} u_4, \quad (6)$$

donde m es la masa del robot, g es la constante de la aceleración de la gravedad y u_1 es la entrada principal de fuerza que engloba los términos F_1, F_2, F_3 y F_4 y un coeficiente de empuje b . En el diagrama esquemático de cuerpo libre mostrado en la Fig. 1, se aprecia que el movimiento en la dirección x se puede regular mediante el desplazamiento angular θ , para la dirección y la regulación es de ϕ y el control de la posición en dirección z denominada altitud, se consigue mediante la manipulación de las fuerzas producidas por cada rotor.

Las entradas de control u_2, u_3 y u_4 se producen debido a la diferencia de fuerzas producidas por cada rotor y la relación de estas con un coeficiente de arrastre d , que corresponden de manera directa con los movimientos de *roll*, *pitch* y *yaw* respectivamente y se utilizan como las entradas de par de control del sistema; Ω_n es la suma algebraica de las velocidades angulares y se observa en la ecuación (7), ésta contribuye a los efectos giroscópicos provocados por la rotación de las aspas de los motores, en diversos trabajos se presenta como una perturbación que debe ser compensada por el controlador propuesto; el momento de inercia del rotor se expresa mediante J_r . Debido a que la estructura del robot se considera simétrica, solamente se presentan los elementos de la diagonal del tensor de inercia J_{xx}, J_{yy} y J_{zz}

$$\Omega_n = -\omega_1 + \omega_2 - \omega_3 + \omega_4. \quad (7)$$

Los parámetros b_r son los coeficientes de amortiguamiento rotacional para $r = \phi, \theta, \psi$, en tanto c_n son los coeficientes de amortiguamiento traslacional para $n = x, y, z$, los coeficientes de empuje y arrastre, b y d respectivamente, relacionan idealmente a las entradas principales de fuerza y par con el cuadrado de las velocidades angulares de cada motor $\omega_1, \omega_2, \omega_3$ y ω_4 , estos coeficientes pueden ser aproximados mediante la teoría del elemento pala.

3. Un esquema de control basado en el seguimiento de trayectorias de referencia

El objetivo del esquema de control presentado en este trabajo, es la regulación y seguimiento de trayectorias de referencia planificadas para los movimientos de rotación y traslación del *Quadrotor*. El esquema de control propuesto esta estructurado en dos etapas principales como se muestra en la Fig. 2. La primera etapa del esquema de control se basa en el seguimiento de los desplazamientos en las direcciones X y Y , donde se determinan los desplazamientos angulares de referencia para θ y ϕ . La segunda etapa se encarga del seguimiento de las trayectorias establecidas para z , ϕ , θ y ψ .

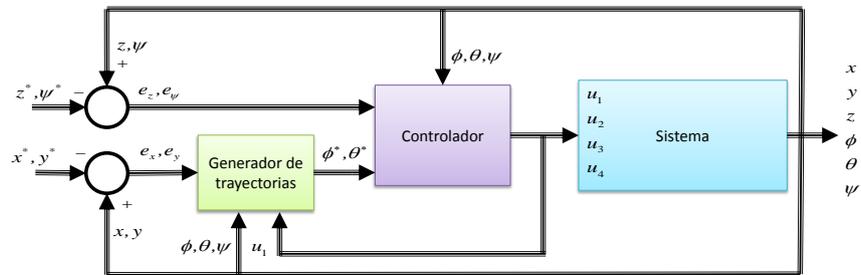


Fig. 2. Diagrama de bloques del esquema de control.

En el esquema de la Fig. 2 se presenta la primera etapa del esquema de control como un bloque generador de trayectorias que determina los desplazamientos angulares de referencia θ^* y ϕ^* , para el correcto seguimiento de las trayectorias planificadas mediante (8) y (9), las expresiones se deducieron de (1) y (2)

$$\phi^* = \sin^{-1} \left(\frac{m}{u_1} \left[\left(v_x + \frac{c_x}{m} \dot{x} \right) \sin \psi - \left(v_y + \frac{c_y}{m} \dot{y} \right) \cos \psi \right] \right), \quad (8)$$

$$\theta^* = \sin^{-1} \left(\frac{m v_x + c_x \dot{x}}{u_1 \cos \phi \cos \psi} - \frac{\sin \phi \sin \psi}{\cos \phi \cos \psi} \right), \quad (9)$$

aquí, los controladores auxiliares v_x y v_y , presentan la estructura expresada en la ecuación (10)

$$v_k = -\beta_2 \dot{e}_k - \beta_1 e_k - \beta_0 \int_0^t e_k dt, \quad (10)$$

para $k = x, y$; donde los errores de seguimiento para cada dirección se define como: $e_x = x - x^*$ y $e_y = y - y^*$. Aquí el * indica las trayectorias planificadas para el *Quadrotor*. Por lo tanto las dinámicas de los errores de seguimiento en lazo cerrado están dadas por

$$\ddot{e}_k^{(3)} + \beta_{2,k}\ddot{e}_k + \beta_{1,k}\dot{e}_k + \beta_{0,k}e_k = 0. \quad (11)$$

En cuanto a la segunda etapa se tiene que el controlador u_1 se encarga de la regulación y seguimiento de la trayectoria vertical del sistema en la dirección Z y se relaciona directamente con la suma total de las fuerzas producidas por los rotores. Los controladores propuestos u_2 y u_3 están relacionados con los movimientos de *roll* y *pitch*, su regulación y seguimiento son de suma importancia ya que de ellos depende el control de desplazamiento en las direcciones X y Y , por su parte u_4 hace referencia al movimiento de *yaw*, las expresiones para estos controladores son

$$u_1 = \frac{1}{\cos \phi \cos \theta} (mv_z + c_z \dot{z} + mg), \quad (12)$$

$$u_2 = J_x v_\phi - (J_y - J_z) \dot{\theta} \dot{\psi} + J_r \dot{\theta} \Omega_n + b_\phi \dot{\phi}, \quad (13)$$

$$u_3 = J_y v_\theta - (J_z - J_x) \dot{\phi} \dot{\psi} - J_r \dot{\phi} \Omega_n + b_\theta \dot{\theta}, \quad (14)$$

$$u_4 = J_z v_\psi - (J_x - J_y) \dot{\phi} \dot{\theta} + b_\psi \dot{\psi}, \quad (15)$$

donde los controladores v_z, v_ϕ, v_θ y v_ψ , presentan la estructura en (16) con $p = z, \phi, \theta, \psi$. Los errores de seguimiento están definidos como: $e_z = z - z^*$, $e_\phi = \phi - \phi^*$, $e_\theta = \theta - \theta^*$ y $e_\psi = \psi - \psi^*$. Por lo tanto los controladores se definen como

$$v_p = -\beta_{2,p}\dot{e}_p - \beta_{1,p}e_p - \beta_{0,p} \int_0^t e_p dt. \quad (16)$$

Las ganancias de control $\beta_{i,j}, i = 0, 1, 2, j = x, y, z, \phi, \theta, \psi$, se deben de seleccionar para que los polinomios característicos asociados a la expresión (11) sean polinomios *Hurwitz* (estables).

4. Resultados

Para validar el desempeño del enfoque de control propuesto, se realizaron diversas simulaciones mediante el uso de métodos numéricos, donde $g = 9.81$ m/s, $m = 0.973$ kg, $J_r = 0.0000286$ Nms², $J_x = J_y = 0.00866$ Nms², $J_z = 0.00866$ Nms², $c_{x,y,z} = 0.0001$ Ns/m y $b_{\phi,\theta,\psi} = 0.0001$ Nms. En la Fig. 3(a) se observan las trayectorias planificadas y el seguimiento realizado por el *Quadrotor* en cada una de las direcciones en el plano cartesiano, la trayectoria planificada está definida para la dirección Y como

$$y^* = r_y \sin(\omega_y t), \quad (17)$$

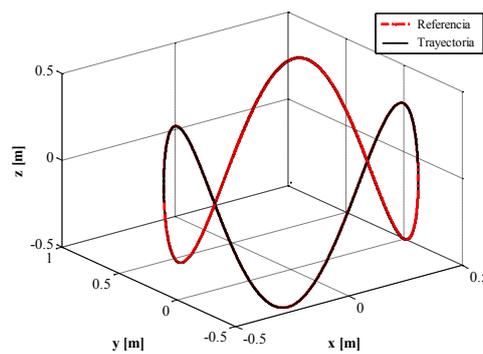
para la dirección X

$$x^* = r_x \sin(\omega_x t), \quad (18)$$

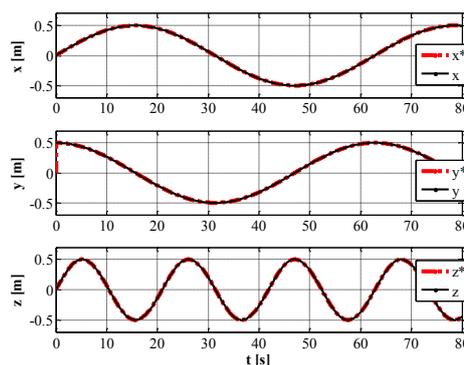
donde los radios $r_{x,y} = 0.5$ m y las velocidades angulares $\omega_{x,y} = 0.1$ rad/s. En cuanto a Z , la trayectoria deseada es

$$z^* = r_z \sin(\omega_z t), \quad (19)$$

con un radio $r_z = 0.5$ m y $\omega_z = 0.3$ rad/s. La referencia $\psi^* = 0.5$ rad para $t > 0$ s. Los valores de las ganancias se definieron como: $\beta_{0,i} = 18.5$, $\beta_{1,i} = 110$, $\beta_{2,i} = 250$, para $i = x, y, z$ y $\beta_{0,j} = 45000$, $\beta_{1,j} = 3450$, $\beta_{2,j} = 101$, para $j = \phi, \theta, \psi$. En la Fig. 3(b) se observa que el vehículo aéreo con cuatro rotores tiene la capacidad de hacer el seguimiento de manera adecuada, el algoritmo de control le permite seguir la referencia de las trayectorias planificadas de manera suave y debido a las características del diseño del controlador el error converge a cero. Un aspecto importante a considerarse en el análisis mediante simulación, son las magnitudes de las señales de control ya que existen limitaciones físicas por parte de los actuadores en la entrega de fuerzas y pares al sistema.



(a)



(b)

Fig. 3. Seguimiento de trayectoria en las direcciones X , Y y Z .

En la Fig. 4 se observa que las respuestas de los controladores principales, no presentan respuestas abruptas que pudieran verse limitadas en la implementación, con propósitos de presentación se analizan durante 2s. La señal de control u_4 permanece en cero después de alcanzar la referencia establecida $\psi = 0.5$ rad. Por su parte u_1 se observa que siempre esta actuando contra la fuerza de gravedad, evitando que el helicóptero entre en contacto con el suelo. El seguimiento de las trayectorias planificadas para los ángulos de referencia, establecidos por el generador de trayectorias se aprecia en la Fig. 5 y la Fig. 6, se observa el comportamiento del sistema bajo la acción de los controladores; es claro apreciar que el controlador tiene la capacidad de que el sistema alcance las referencias en tiempo finito. Sin embargo, se debe de ser cautelosos cuando se determinan las ganancias del controlador ya que las señales de control podrían saturar el sistema, lo que no es deseable en la implementación de laboratorio.

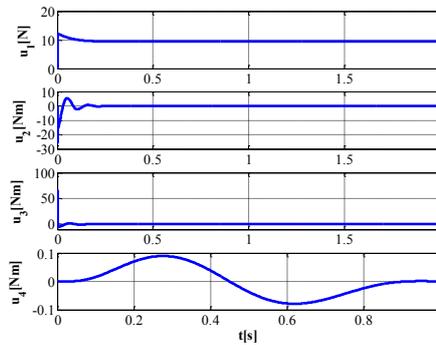


Fig. 4. Señales de control.

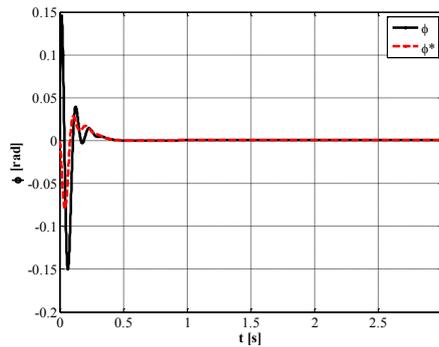


Fig. 5. Seguimiento de la trayectoria planificada de ϕ .

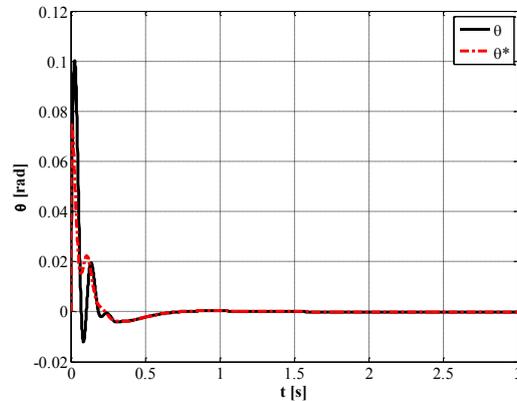


Fig. 6. Seguimiento de la trayectoria planificada de θ .

5. Conclusiones

En este artículo se ha descrito un esquema de control para tareas de regulación y seguimiento de trayectorias planificadas para el movimiento de un helicóptero con cuatro rotores. El enfoque de control propuesto es capaz de seguir trayectorias variables planificadas de manera eficiente. Además, la implementación del esquema de control propuesto es relativamente simple, obteniendo un desempeño aceptable para la regulación y seguimiento de trayectorias planificadas para todas las variables de estado que describen la configuración de movimiento de un *Quadrotor*. En futuros trabajos se consideraran la influencia de posibles perturbaciones, afectando la dinámica del sistema, en el diseño y evaluación de esquemas de control de movimiento.

Referencias

1. Corke, P.I.: Robotics, vision and control: fundamental algorithms in MATLAB. Springer, Berlin (2011)
2. Alderete, T.S.: Simulator Aero Model Implementation. Technical Paper, pp. 1–21 (1997)
3. Castillo, P., García, P., Lozano, R., Albertos, P.: Modelado y estabilización de un helicóptero con cuatro rotores. Revista Iberoamericana de Automática E Informática Industrial RIAI, 4(1), pp. 41–57 (2007)
4. Bouabdallah, S., Murrieri, P., Siegwart, R.: Design and control of an indoor micro quadrotor. In: Proceedings of the IEEE International Conference on Robotics and Automation ICRA '04, Vol. 5, pp. 4393–4398 (2004)
5. Bouabdallah, S., Noth, A., Siegwart, R.: PID vs LQ control techniques applied to an indoor micro quadrotor. In: Proceedings of the IEEE/RSJ International Conference In Intelligent Robots and Systems, (IROS 2004), Vol. 3, pp. 2451–2456 (2004)

6. Fernando, H.C.T.E., Silva, A.T.A.De, Zoysa, M.D.C.De, Dilshan, K.A.D.C., Munasinghe, S.R.: Modelling, simulation and implementation of a quadrotor UAV. In: Proceedings of the IEEE 8th International Conference on Industrial and Information Systems ICIIS , pp. 207–212 (2013)
7. Johnson, N.L., Leang, K.K.: Enhanced Proportional-Derivative Control of a Micro Quadcopter. In: Proceedings of the ASME Dynamic Systems and Controls Conference, pp. 1–5 (2013)
8. Erginer, B., Altug, E.: Modelling and PD Control of a Quadrotor VTOL Vehicle. In: Proceedings of the IEEE Intelligent Vehicles Symposium, pp. 894–899 (2007)
9. Leong, B.T.M., Low, S.M., Ooi, M.P.L.: Low-cost microcontroller-based hover control design of a quadcopter. In: Proceedings of the International Symposium on Robotics and Intelligent Sensors (IRIS), Vol. 41, pp. 458–464 (2012)
10. Bouabdallah, S., Siegwart, R.: Backstepping and sliding-mode techniques applied to an indoor micro Quadrotor. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 224–2252 (2005)
11. Ha, C., Zuo, Z., Choi, F. B., Lee, D.: Passivity-based adaptive backstepping control of quadrotor-type UAVs. *Robotics and Autonomous Systems*, 62(9), pp. 1305–1315 (2014)
12. Zheng, E.H., Xiong, J.J., Luo, J.L.: Second order sliding mode control for a quadrotor UAV. *ISA Transactions*, 53(4), pp. 1–7 (2014)
13. Luque-Vega, L., Castillo-Toledo, B., Loukianov, A. G.: Robust block second order sliding mode control for a quadrotor. *Journal of the Franklin Institute*, 349(2), pp. 719–739 (2012)
14. Raffo, G.V., Ortega, M.G., Rubio, F.R.: An integral predictive/nonlinear H_∞ control structure for a quadrotor helicopter. *Automatica*, 46(1), pp. 29–39 (2010)
15. Raharja, N.M., Iswanto, Faris, M., Cahyadi, A.I.: Hover position quadrotor control with fuzzy logic. In: Proceedings of the 0st International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE), pp. 89–92 (2014)
16. Emran, B.J., Yesildirek, A.: Robust Nonlinear Composite Adaptive Control of Quadrotor. *International Journal of Digital Information and Wireless Communications*, 4(2), pp. 213–225 (2014)
17. Hsu, C.F., Chang, C.W.: Intelligent dynamic sliding-mode neural control using recurrent perturbation fuzzy neural networks. *Neurocomputing*, 173, pp. 734–743 (2016)
18. Liu, H., Li, D., Zuo, Z., Zhong, Y.: Robust three-loop trajectory tracking control for quadrotors with multiple uncertainties. *IEEE Transactions on Industrial Electronics*, 63(4), pp. 2263–2274 (2016)
19. Sira-Ramírez, H.: On the linear control of the quad-rotor system. In: Proceedings of the 2011 American Control Conference. pp. 3178–3183 (2011)
20. Aguilar-Ibáñez, C., Sira-Ramírez, H., Suárez-Castañón, M.S., Martínez-Navarro, E., Moreno-Armendariz, M.A.: The trajectory tracking problem for an unmanned four-rotor system: flatness-based approach. *Int. J. Control*. 85, pp. 69–77 (2012)
21. Mian, A.A., Mian, I.A., Wang, D.: Backstepping based PID control strategy for an underactuated aerial robot. In: IFAC Proceedings (IFAC-PapersOnline), Vol. 17, pp. 15636–15641 (2008)
22. Fatan, M., Sefidgari, B.L., Barenji, A.V.: An adaptive neuro PID for controlling the altitude of quadcopter robot. In: Proceedings of the IEEE 18th International Conference on Methods & Models in Automation & Robotics (MMAR), pp. 662–665 (2013)

Comparación cuantitativa de matrices de datos

Angel Mandujano-García, Jesús Figueroa-Nazuno, Hiram Calvo

Instituto Politécnico Nacional,
Centro de Investigación en Computación, Ciudad de México,
México

b140477@sagitario.cic.ipn.mx, {jfn,hcalvo}@cic.ipn.mx

Resumen. Son pocas las técnicas de alineamiento elástico entre matrices de datos. Las que existen realizan un análisis comparando la similitud de características, o algunas son puramente estadísticas. En este trabajo se presenta una técnica moderna para la comparación elástica de matrices numéricas que considera todos los datos disponibles de manera global y no por análisis de extracción de características como hacen otras técnicas. Se presentan resultados de la experimentación utilizando como ejemplo datos de funciones matemáticas; sin embargo, este método puede funcionar para cualquier objeto o fenómeno que pueda ser representado en forma matricial.

Palabras clave: Medida de distancia, alineamiento elástico, matrices numéricas, Fréchet moderno, funciones matemáticas.

Quantitative Comparison on Data Matrices

Abstract. There are few elastic matching techniques that take matrices as input. Perform an analysis comparing the similarity of features while the existing ones techniques are purely statistical. A modern technique for elastic matching comparison is performed in this paper. This technique considers all data in the matrices, globally, and not just analyzing extracted features like other available techniques do. Experimental results are presented using a synthetic dataset of mathematical functions; however this method works for any object or phenomenon that can be represented as matrix.

Keywords: Distance measurement, elastic matching, matrix representation, modern Fréchet, mathematical functions.

1. Introducción

La comparación es un aspecto muy importante dentro de la vida diaria y de la computación, siempre estamos tomando decisiones que se basan en algún tipo de comparación. Cuando comparamos, lo primero que realizamos es obtener de alguna manera, las características o rasgos descriptivos de los objetos implicados en la comparación, y así, se pueden observar las diferencias y/o similitudes entre éstos. Esta estrategia en los seres humanos es realizada con naturalidad; sin embargo, dentro de la computación llevarla a cabo no es una tarea sencilla. Las computadoras necesitan de alguna herramienta matemática que se ocupe de realizar una medición cuantitativa y realizar una comparación. Con base en eso, es común realizar una métrica de los rasgos característicos de los objetos. Una alternativa, que se presenta en este artículo, es que una computadora pueda realizar una comparación no considerando rasgos descriptivos, sino tomando en cuenta todos los datos que se tienen disponibles, es decir, de manera global. La alternativa se basa en el uso de una técnica que realice un alineamiento elástico (*Elastic Matching*) sobre objetos en dos dimensiones.

Elastic Matching (EM) se ha utilizado en diversos problemas, como el reconocimiento de rostros, el reconocimiento de huellas dactilares, análisis de imágenes médicas, visión por computadora, entre otros. EM es una técnica que en general ha dado buenos resultados. Formalmente hablando, EM es definido como un problema de optimización con respecto a un mapeo elemento-elemento, lineal o no lineal [6]. En otras palabras, EM mide el esfuerzo que se tiene al ajustar A sobre B , siendo A y B dos matrices. EM optimiza el problema de alineamiento en varias dimensiones, 1D, 2D, 3D, etc. Para este trabajo se realiza un alineamiento en dos dimensiones, también llamado *Two-Dimensional Warping* (2DW). Este alineamiento ocurre en arreglos bidimensionales también llamados matrices, por lo tanto el objeto A y B son dos matrices que contienen datos numéricos. EM es un problema que presenta una complejidad computacional NP-Completo [7] debido a los grados de libertad que poseen las matrices. Las ventajas de EM sobre las técnicas de selección de rasgos son: EM es adaptativa, así generalmente posee mayor capacidad para obtener diversas deformidades que las técnicas clásicas no pueden, la optimización 2DW por sí misma, describe la deformación de carácter dominante. Este hecho muestra que EM posee propiedades útiles de técnicas de análisis estructural. EM puede estar relacionado con los marcos estadísticos y estocásticos. *Active Shape Models* y *2D HMMs* son dos buenas técnicas de ejemplos [10]. Las características de EM principalmente dependen de dos factores: (i) la formulación de 2DW, afecta el rango de deformaciones compensables. Esto quiere decir que la formulación de 2DW está relacionada con el problema que se quiere resolver; (ii) la estrategia de optimización de 2DW, afecta la precisión de los resultados de EM. De manera general, en [10] se menciona que las estrategias para obtener una solución óptima global proveen resultados más precisos que aquellas soluciones que son sub-óptimas. La técnica propuesta se basa en Programación Dinámica, donde se proporciona una solución óptima global.

Fréchet Moderno (FM) es el nombre de la técnica de comparación matricial que se ha desarrollado en el presente trabajo. FM es una medida elástica entre matrices de datos. Se le ha puesto ese nombre debido a que se basa en la Distancia de Fréchet. La distancia evaluada bajo esta aproximación es invariante a las deformaciones y surge debido a la falta de técnicas que analizan los datos representados en dos matrices de manera directa.

Existen técnicas que realizan la comparación de matrices, como la presentada en [6], que emplea una reducción de dimensionalidad en las matrices, usando Eigenvalores [9], esto es, se obtienen los datos característicos de cada matriz y eventualmente se realiza una comparación de rasgos característicos. Sin embargo son pocas y es muy difícil encontrar técnicas de comparación global. El presente trabajo se desarrolla la técnica Fréchet Moderno (FM), es una técnica muy útil cuando se requiere comparar datos de manera global, es decir, cuando se necesita considerar todos los datos que se tienen en las matrices y no los datos característicos. Se puede aplicar en diferentes ramas de la ciencia como: la minería de datos, la bioinformática, tratamiento de imágenes, reconocimiento de patrones, entre otros.

2. Antecedentes

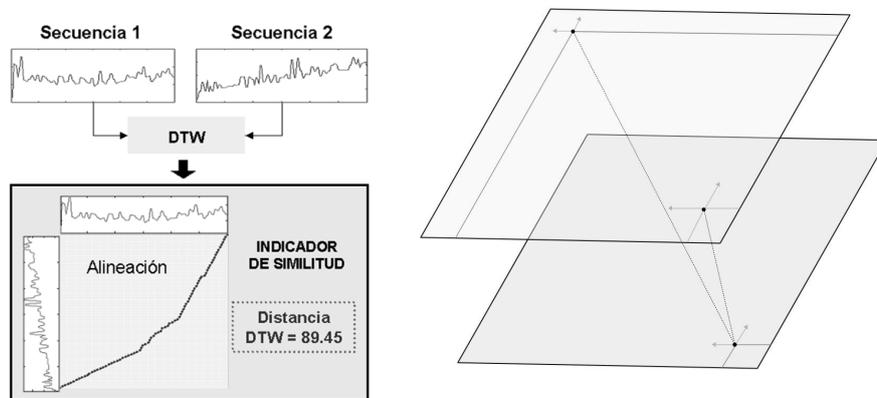
En esta sección se explican conceptos importantes para la descripción de la distancia Fréchet Moderno, se explican brevemente conceptos necesarios para la explicación y realización de la técnica:

Similitud entre matrices: El análisis convencional para la medición de similitud es mediante la selección de aquellas propiedades que se consideran importantes o relevantes; sin embargo, hacer una selección apropiada es una tarea muy difícil, aunada a esto, siempre hay pérdida de información en los datos originales, porque existe algún proceso de descarte [4]. El reto de investigación es desarrollar procedimientos sin hacer dicha selección, sino utilizar técnicas que permitan emplear toda la información disponible sobre las entidades involucradas. Para este caso se trata de matrices de datos numéricos.

Distancia de Fréchet: Maurice Fréchet fue un matemático francés que hizo trabajos de topología muy importantes. En su tesis doctoral [5, 8], presenta un procedimiento general para medir la similitud entre dos curvas F y G donde se tiene en cuenta el orden de los puntos a lo largo de las dos curvas o secuencias. De manera informal, Fréchet presentó una analogía sencilla: la Distancia de Fréchet es la longitud mínima de la correa necesaria para conectar a una persona con su perro que salen de paseo, ambos caminan sobre trayectorias diferentes, pero que tienen la misma dirección, sin poder regresar [1].

Dynamic Time Warping (DTW): Es una transformación que permite la expansión y comparación de una secuencia. Generalmente donde más se utiliza es en series de tiempo para la alineación local y global con respecto a otra secuencia,

con el objeto de minimizar la distancia base, que comúnmente es la distancia euclidiana. DTW es propia para computar la semejanza entre secuencias que se encuentran desfasadas una con respecto a la otra, o cuando una ellas presenta o ausenta segmentos con respecto a la otra [2]. Dadas dos secuencias F y G , DTW forma una matriz de dos dimensiones. Intuitivamente cada celda de la matriz representa un mapeo de un valor en la secuencia F con un valor de la secuencia G . El mapeo es la suma de la distancia individual de los mapeos previos antes calculados (ver Fig. 1a). La salida de DTW es la distancia mínima entre las secuencias y se encuentra en la última posición de la matriz generada. DTW está basada en la idea fundamental de la Distancia de Fréchet.



(a) Funcionamiento de *Dynamic Time Warping*. (b) Recorrido completo de la distancia de Fréchet Moderno.

Fig. 1. Distancias en secuencias y distancias en matrices.

3. Descripción de la técnica Fréchet moderno

Fréchet Moderno (FM) extiende el método de DTW a dos dimensiones, es decir, los datos ahora no son secuencias o arreglos unidimensionales, sino matrices numéricas. FM se basa en DTW de tal manera que se pueden analizar datos en R^2 . La entrada para FM son matrices de datos A y B de tamaño (P, Q) y (R, S) respectivamente. Fréchet Moderno es un método de comparación elástica que se obtiene comparando la distorsión que se encuentra en filas y columnas de la matriz A contra todas las filas y columnas de la matriz B , de esta manera se hace un recorrido completo en las dos matrices de entrada, como se muestra en la Fig. 1b.

Se forma una matriz de distancias acumuladas de cuatro dimensiones llamada M , cada celda de M se refiere a una alineación entre algunas celdas de A y B .

Esto es análogo a lo que ocurrió con DTW (ahora no son valores de secuencias, sino secuencias completas). $M(p, q, r, s)$ se refiere a una alineación entre $A(p, q)$ y $B(r, s)$.

De manera similar que en DTW, ahora se debe ir llenando la matriz M que guarda la distancia acumulada; una celda de M está dada por la mínima distancia de los mapeos previos de la posición actual. De este modo, la distancia entre las matrices de datos se encuentra en la última posición de M . Dados 4 índices (i, j, k, l) , en la matriz M se calcula lo siguiente:

$$M(i, j, k, l) = \min\{\text{ETAPAS_PREVIAS}(i, j, k, l) + \text{Costo}(i, j, k, l)\}$$

$$\text{costo} = \text{DTW}(R_1, R_2) + \text{DTW}(C_1, C_2).$$

Para una sola matriz, cada coordenada (i, j) tiene tres posibles etapas previas, sujetas a las limitaciones de frontera: $(i-1, j-1)$, $(i, j-1)$ y $(i-1, j)$. En la Fig. 2 se muestran las etapas previas de las matrices separadas, es decir, los incisos $a), b), c)$ son las etapas previas de la coordenada (i, j) en la matriz A que sería el inciso $d)$, mientras que los incisos $e), f), g)$ son las correspondientes en la matriz B de una coordenada específica $h)$.

Las etapas previas totales que se ocupan en Fréchet Moderno son el resultado de combinar las etapas previas de la matriz A con las de la matriz B , exceptuando las coordenadas $d)$ y $h)$, que indican la etapa actual que se está calculando. En otras palabras, las etapas previas son los puntos adyacentes con coordenada individual más pequeña en la matriz M . En la Tabla 1, se tienen las etapas previas totales con su respectiva función costo.

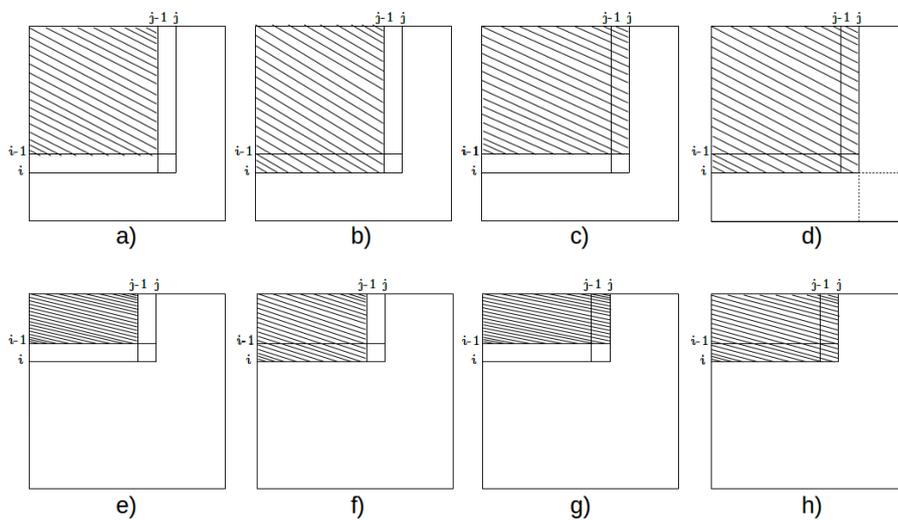


Fig. 2. Etapas previas para la matriz A , son $a), b)$ y $c)$. $d)$ es la etapa actual. Similarmente son las etapas previas de la matriz B .

Tabla 1. Etapas totales con el cálculo de la función costo correspondiente a cada etapa.

| # | Etapas Previa | Costo |
|----|--------------------------------|-----------------------------|
| 1 | $(p - 1, q - 1, r - 1, s - 1)$ | $DTW(R1, R2) + DTW(C1, C2)$ |
| 2 | $(p - 1, q - 1, r - 1, s)$ | $DTW(R1, R2) + DTW(C1, C2)$ |
| 3 | $(p - 1, q - 1, r, s - 1)$ | $DTW(R1, R2) + DTW(C1, C2)$ |
| 4 | $(p - 1, q - 1, r, s)$ | $DTW(R1, R2) + DTW(C1, C2)$ |
| 5 | $(p - 1, q, r - 1, s - 1)$ | $DTW(R1, R2) + DTW(C1, C2)$ |
| 6 | $(p - 1, q, r - 1, s)$ | $DTW(R1, R2)$ |
| 7 | $(p - 1, q, r, s - 1)$ | $DTW(R1, R2) + DTW(C1, C2)$ |
| 8 | $(p - 1, q, r, s)$ | $DTW(R1, R2)$ |
| 9 | $(p, q - 1, r - 1, s - 1)$ | $DTW(R1, R2) + DTW(C1, C2)$ |
| 10 | $(p, q - 1, r - 1, s)$ | $DTW(R1, R2) + DTW(C1, C2)$ |
| 11 | $(p, q - 1, r, s - 1)$ | $DTW(C1, C2)$ |
| 12 | $(p, q - 1, r, s)$ | $DTW(C1, C2)$ |
| 13 | $(p, q, r - 1, s - 1)$ | $DTW(R1, R2) + DTW(C1, C2)$ |
| 14 | $(p, q, r - 1, s)$ | $DTW(R1, R2)$ |
| 15 | $(p, q, r, s - 1)$ | $DTW(C1, C2)$ |

La función costo consiste en utilizar DTW sobre los índices en filas y columnas de A y B . En la Tabla 1, $R1$ es la fila en la coordenada (p, q) , los valores de las casillas que se encuentran entre la posición 0 a la columna q , sobre la fila p . $C1$ es la columna en la coordenada (p, q) , los valores de las casillas que están entre la posición 0 a la fila p , sobre la columna q . De manera similar se obtienen $R2$ y $C2$. Ver Fig. 3. Para determinar el valor de la casilla $M(p, q, r, s)$ se debe obtener un costo para cada una de las etapas previas, que está dado por la alineación de las filas y columnas correspondientes en A y B , es decir, dada la posición $M(p, q, r, s)$ se obtienen $R1$ y $R2$, así como $C1$ y $C2$.

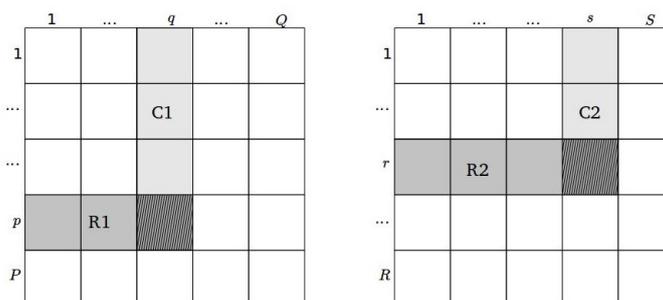


Fig. 3. Casillas que seleccionadas de cada matriz A y B respectivamente.

El algoritmo 1 es utilizado para calcular cada una de las etapas previas de la Tabla 1. La función costo se puede ir obteniendo conforme avanza la

iteración de las etapas actuales. Este proceso debe ser realizado para cada una de las coordenadas de M . Una manera eficiente de realizar este procedimiento es mediante el uso de Programación Dinámica; se debe hacer un recorrido de alta complejidad que es representado en el algoritmo.

Algoritmo 1 Algoritmo para calcular la distancia Fréchet moderno

Entrada: Matrices $A(p, q)$, $B(r, s)$ de datos numéricos.

Salida: Número escalar k , indica la distancia entre A y B.

```

1:  $P \leftarrow$  Filas de  $A$ 
2:  $Q \leftarrow$  Columnas de  $A$ 
3:  $R \leftarrow$  Filas de  $B$ 
4:  $S \leftarrow$  Columnas de  $B$ 
5:  $d(P, Q, R, S)$  Matriz de 4 índices, inicializada en  $\infty$ 
6: para  $p = 2$  hasta  $P$  hacer
7:   para  $q = 2$  hasta  $Q$  hacer
8:     para  $r = 2$  hasta  $R$  hacer
9:       para  $s = 2$  hasta  $S$  hacer
10:          $d(p, q, r, s) = \min \left\{ \begin{array}{l} DTW(R1, R2) + DTW(C1, C2) + d(p-1, q-1, r-1, s-1) \\ DTW(R1, R2) + DTW(C1, C2) + d(p-1, q-1, r-1, s) \\ DTW(R1, R2) + DTW(C1, C2) + d(p-1, q-1, r, s-1) \\ DTW(R1, R2) + DTW(C1, C2) + d(p-1, q-1, r, s) \\ DTW(R1, R2) + DTW(C1, C2) + d(p-1, q, r-1, s-1) \\ DTW(R1, R2) + DTW(C1, C2) + d(p-1, q, r-1, s) \\ DTW(R1, R2) + DTW(C1, C2) + d(p-1, q, r, s-1) \\ DTW(R1, R2) + DTW(C1, C2) + d(p-1, q, r, s) \\ DTW(R1, R2) + DTW(C1, C2) + d(p, q-1, r-1, s-1) \\ DTW(R1, R2) + DTW(C1, C2) + d(p, q-1, r-1, s) \\ DTW(C1, C2) + d(p, q-1, r, s-1) \\ DTW(C1, C2) + d(p, q-1, r, s) \\ DTW(R1, R2) + DTW(C1, C2) + d(p, q, r-1, s-1) \\ DTW(R1, R2) + d(p, q, r-1, s) \\ DTW(C1, C2) + d(p, q, r, s-1) \end{array} \right.$ 
11:       fin para
12:     fin para
13:   fin para
14: fin para
15: devolver  $k = d(P, Q, R, S)$ 

```

4. Desarrollo

La técnica recibe como entrada dos arreglos bidimensionales de la misma clase, en este caso se utilizan matrices de datos de funciones matemáticas previamente creadas y se obtiene como resultado un indicador de distancia que representa la diferencia entre los objetos de entrada, éste es un número escalar. El método de comparación responde a las propiedades de métrica [3].

Se presenta el procedimiento que se utilizó para la comparación matricial de datos numéricos. Éste es aplicado a datos generados artificialmente a partir de funciones matemáticas, pero también puede ser aplicada a cualquier fenómeno que se pueda representar en matrices, como por ejemplo grafos, ontologías, entre otros.

Creamos un *corpus* de matrices de datos numéricos provenientes de funciones matemáticas, se toma una función y se varía un valor t de manera incremental, así se produce una matriz por cada valor de t que es variado. Se repite el proceso para

las funciones que se desean comparar. Posteriormente se realizan experimentos de Búsqueda, donde se escoge una matriz a buscar (*query*) y se proporciona un directorio con varias matrices. En el experimento, nuestro algoritmo obtiene las matrices con distancia mínima con respecto a la matriz *query* que se escogió. Una muestra del conjunto de matrices se muestra en la Fig. 4. La muestra contiene tres categorías y cada categoría tiene cuatro matrices.

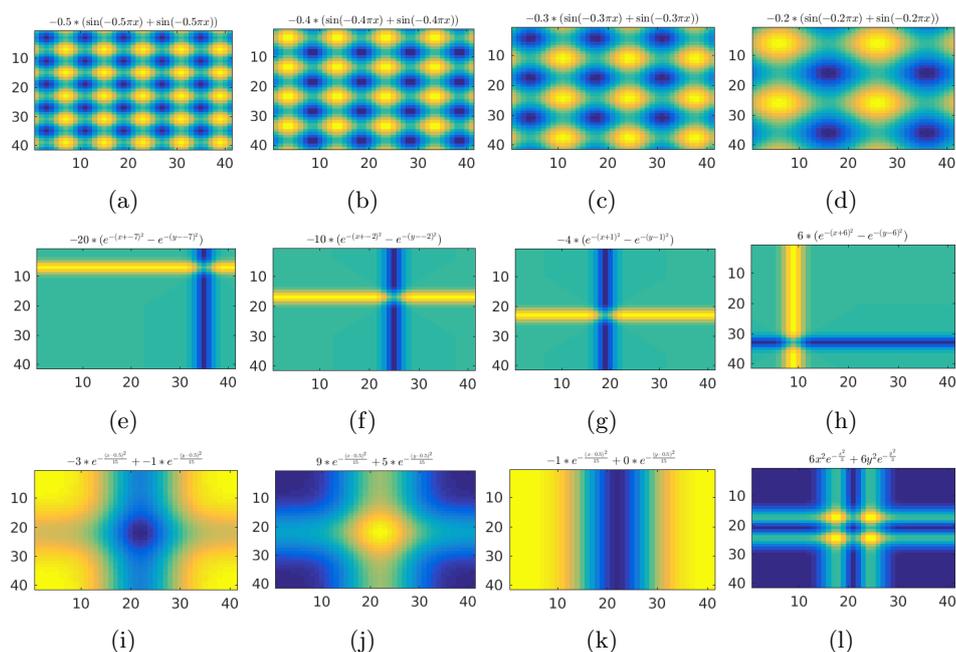


Fig. 4. Algunas matrices del corpus.

Las matrices fueron creadas por cada función; entonces es posible agruparlas como si fueran categorías. Se utilizó la técnica de distancia Fréchet Moderno para realizar la experimentación: primero se realizaron comparaciones entre elementos de las mismas categorías, posteriormente se realizaron comparaciones haciendo una mezcla de diferentes categorías. De estas matrices se selecciona una que será el *query*:

1. Se realiza la comparación dentro de la misma categoría que la del *query*, las funciones evaluadas para la categoría están en la Tabla 2, se observa el valor de la constante es variado un decimal. Los resultados se muestran en la Fig. 5, vemos la función que corresponde al *query*: $[-0.2(\sin(-0.2\pi x) + \sin(-0.2\pi x))]$, las mas cercanas son las que tienen el valor pequeño en las constantes de las funciones evaluadas.

Tabla 2. Matrices de la misma función.

| ID | Cercanas |
|--------|--|
| [ID_9] | $-0.2 * (\sin(-0.2\pi x) + \sin(-0.2\pi x))$ |
| [ID_8] | $-0.3 * (\sin(-0.3\pi x) + \sin(-0.3\pi x))$ |
| [ID_7] | $-0.4 * (\sin(-0.4\pi x) + \sin(-0.4\pi x))$ |
| [ID_6] | $-0.5 * (\sin(-0.5\pi x) + \sin(-0.5\pi x))$ |

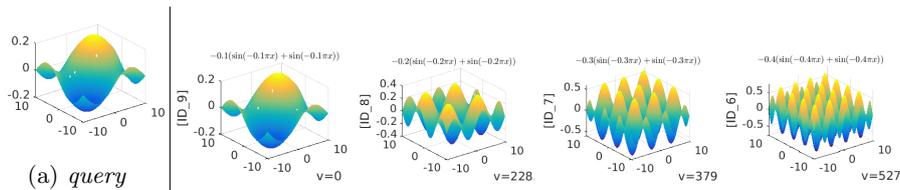


Fig. 5. Resultados de la búsqueda utilizando Fréchet Moderno. Se muestran las matrices con las distancias más cercanas al *query*, dentro de la misma categoría.

- Se realiza una comparación considerando matrices de otras categorías. La Fig. 6 muestra en la primera columna la matriz *query* seleccionada, mientras que las otras tres columnas son las que corresponden a las matrices más cercanas. Se muestra en la parte superior la función que se evaluó para obtener esa matriz. Se puede observar que las matrices cercanas corresponden a las matrices de la misma categoría del *query*.

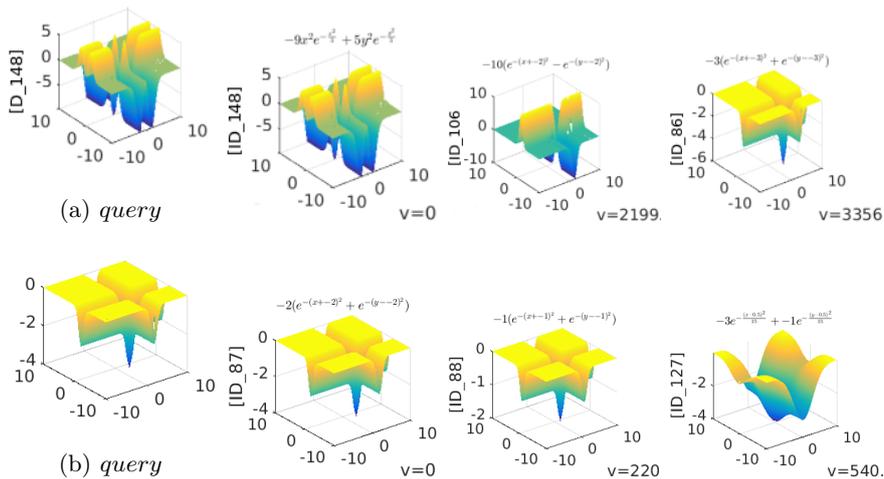


Fig. 6. Experimento donde se muestra la matriz *query*, se realiza la búsqueda en el directorio de matrices.

5. Resultados y conclusiones

Ordenando de menor a mayor el indicador que se obtuvo del método Fréchet Moderno, observamos que los números más pequeños son aquellos que están más cercanos (desde el punto de vista técnico) a la matriz *query*, es decir, las matrices más similares; de esta manera el *query* comparado contra ella misma da como resultado un cero, lo cual indica que son la misma matriz y no hay diferencia. Esto corresponde a las propiedades de las métricas.

La técnica propuesta de este trabajo es una analogía a la distancia de Fréchet, donde Fréchet ocupa una correa para conectar a un perro con su amo, y cada uno hace un recorrido. De este modo la Distancia de Fréchet Moderno también realiza un recorrido, pero es a través de datos en R^2 , donde también se tiene una “correa” en cada celda visitada y va midiendo la diferencia en cada una como se mostró en la Fig. 1b. Fréchet Moderno ha resultado una técnica para la comparación de matrices de datos numéricos que es sencilla en comparación con la técnica utilizada en [6], pero es poderosa en el sentido de que considera todos los datos de las matrices. Es importante mencionar que esta técnica puede ser utilizada en muchas más situaciones, se pueden realizar experimentos con imágenes, grafos, ontologías, etc.

Referencias

1. Agarwal, P.K., Rinat, A.B., Kaplan, H., Sharir, M.: Computing the Discrete Fréchet Distance in Subquadratic Time. CoRR abs/1204.5, pp. 1–18 (2012)
2. Angeles-Yreta, A., Figueroa-Nazuno, J., Ramírez-Amaro, K.: Búsqueda de Similitud entre Objetos 3D por Indexado. In: Reunión de Otoño Comunicación, Computación Electrónica y Exposición Industrial, ROC&C, IEEE Sección México, Acapulco, Guerrero. pp. 112–117 (2005)
3. Deza, M.M., Deza, E.: Encyclopedia of Distances, vol. 3. Springer, 3 edn. (2009)
4. Figueroa-Nazuno, J., Angeles-Yreta, A., Medina-Apodaca, J., Ortega-González, V., Ramírez-Amaro, K., Mirón-Bernal, M., Landassuri-Moreno, V.: Sobre el problema de Similitud. Tech. rep., Centro de Investigación en Computación. Instituto Politécnico Nacional Unidad Profesional “Adolfo López Mateos”, Zacatenco, México, DF (2008)
5. Fréchet, M.M.: Sur quelques points du calcul fonctionnel. Rend. del Circ. Mat. di Palermo 22(1), pp. 1–72 (1906)
6. González-Ortega, E.: Una Técnica para el Análisis de Similitud entre Imágenes. Master’s thesis, Centro de Investigación en Computación, Instituto Politécnico Nacional, México (2013)
7. Keysers, D., Unger, W.: Elastic image matching is NP-complete. Pattern Recognition. Letters. 24(1-3), pp. 445–453 (2003)
8. Pitcher, A.D., Chittenden, E.W.: On the Foundations of the Calcul Fonctionnel of Fréchet. Trans. Am. Math. Soc. 19(1), pp. 66–78 (1918)
9. Rico-Martínez, J.: Determinación Numérica de Eigenvalores y Eigenvectores. Tech. rep., ICT 224, Departamento de Ingeniería Mecánica, Universidad de Guanajuato, F.I.M.E.E., Mexico (2003)
10. Uchida, S., Sakoe, H.: A survey of elastic matching techniques for handwritten character recognition. IEICE Trans. Inf. Syst. E88-D(8), pp. 1781–1790 (2005)

Algoritmos del muestreo/reconstrucción para realizaciones de procesos Gaussianos con espectros rectangulares

Vladimir Kazakov, Francisco Mendoza Sánchez

Instituto Politécnico Nacional, Depto. Telecomunicaciones,
Unidad Zacatenco, Ciudad de México, México

vkaz41@hotmail.com, fcm2709@gmail.com

Resumen. Con base en la regla de la esperanza matemática condicional se investiga el procedimiento de muestreo-reconstrucción para procesos Gaussianos con un espectro rectangular. Se estudian dos variantes para este tipo de espectro: 1) cuando el espectro está en el origen y 2) al estar desplazado. En ambos casos se reportan los resultados obtenidos tanto en las funciones básicas y en las funciones de error de reconstrucción cuando el intervalo de muestreo y el número de muestras son arbitrarios.

Palabras clave: Proceso Gaussiano con espectro rectangular, función básica, error de reconstrucción, número de muestras finitas e intervalos de muestreos arbitrarios.

Algorithms of the Sample/Reconstruction for Realizations of Gaussian Process with Rectangular Spectrum

Abstract. On the basis of the conditional mathematical expectation rule the procedure sampling - reconstruction (PMR) is investigated for the sampling process is Gaussian with a rectangular spectrum: when the spectrum is 1) at origin and 2) displaced; in both cases, they studied the basics functions and functions of reconstruction error, when when the number of samples and interval samples are arbitrary.

Keywords: Gaussian process with rectangular spectrum, basic function, function of reconstruction error, number of samples and interval samples are arbitrary.

1. Introducción

La descripción del Procedimiento de Muestreo-Reconstrucción (PMR) de las realizaciones en un proceso aleatorio ha sido investigado durante muchos años (ver por ejemplo dos revisiones [1,2]). Según el teorema de Balakrishnan (TB) [3] se puede reconstruir cualquier realización de un proceso aleatorio estacionario con un espectro de potencia restringido a una frecuencia ω_b usando la siguiente expresión:

$$x(t) = \lim_{N \rightarrow \infty} \sum_{i=-N}^N (x(T_i)) \frac{\text{sen}\omega_b(t - iT_b)}{\omega_b(t - iT_b)}, \quad (1)$$

donde $x(T_i)$ la muestra de una realización en el instante T_i ; ΔT es un intervalo de muestreo *periódico* entre las muestras vecinas:

$$T_b = T_i - T_{i-1} = \frac{\pi}{\omega_b}. \quad (2)$$

Enfatizamos que en TB el error es cero si y solo si el número de las muestras es *infinito*. En [5,6] fue probado que TB es válido solamente para procesos Gaussianos.

El algoritmo (1) se caracteriza por tener un error de reconstrucción igual a cero para todos los procesos independientemente de su función de densidad de probabilidad [4]. Introducimos la función básica $b_i(t)$. Como se ve en (1) todas las muestras tienen la misma función básica del tipo $\frac{\text{sen}(x)}{x}$:

$$b_i(t) = \frac{\text{sen}\omega_b(t - iT_b)}{\omega_b(t - iT_b)}. \quad (3)$$

Notamos, que el algoritmo (1) y la función básica (3) ignoran tales características de gran importancia como: la función de densidad de probabilidad (*fdp*), y la función de covarianza del proceso.

El modelo matemático de procesos aleatorios con espectro restringido es citado con mucha frecuencia en la literatura. Pero TB no da ninguna información acerca de la de la influencia para PMR cuando el número N de muestras son arbitrarias. El artículo presente da la claridad dentro de dichos problemas, porque para cada variante se obtienen dos características más importantes para cada PMR: la función básica y del error de reconstrucción. Estos resultados pueden ser útiles para los investigadores de los sistemas de comunicación con mensajes gaussianos con espectros rectangulares.

La meta del artículo presente es investigar el problema PMR de las realizaciones de procesos Gaussianos con un espectro rectangular, centrado en el origen y al encontrarse desplazado, cuando ΔT y N son arbitrarios. El conocimiento de la forma del espectro nos da la posibilidad de conocer la función de covarianza del proceso dado. Para cumplir este análisis usamos la metodología de la regla de esperanza matemática condicional [7]. La aplicación de dicha regla para la descripción de PMR de realizaciones de procesos aleatorios de varios tipos, fue descrita en algunas publicaciones [5,6], por mencionar algunos. En la investigación de PMR hay dos características principales: 1) la función óptima de reconstrucción; 2) la función mínima del error de la reconstrucción.

2. La regla de la esperanza matemática condicional en PMR de realizaciones gaussianas

Cada proceso gaussiano no estacionario esta descrito con tres características principales: la esperanza matemática $m(t)$, la varianza $\sigma^2(t)$ y la función de covarianza $K(t_1, t_2)$. Elegimos una realización del proceso y fijamos una multitud de las muestras $X, T = \{X(T_1), X(T_2), \dots, x(T_N)\}$. En esta multitud el número de muestras N y la locación de la muestras $T = \{T_1, T_2, \dots, T_N\}$ son arbitrarios. Usando X, T las características a priori $m(t), \sigma^2(t), K(t_1, t_2)$ se puede obtener las características del proceso condicional $x(t)|X, T = \tilde{x}(t)$, es decir, la esperanza matemática condicional $\langle x(t)|X, T \rangle = \tilde{m}(t)$, y la varianza condicional $\langle [\tilde{x}(t) - \tilde{m}(t)]^2 | X, T \rangle = \tilde{\sigma}^2(t)$. La función $\tilde{m}(t)$ es la función de reconstrucción de la realización muestreada y la función $\tilde{\sigma}^2(t)$ es la función de error de reconstrucción. La función $\tilde{m}(t)$ es la mejor estimación de la realización en el instante de tiempo actual t y la función $\tilde{\sigma}^2(t)$ es el error mínimo de reconstrucción de dicha realización en el mismo instante de tiempo t . Las características principales de PMR son $\tilde{m}(t)$ y $\tilde{\sigma}^2(t)$ pueden ser descritas según [8]:

Considerando el caso estacionario para $\tilde{m}(t)$ y $\tilde{\sigma}^2(t)$; $m(t) = m = 0, \sigma^2(t) = \sigma = 1, K(t_1, t_2) = K(t_1 - t_2)$, tienen las formulas siguientes:

$$\tilde{m}(t) = \sum_i = 1^N \left(\sum_j = 1^N (K(t - T_i) a_{ij} x(T_j)) \right), \quad (4)$$

$$\tilde{\sigma}^2(t) = 1 - \sum_i = 1^N \left(\sum_j = 1^N (K(t - T_i) a_{ij} K(T_j - t)) \right), \quad (5)$$

$$\|K(T_i - T_j)\| = \left\| \begin{matrix} K(T_1 - T_1) \dots K(T_1 - T_N) \\ \dots \end{matrix} \right\|, \quad (6)$$

$$\|a_{ij}\| = \|K(T_i - T_j)\|. \quad (7)$$

La expresión (4) se reescribe en forma simplificada

$$\tilde{m}(t) = \sum_{j=1}^N (x(T_j) b_j(t)), \quad (8)$$

aquí es una función básica determinada por la formula

$$b_j(t) = \sum_j = 1^N (k(t - T_i) a_{ij}). \quad (9)$$

Cada muestra tiene su propia función $b_j(t)$. En (3) y (9) vemos que generalmente las funciones básicas son diferentes en TB y en el algoritmo fundado en la regla de esperanza matemática condicional. Para comparar correctamente con TB vamos a considerar el caso cuando el intervalo de muestreo es periódico. El proceso investigado es obtenido con un filtro ideal afectado con ruido blanco

gaussiano con densidad espectral $\frac{N_0}{2}$. La característica de tal filtro en el dominio de la frecuencia es

$$K(j\omega) = \begin{cases} \text{si } \omega \leq \omega_b & K_0 \\ \text{si } \omega > \omega_b & 0 \end{cases}. \quad (10)$$

Entonces, el espectro de la salida será

$$K(j\omega) = \frac{N_0}{2} |K(j\omega)|^2. \quad (11)$$

Según el teorema Wiener - Khinchin usando (11) tenemos la expresión para la función de covarianza

$$K(\tau) = \sigma^2 \frac{\text{sen}\omega_b\tau}{\omega_b\tau} = \frac{\text{sen}\tau/T_b}{\tau/T_b}, \quad (12)$$

donde

$$\sigma^2 = N_0 K_0^2 \frac{\omega_b}{2\pi}. \quad (13)$$

Considerando un proceso estacionario $\sigma^2 = 1$. Es la función $\text{sen}(x)/x$ la cual tiene sus ceros o puntos de cruce con respecto al eje τ ocurren cada nT_b , donde $n = \{1, 2, 3, \dots\}$.

Este proceso tiene una función de covarianza del tipo $\text{sen}(x)/x$. Cuando $\Delta T = T_b$ en estos puntos las muestras son *independientes*. Esta propiedad es muy importante, porque en (1) las muestras localizadas en tales instantes, no tienen conexiones estadísticas entre ellas. Pero la meta del presente artículo es investigar la influencia sobre PMR, cuando el intervalo $\Delta T \leq T_b$ y cuando el número N de muestras es arbitrario, y mostrar el efecto que existe sobre PMR al desplazar el espectro. Por ello, es necesario describir PMR de las realizaciones del proceso considerando como principio la regla de esperanza matemática condicional. En este caso primeramente consideremos las funciones básicas. Usando (9), (11) y recordando $\sigma^2 = 1$, tenemos

$$b_j(t) = \sum_{i=1}^N \frac{\text{sen}\omega_b(t - T_i)}{\omega_b(t - T_i)} a_{ij}. \quad (14)$$

Entonces, las características principales (4), (5) de PMR se describen con siguientes formulas:

$$\tilde{m}(t) = \sum_{j=1}^N x(T_j) \sum_{i=1}^N \frac{\text{sen}\omega_b(t - T_i)}{\omega_b(t - T_i)} a_{ij}, \quad (15)$$

$$\tilde{\sigma}^2(t) = 1 - \sum_{i=1}^N \frac{\text{sen}\omega_b(t - T_i)}{\omega_b(t - T_i)} a_{ij} \frac{\omega_b(T_j - t)}{\omega_b(T_j - t)}. \quad (16)$$

Abajo vamos a investigar las funciones básica $b_j(t)$ en lugar de las funciones de reconstrucción $\tilde{m}(t)$, porque las ultimas dependen de muestras concretas $x(T_i)$, donde $i = \{1, 2, 3, \dots\}$.

El caso anterior muestra cuando el espectro está en el origen, sin embargo al desplazar el espectro a una frecuencia ω_0 , esta definida en (17).

$$K(j\omega) = \begin{cases} \text{si } \omega_0 - \omega_b \leq \omega \leq \omega_0 + \omega_b & K_0 \\ \text{si } \omega < \omega_0 - \omega_b \text{ o } \omega > \omega_0 + \omega_b & 0 \end{cases} \quad (17)$$

es la frecuencia central del espectro, mientras ω_b es la mitad del ancho de banda del espectro, estos parámetros están definidos en el tiempo, considerando $\omega_b = \frac{\pi}{T_b}$ y $\omega_0 = \frac{2\pi}{T_0}$. De este modo queda definida la función de covarianza (18) con un espectro desplazado del origen.

$$K(\tau) = \sigma^2 \frac{\text{sen}(\Delta\omega_b\tau/2)}{\Delta\omega_b\tau/2} \cos(\omega_0\tau) = \frac{\pi\tau/T_b}{\pi\tau/T_b} \cos(2\pi\tau/T_0), \quad (18)$$

donde $\sigma^2 = 1$ en la segunda parte de la ecuación (18), en Fig. 1 se muestra la forma de onda de (18), queda de manifiesto que el comportamiento de la función de covarianza corresponde a las propiedades del $\cos(x)$, donde los ceros de la función están determinado por el parámetro T_0 , pero modulada por $\text{sen}(x)/x$.

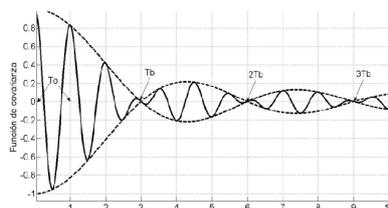


Fig. 1. Función de covarianza con espectro desplazado del origen.

Por tanto, se reescriben las ecuaciones (14)-(16) en términos de la función de covarianza (18). Función básica con espectro rectangular desplazado del origen (19):

$$b_j(t) = \sum_{i=1}^N \left(\frac{\text{sen}\pi(t - T_i)/T_b}{\pi(t - T_i)/T_b} \cos(2\pi(t - T_i)/T_0) a_{ij} \right). \quad (19)$$

Función de reconstrucción de con espectro desplazado del origen (20):

$$\tilde{m}(t) = \sum_{j=1}^N \left(\sum_{i=1}^N (x(T_j) \frac{\text{sen}\pi(t - T_i)/T_b}{\pi(t - T_i)/T_b} \cos(2\pi(t - T_i)/T_0) a_{ij} \right). \quad (20)$$

Función de error de reconstrucción para un proceso con espectro rectangular desplazado del origen (21):

$$\tilde{\sigma}(t) = 1 - \sum_{j=1}^N \left(\sum_{i=1}^N \left(\frac{\text{sen}\pi(t - T_i)/T_b}{\pi(t - T_i)/T_b} \cos(2\pi(t - T_i)/T_0) a_{ij} \frac{\text{sen}\pi(T_j - t)/T_b}{\pi(T_j - t)/T_b} \cos(2\pi(T_j - t)/T_0) \right) \right). \quad (21)$$

3. Resultados obtenidos cuando el intervalo dentro de muestras y el número de muestras son arbitrarios

Usando la función de covarianza (12), en la PMR, y para simplificar, elegimos en (2) $\omega_b = \pi$, obtenemos $T_b = 1$. Obtenemos una variante a la que llamamos *clásica*. Sin embargo esta variante presenta independencia estadística en cada instante de muestreo. En dicho caso, la matriz de covarianza queda de la forma (22):

$$\|K(T_i - T_j)\| = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix}. \quad (22)$$

Esta matriz demuestra formalmente la independencia de todas las muestras. Dicho efecto influye sobre las ecuaciones (14) - (16).

Considerando (21) las ecuaciones se reducen, por ello la función básica (14) se reduce a (23):

$$b_j(t) = \frac{\text{sen}\omega(t - T_i)}{\omega(t - T_i)}. \quad (23)$$

Vemos que la función básica no depende del número N de muestras participantes en la operación de reconstrucción. De esta manera se reescribe $\tilde{m}(t)$, para este tipo de función de covarianza (12), como se muestra en (24):

$$\tilde{m}(t) = \sum_{i=1}^N (x(T_j) \frac{\text{sen}\omega_b(t - T_i)}{\omega_b(t - T_i)}). \quad (24)$$

En (25) se observa que hemos llegado casi al mismo resultado como en TB (1). La diferencia es: en (24) no hay un número infinito de muestras. En (24) se puede sustituir $N \rightarrow \infty$:

$$\tilde{m}(t) = \sum_{i=-\infty}^{\infty} (x(T_j) \frac{\text{sen}\omega_b(t - T_i)}{\omega_b(t - T_i)}). \quad (25)$$

Entonces, vamos a tener una coincidencia completa entre TB (1) y el algoritmo (25). Este resultado significa que la función de reconstrucción (1) es un caso

particular del algoritmo general (4), fundado en la regla de la esperanza matemática condicional. Notamos, que la función de reconstrucción óptima guarda su forma cuando el número N de las muestras es infinito. Y por último $\tilde{\sigma}^2(t)$ queda definida en (26):

$$\tilde{\sigma}^2(t) = 1 - \sum_{i=1}^N \sum_{j=1}^N \frac{\text{sen}\omega_b(t - T_i)}{\omega_b(t - T_i)} \frac{\text{sen}\omega_b(T_j - t)}{\omega_b(T_j - t)}. \quad (26)$$

Para ejemplificar, supondremos una $\Delta T = T_b$, por tanto obtendremos una matriz identidad (22) para su matriz inversa de covarianza, lo que significa la independencia estadística entre las muestras. Ahora bien para ilustrar PMR utilizando este intervalo se muestra tanto las funciones básicas clásicas del tipo $\text{sen}(x)/x$ y la función de error de reconstrucción Fig. 2, para ello se realizan la calculaciones correspondientes usando (23) - (25).

Se observa que la forma de onda de las funciones básicas son del tipo $\text{sen}(x)/x$, lo que muestra de manera clara la independencia que existe en la muestras a causa que en $T_b = 1$ hay un cruce en cero de la función de covarianza, y por tanto cada muestra coincide en cero de la función de covarianza.

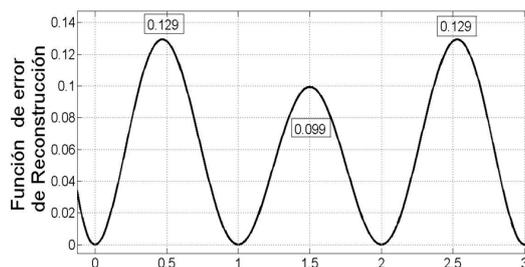


Fig. 2. Función de error de reconstrucción con $\Delta T = 1, T_b = 1N = 4$.

Sin embargo, cuando $\Delta T < T_b$ obtenemos resultados diferentes al caso anterior, pues $a_{i,j}$ deja de ser una matriz identidad. Esto significa que existe una dependencia estadística durante el proceso de reconstrucción de las muestras no vecinas según (7).

Consideramos el PMR de un proceso con $N = 4y\Delta T = 0.4$, de este modo se mantiene la relación $\Delta T < T_b$, evitando que una muestra coincida en algún instante con los valores múltiplos de T_b , con el objetivo de ver la dependencia estadística entre las muestras se presenta en (27) la matriz inversa de covarianza

$$\|a_{ij}\| = \|K(T_i - T_j)\|^{-1} = \begin{pmatrix} 17.73 & -35.36 & 33.38 & -14.23 \\ -35.36 & 76.82 & -75.14 & 33.38 \\ 33.38 & -75.14 & 76.82 & -35.36 \\ -14.23 & 33.38 & -35.36 & 17.73 \end{pmatrix}. \quad (27)$$

Con el uso de (14) se obtienen las funciones básicas observadas en Fig. 3. En donde se destaca que las formas de onda son diferentes a las formas de onda $sen(x)/x$, este efecto es debido a la dependencia estadística existente, es necesario enfatizar que la cuarta muestra presenta un tiempo de covarianza negativo; sin embargo esto es indiferente en la reconstrucción del proceso aleatorio.

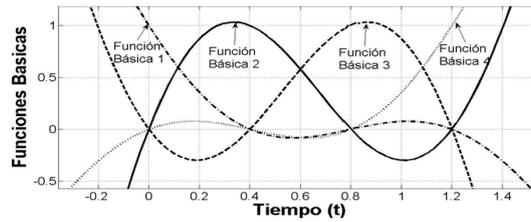


Fig. 3. Funciones básicas con $\Delta T = 0.4, T_b = 1N = 4$.

Utilizando (16) se obtiene la función de error de reconstrucción mostrada en Fig.4. La función de reconstrucción al usar este intervalo presenta menores valores de error que con respecto al caso cuando los instantes de muestreo coinciden con los ceros de la función de covarianza, es decir, cuando $\Delta T = T_b$, lo que hace ver la limitación de TB en cuanto a considerar un cifra específica en la reconstrucción en PMR.

El aumento en la función de error es debido a la longitud del intervalo, es decir, a mayor el intervalo mayor el error de reconstrucción.

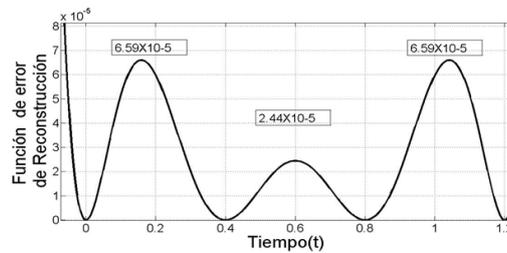


Fig. 4. Función de error de reconstrucción con $\Delta T = 0.4, T_b = 1N = 4$.

4. Resultados obtenidos cuando espectro rectangular de un proceso es desplazado del origen

En Fig. 1, se observa el comportamiento de la función de covarianza cuando el espectro es desplazado del origen y centrado en ω_0 (17), únicamente se reportara

el caso cuando $T_0 < T_b$, pues presenta una forma de onda semejante $\text{sen}(x)/x$, sin embargo el comportamiento en los ceros de la función la determina la parte del $\text{cos}(x)$ mediante el parámetro T_0 y la tendencia está controlada por $\text{sen}(x)/x$.

Para observar el comportamiento análogo de la función de covarianza (12) y (18), se presenta dos casos: 1) la existente independencia estadística entre las muestras, 2) cuando los instantes de muestreo no coinciden con los ceros de la función de covarianza (18). El primer caso ocurre cuando $\Delta T = 0.25T_0$, donde se obtiene una matriz identidad (21) en la matriz inversa de covarianza, por tanto (20) se reescribe en forma de (28):

$$b_j(t) = \frac{\text{sen}\pi(t - T_i)/T_b}{\pi(t - T_i)/T_b} \cos \frac{2\pi(t - T_i)}{T_0}. \quad (28)$$

Al igual, se simplifica la formula (21) en forma de (29):

$$\tilde{\sigma}^2 = 1 - \sum_{i=1}^N \sum_{j=1}^N \frac{\text{sen}\pi(t - T_i)/T_b}{\pi(t - T_i)/T_b} \cos \frac{2\pi(t - T_i)/T_b}{T_0} \frac{\text{sen}\pi(T_j - t)/T_b}{\pi(T_j - t)/T_b} \cos \frac{2\pi(T_j - t)/T_b}{T_0}. \quad (29)$$

Para ilustrar los resultados, suponiendo $\Delta T = 0.25, T_b = 3, N = 4$, observamos que en Fig. 5 las funciones básicas presentan la forma de onda de la función de covarianza (18) y también que es distinta la forma de onda $\text{sen}(x)/x$ a pesar, que en ambos casos ponen de manifiesto la independencia estadística entre las muestras.

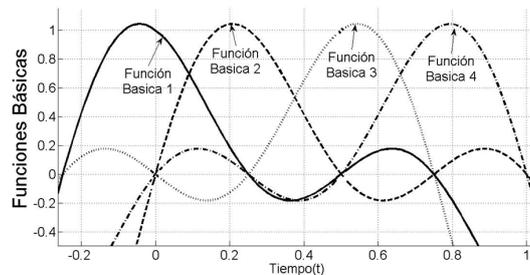


Fig. 5. Funciones básicas con $\Delta T = 0.25, T_0 = 1, T_b = 3N = 4$.

La función de error de reconstrucción mostrada en Fig. 5 se observan valores menores en la función de error que en comparación con la Fig. 2. Cuando se consideran intervalos de muestreo, donde se evite que en algún instante del muestreo coincida con los ceros de la función de covarianza, significa la presencia de dependencia estadística entre las muestras, lo que, también, implica que la matriz de covarianza inversa tenga la forma distinta a (21), y dependiendo de

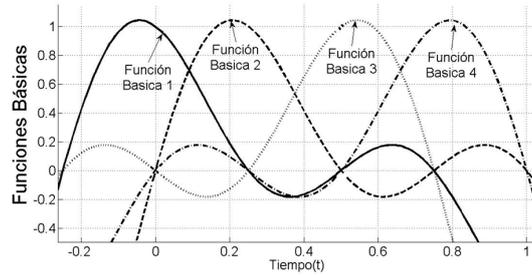


Fig. 6. Función de error de reconstrucción con $\Delta T = 0.25, T_0 = 1, T_b = 3N = 4$.

número N de muestras se llegan a tomar valores negativos de la función de covarianza, por todo ello, es necesario considerar que para el cálculo de la función de reconstrucción y la función de error de reconstrucción las ecuaciones (19) - (21) sin modificaciones.

Para ilustrar este hecho supongamos $\Delta T = 0.4, T_0 = 1, T_b = 3; N = 4$, de este modo se evita que los instantes de muestreo coincidan con los ceros de la función de covarianza, por tanto las muestras presentan dependencia estadísticas y se usan las ecuaciones (19) - (21). Usando la (24) se obtiene las funciones básicas en Fig.7. Donde se ve claramente que las formas de onda de estas funciones básicas de la Fig. 5 y Fig. 7 son diferentes, debido a que en el primer caso no existe dependencia estadística en comparación con el caso de la Fig. 7.

También se observa como cada función básica contribuye de manera mínima en la reconstrucción del proceso aleatorio de modo diferente en Fig. 5 y Fig. 7.

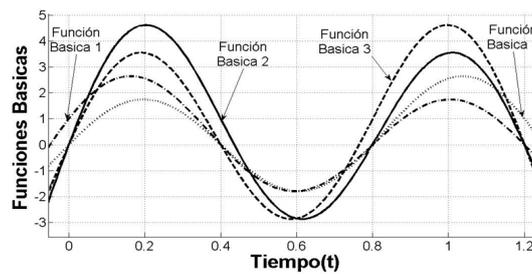


Fig. 7. Funciones Básicas $\Delta T = 0.4, T_0 = 1, T_b = 3N = 4$.

Mientras utilizando (26) resultan valores mayores en la función de error de reconstrucción que con respecto a la Fig. 8, debido al aumento del intervalo de muestreo el error es mayor que en comparación con la Fig. 6.

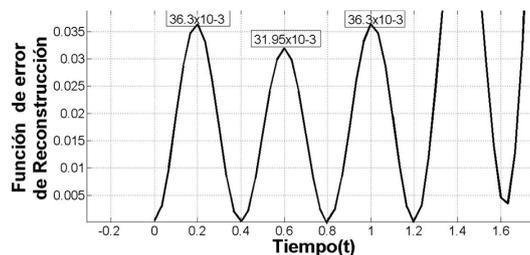


Fig. 8. Función de error de reconstrucción con $\Delta T = 0.4, T_0 = 1, T_b = 3N = 4$.

5. Conclusiones

Utilizando la regla de la esperanza matemática condicional, como metodología de la investigación, para el cálculo de errores de reconstrucción cuando el número de muestras es limitado y cuando intervalos de muestreo son arbitrarios, en un proceso Gaussiano con espectro rectangular para cuando se encuentra en el origen y al estar desplazado. Se reportan los resultados en PMR al utilizar segmentos con valores negativos de la función de covarianza.

Por último, al evaluar la función de covarianza de un espectro rectangular desplazado del origen, se observó que el valor del error en las gráficas Fig. 8. y Fig. 10 de la función de error de reconstrucción es mayor en relación con los errores reportados al usar la función de covarianza del espectro centrado en el origen, debido a la influencia del aumento del intervalo de muestreo.

Referencias

1. Jerry, A.: The Shannon sampling theorem: Its various extensions and applications. A tutorial review, Proc. IEEE, vol. 65, pp. 1565–1596 (1977)
2. Jerry, A.: Bibliographic List. In Advanced Topics in Shannon sampling and interpolation theory, Springer, N.Y (1992)
3. Balakrishnan, A.: A Note on the Sampling Principle for Continuous Signals. IRE Trans, On informations theory, Vol. IT-3, pp. 143–146 (1957)
4. Balakrishnan, A.: On the Problem of Time Jitter in Sampling. IRE Trans, On information theory, Vol. IT-3, pp. 226–236 (1962)
5. Kazakov, V.: The sampling-reconstruction procedure with a limited number of samples of stochastic processes and fields on the basis of the conditional mean rule. Electromagnetic waves and electronic system, Vol. 10, Num. 1-2, pp. 98–116 (2005)
6. Kazakov, V.: Sampling-Reconstruction procedures of Gaussian process realizations. Chapter 9 in the book: Probability: interpretation, theory and applications, Nova science publisher inc., USA, NY., pp. 269–297 (2012)
7. Pfeifer, P.E.: Probability for applications. Springer Verlag (1990)
8. Stratonovich, R.: Topics in the theory of Random Noise. Vol I, Gordon and Breach, N.Y., London (1963)

White-donkey: búsqueda de personas con vehículos aéreos no tripulados basada en visión por computadora

Jesús Jaime Moreno Escobar, Edgar Antonio Domínguez Ramírez,
Jesús Cruz Hernández, Oswaldo Morales Matamoros, Ricardo Tejeida Padilla

Instituto Politécnico Nacional,
México

jemoreno@esimez.mx

Resumen. La búsqueda de una persona extraviada no es una tarea sencilla de llevar a cabo, por lo que a lo largo de los años se han desarrollado métodos de búsqueda, el problema es que los métodos actualmente existentes cuentan con ciertas limitantes y estas limitantes se ven reflejadas en el tiempo de localización. En la búsqueda de personas el tiempo de localización es un factor muy importante que los rescatistas no se pueden dar el lujo de desperdiciar ya que la persona extraviada está expuesta a grandes peligros. Haciendo uso de la visión por computadora se espera realizar la detección y búsqueda de alguna persona extraviada. Así, se diseñaron e implementaron algoritmos para que un Vehículo aéreo no tripulado o UAV (*Unmanned Aerial Vehicle*) lleve a cabo esta tarea, además gracias a la velocidad de este se espera aminorar el tiempo de localización. Con el uso de un UAV no se pretende reemplazar al ser humano en la difícil tarea de buscar y rescatar personas sino que se pretende que sirva como una herramienta de apoyo en la realización de esta difícil tarea.

Palabras clave: Visión por computadora, vehículos aéreos no-tripulados, sistemas de búsqueda y rescate, sistema visual humano, cuadricópteros.

White Donkey: People Search with Unmanned Aerial Vehicles based on Computer Vision

Abstract. Searching for a missing person is not an easy task to accomplish, so over the years search methods have been developed, the problem is that the methods currently available have certain limitations and these limitations are reflected in time location. Time location in a person search is a very important factor that rescuers cannot afford to waste because the missing person is exposed to great dangers. Using computer vision is expected to search for any missing person designing and implementing algorithms in order to an Unmanned Aerial Vehicle perform this task, also thanks to the speed of this is expected to reduce the time location.

By using of a Unmanned Aerial Vehicle is not intended to replace the human being in the difficult task of searching and rescuing people but rather is intended to serve as a support tool in performing this difficult task.

Keywords: Computer vision, unmanned aerial vehicles, search and rescue systems, human visual system, quadricopters.

1. Introducción

El uso de drones en situaciones de emergencia cada vez es más frecuente, debido a que los drones permiten el acceso de manera efectiva, rápida a la hora de acceder a todo tipo de terrenos desconocida, además de desplazarse fácilmente por terrenos irregulares gracias a su gran estabilidad. Además es posible equipar a los drones con cámaras de alta definición por lo que el equipo de rescate tiene la facilidad de dirigir los drones haciendo uso de computadores portátiles y equipos de control [3]. El equipo de rescate hace uso de drones para tareas de búsqueda de personas desaparecidas. A continuación se mencionan algunos ejemplos.

España destaca dentro de este ramo, a continuación se citan tres ejemplos:

1. Ya se han utilizado drones de forma efectiva en Zaragoza en la búsqueda de personas desaparecidas por parte de una empresa privada que ha colaborado con la policía aragonesa,
2. La dotación de bomberos de valencia ya está utilizando drones en labores de búsqueda de personas desaparecidas con éxito,
3. El departamento de ingeniería de sistemas y automática de la universidad de Málaga está realizando un proyecto de investigación en el cual une los equipos de rescate canino con drones, estos drones hacen uso de cámaras térmicas Figura 1(a).

Por otro lado en enero de 2014, *Intel y Ascending Technologies* presentaron prototipos de drones multicópteros que son capaces de desplazarse por una pista de obstáculos y evitar automáticamente a quien encuentre en su camino. Los prototipos utilizan el módulo de cámara RealSense de Intel, cuyo peso es de solo 8 g y su grosor menor a 4 mm.

Mientras tanto en *México*, en el estado de Morelos la cruz roja mexicana firmo un convenio con la empresa SkyBotica para el uso de un grupo de drones Figura 1(b). Estos se emplearan en la tarea de búsqueda y rescate de personas extraviadas

2. Marco teórico

2.1. Visión por computadora: algoritmo Viola & Jones

Este algoritmo tiene un bajo coste en hardware y consta de dos partes principales: Clasificador en cascada el cual garantiza una discriminación rápida



Fig. 1. (a) Drones en Málaga y (b) Cruz roja mexicana y SkyBotica.

y un entrenador de clasificadores basado en Adaboost [2]. Viola Jones tiene una probabilidad de verdaderos positivos del 99.9% y una probabilidad de falso positivos del 3.33% y a diferencia de otros algoritmos utilizados en métodos de caracteres invariantes, este sólo procesa la información presente en una imagen en escala de grises. No utiliza directamente la imagen sino que utiliza una representación de esta llamada imagen integral.

Para determinar si en una imagen se encuentra una cara o no, el algoritmo divide la imagen integral en subregiones de tamaños diferentes y utiliza una serie de clasificadores (clasificadores en cascada), cada uno con un conjunto de características visuales. En cada clasificador se determina si la subregión es una cara o no. El uso de este algoritmo supone un ahorro de tiempo considerable ya que no serán procesadas subregiones de la imagen que no se sepa con certeza que contienen una cara y sólo se invertirá tiempo en aquellas subregiones que posiblemente sí contengan una cara, Figura 2.

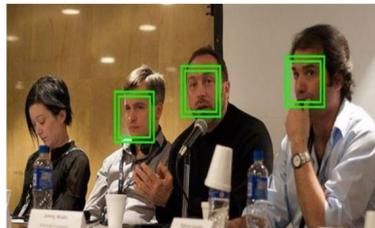


Fig. 2. Detección facial por medio del Algoritmo Viola & Jones.

2.2. Principio de operación del tetramotor

Un tetramotor es un sistema multivariable que cuenta con seis grados de libertad (x , y , z , pitch, roll y yaw) y subactuado ya que consta de menos actuadores que grados de libertad. Los rotores del tetramotor producen las fuerzas y momentos principales que actúan sobre este, Figura 3 [1]. El balance del torque total del sistema se logra debido a que dos pares de motores rotan en direcciones

opuestas. Para el estudio del sistema se utilizan dos sistemas de referencia el sistema T fijo a tierra y el sistema B fijo al vehículo [4].

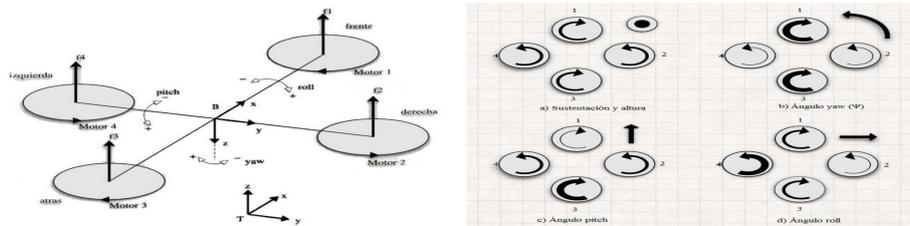


Fig. 3. Operación del tetramotor: (a) Sistemas de referencia y (b) Su dinámica.

La dinámica de un tetramotor se puede resumir en los siguientes 4 puntos:

1. Sustentación y altura: Los cuatro motores giran con la misma potencia y en direcciones opuestas.
2. Diferencia en torque para manipular yaw (ψ): Para girar hacia la derecha sobre su propio eje, los motores 1 y 3 aumentan su potencia girando ambos hacia el mismo sentido. Para girar hacia la izquierda sobre su propio eje, los motores 2 y 4 aumentan su potencia girando ambos hacia el mismo sentido.
3. Diferencia en torque para manipular pitch (θ): Para avanzar hacia adelante, los motores 2 y 4 aumentan de potencia, de igual forma el motor 3 aumenta su potencia pero a un grado mayor que los motores 2 y 4. Para avanzar hacia atrás en lugar de que el motor 3 aumente el que aumenta es el motor 1.
4. Diferencia en torque para manipular roll (φ): Para avanzar hacia la derecha, los motores 1 y 3 aumentan de potencia, de igual forma el motor 4 aumenta su potencia pero a un grado mayor que los motores 1 y 3. Para avanzar hacia la izquierda en lugar de que el motor 4 aumente el que aumenta es el motor 2.

Los motores utilizados en este proyecto son de tipo brushless. Un motor eléctrico brushless (sin escobillas) se define como un motor eléctrico que no emplea escobillas para realizar el cambio de polaridad en el rotor. Los motores brushless están compuestos por una parte móvil, el rotor, que es donde se encuentran los imanes permanentes y una parte fija denominada estator o carcasa sobre la cual van dispuestos los bobinados de hilo conductor. Para hacer un funcionar un motor brushless es necesario el uso del electronic speed control (ESC). Un electronic speed control (control electrónico de velocidad o ESC) es un circuito electrónico que tiene como función variar la velocidad de un motor eléctrico, su dirección y posiblemente también para actuar como un freno dinámico. Los ESC son comúnmente utilizados en los motores accionados eléctricamente por radio control, con la variedad más utilizada para motores brushless, proporcionando una fuente generada electrónicamente de tres fases de bajo voltaje. Un ESC

puede ser una unidad independiente que se conecta en el canal del control de aceleración en el receptor o puede ya estar incorporado en el propio receptor, como es el caso en la mayoría de los vehículos de juguete de calidad de R / C.

La potencia de los tetramotores será controlada por un transceptor, cuya definición es la siguiente: Un transceptor es un dispositivo que cuenta con un transmisor y un receptor que comparten partes de su propio circuito. Cuando el transmisor y el receptor no tienen en común partes del circuito electrónico se conoce como transmisor-receptor. Dado que determinados elementos del circuito se utilizan tanto para la transmisión como para la recepción, la comunicación que provee un transceptor solo puede ser semidúplex, lo que significa que pueden enviarse señales en ambos sentidos, pero no simultáneamente. El Transceptor RF1100 es un módulo de RF a una frecuencia de 433 MHz. El módulo transmite de forma inalámbrica a velocidades de transmisión de 4800, 9600 y 19200 bit/s, tiene 256 canales individuales que pueden ser seleccionados. Este módulo soporta la transmisión punto a multipunto.

3. Metodología

3.1. Descripción general del sistema

Para el estudio y planteamiento de la presente Metodología es necesario conocer los diferentes métodos existentes que realizan la detección de objetos mediante técnicas de visión por computadora. También, los principios de operación de los elementos principales que conforman al UAV para su diseño y armado.

Este sistema busca personas en el entorno haciendo uso de vehículos aéreos no tripulados y de técnicas de visión por computadora. En la actualidad ya se hace uso de drones para buscar personas pero estos drones son manipulados por los rescatistas, ya sea por medio de control remoto o con una computadora portátil. El rescatista maneja el dron para detectar personas en el entorno.

El presente sistema hace uso de algoritmos de visión por computadora para que el sistema de visión del dron detecte a una persona de manera autónoma por lo que mientras que el dron lleva a cabo esta tarea, el rescatista puede realizar otras acciones y una vez que el sistema de visión del dron encuentra a una persona en el entorno, este le manda una alerta al rescatista además de que el dron se dirige hacia la persona detectada para que de esta manera el rescatista pueda obtener más detalles de la persona y del entorno.

Para darle autonomía al dron se diseñó e implementó un control digital para este con el fin de poder controlarlo a partir una computadora. Al decir poder controlarlo a través de una computadora no hace referencia a que un usuario manipula al dron, si no que a través de los datos que se obtengan a través del procesamiento de imágenes, la computadora le manda las acciones a realizar al dron, es decir, el sistema de visión con el que cuenta el dron no solo es para detectar personas, si no que también son los ojos de este.

En la Figura 4 se muestra el sistema propuesto de manera general, el cual está dividido en tres Fases principales:

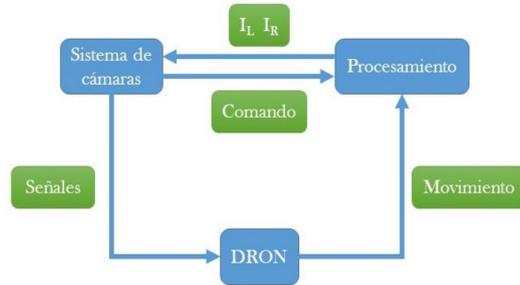


Fig. 4. Sistema General de WHITE – DONKEY.

1. Sistema de Cámaras o Fase de Captura,
2. Procesamiento Digital de Señales e Imágenes,
3. UAV o Interfaz de Control.

Es importante resaltar que se realizó una Interfaz de Usuario para conjugar las tres fases, la cual se considera una fase paralela a las principales.

3.2. Fase de captura

Se realiza el diseño del sistema de visión estereoscópica para ello, se hace uso de la nueva tecnología de modelado e impresión 3D para la construcción de una base donde estarán sujetas las dos cámaras, con una distancia de 10cm entre ellas. Ahora es necesario calibrar las cámaras que conforman al sistema de visión, es decir, obtener los parámetros intrínsecos y extrínsecos de ambas cámaras y así como los parámetros del sistema estereoscópico, Figura 5. Así, para calibrar el sistema de visión se utiliza el método de Zhang [5], este método ya está implementado en el software Matlab por lo que para utilizarlo basta con llamar a la función en Matlab con el comando *cameraCalibrator*, una vez realizada la calibración esta herramienta devuelve un objeto con los parámetros del sistema estereoscópico.

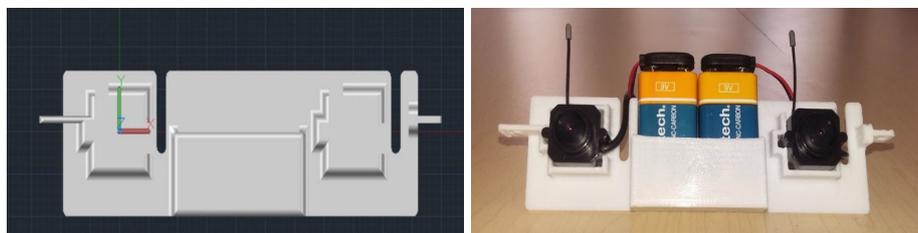


Fig. 5. Modelado en 3D de la base del sistema de visión estereoscópica.

Ya estando listo el sistema de visión para detectar una persona en una imagen se utiliza el algoritmo Viola-Jones el cual ya viene implementado en Matlab.

A continuación se muestra el algoritmo de detección de persona y rostro:

1. Inicio.
2. Configurar la cámara 1.
3. Abrir y Configurar el puerto serial.
4. Tomar una foto del entorno y guárdarla en I.
5. Definir y configurar el detector de objetos utilizando el constructor *bodyDetector*, como imagen de entrada I nombrarlo *bboxBody*.
6. Definir y configurar el detector de objetos utilizando el constructor *faceDetector*, como imagen de entrada I y nombrarlo *bboxes*.
7. Si el contenido de *bboxes* o *boxBody* es mayor a cero dibujar un rectángulo en la imagen I del tamaño y coordenadas *bboxes* también dibujar un rectángulo del tamaño y coordenadas *bboxBody*.
8. Si no hay contenido en *bboxes* mandar un carácter específico por el puerto serial.

También se plantea la metodología general para encender un LED si se detectó una persona, se lleva a cabo en el microcontrolador MSP430G2553:

1. Inicio.
2. Configurar entradas.
3. Configurar salidas.
4. Configurar la comunicación serial.
5. Si el MSP430G2553 en el registro UCA0RXBUF recibe un carácter específico encender un led.
6. Si el MSP420G2553 en el registro UCA0RXBUF recibe un carácter específico apagar el led.

Así para medir la distancia hacia la persona detectada, se propone el siguiente algoritmo:

1. Inicio.
2. Cargar los parámetros de las cámaras, *camStereoParams*.
3. Tomar foto con la cámara derecha, guardarla en *frameRigth*.
4. Tomar foto con la cámara izquierda, guardarla en *frameLeft*.
5. Rectificación estereoscópica de *frameLeft* y *frameRigth* haciendo uso de *camStereoParams*, guardar las nuevas imágenes en *frameLeftRect* y *frameRightRect*.
6. Pasar *frameLeftRect* y *frameRightRect* a escala de grises y guardarlas en *frameLeftGray* y *frameRigthGray*.
7. Detectar cara en la imagen *frameRightGray* y en la imagen *frameLeftGray*.
8. Encontrar los centros de los rostros detectados.
9. Triangulación usando los centros de los rostros detectados.
10. Calcular la distancia en metros
11. Si la distancia es menor de dos metros mandar un carácter específico por el puerto serial.
12. Si la distancia es mayor de dos metros mandar un carácter específico por el puerto serial
13. Si no hay persona mandar un carácter específico por el puerto serial.

3.3. Interfaz de control

Debido a la complejidad de diseño, el tiempo que toma la creación desde cero de un UAV y el poder controlarlo desde una computadora, se decidió utilizar la tarjeta de control kk2.0. Esta tarjeta es utilizada en diferentes vehículos aéreos, tales como:

1. Tricopter
2. Quadcopter +
3. Quadcopter X
4. Hexcopter +
5. Hexcopter X
6. Octocopter +
7. Octocopter X
8. Aero 1S Aileron
9. Aero 2S Aileron
10. Flying Wing
11. Singlecopter 2M 2S
12. Singlecopter 1M 4S

La tarjeta de control kk2.0 tiene 6 grados de libertad que es bastante sensible. También el MPU6050 consta de un giroscopio y un acelerómetro. Esta tarjeta es controlada con el receptor hobbyking hk-t6a, el problema de este receptor radica en la frecuencia ya que es la misma en la que las cámaras transmiten, lo cual provoca demasiada interferencia haciendo imposible la detección de personas. La solución a este problema es remplazar el receptor y el control, en su lugar se realizará un circuito que entregue las mismas señales que el receptor entrega a la tarjeta de control Tabla 1, solo se utilizan 5 canales de los 6 disponibles:

Tabla 1. Valores medidos de cada canal que entrega el receptor.

| Canal | Función | TH | TL |
|-------|----------|--------|---------|
| 1 | Aileron | 1.54mS | 16.79mS |
| 2 | Elevator | 1.54mS | 16.79mS |
| 3 | Throttle | 1.02mS | 17.33mS |
| 4 | Ruder | 1.54mS | 16.79mS |
| 5 | Aux | 2mS | 16mS |

Así, se propone el siguiente algoritmo para controlar los movimientos el UAV a distancia:

- Inicio.
- Configuración de registros para comunicación serial.
- Configuración de los timers.
- Generar 3 PWM's utilizando el timer.
- Con cada carácter recibido en el registro UAC0RXBUF de la comunicaciones serial, el ciclo útil cambiaran según el carácter.

3.4. Interfaz de usuario

Para que el uso del software de búsqueda de personas sea amigable al usuario, se diseña una interfaz gráfica de usuario (GUI, Figura 6) la cual se diseñada con la herramienta GUI que matlab proporciona. En esta aplicación el usuario introduce el puerto de comunicación con el cual se enlaza al transmisor con la computadora. La aplicación cuenta con el panel de ARMED Y SEARCHING. El panel de ARMED nos permite establecer la comunicación con el UAV y el panel SEARCHING nos permite comenzar o detener la búsqueda. Finalmente contamos con una ventana en la cual se muestra a la persona detectada, un indicador gráfico y uno textual. Cuando se detecta a una persona el indicador gráfico está en color verde y el indicador textual indica DETECTED PERSON, cuando no se detecta nada el indicador gráfico está en color rojo y el indicador textual indica UNDETECTED PERSON.

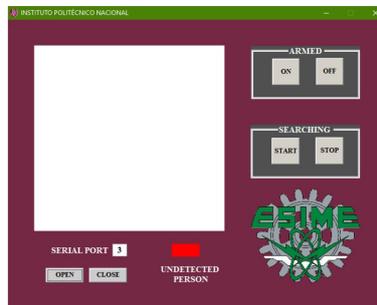


Fig. 6. Interfaz de usuario.

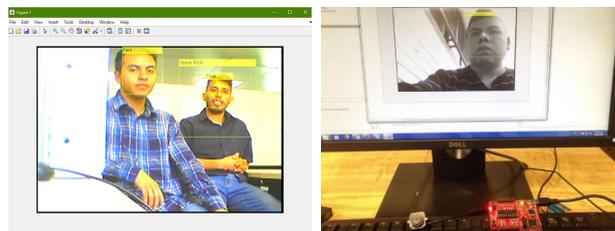


Fig. 7. Detección de rostro y parte superior del cuerpo, además de su representación en una la herramienta de desarrollo LaunchPad.

4. Resultados

1. Detección de rostro y parte superior del cuerpo, Figura 7.

2. Medir la distancia hacia la persona. Ya estando listo el sistema visión estereoscópico se colocó el sistema enfocando hacia una persona, después se ejecutó el programa en matlab y como se encontró persona. La distancia real medida desde las cámaras hacia la persona era 1 m, la distancia calculada fue de 90cm.

3. Enlazar y desenlazar el transmisor con el receptor en el UAV para el control desde una computadora.

4. Aplicación de búsqueda de personas: Detectando el rostro de personas en un escenario de búsqueda y rescate, Figura 8.

Dentro de la aplicación introducimos el puerto serial a utilizar, en este caso usamos el puerto 3 y damos click en el botón OPEN. Ahora para enlazar con el UAV damos clic en el botón ON del panel ARMED.

Al dar clic en el botón START del panel SEARCHING el UAV comienza a elevarse verticalmente y posteriormente gira sobre su propio eje. Cuando el UAV detecta a una persona, esté avanza hacia ella si la distancia es menor de 2m el UAV se detiene, por seguridad de la persona encontrada de lo contrario el UAV continúa avanzando. En la ventana principal de la aplicación se muestra el rostro de la persona detectada además de indicarlo de manera textual y gráfica con un indicador de color verde.



Fig. 8. Representación de la detección de un rostro.

5. Conclusiones

El presente trabajo propuso un sistema formado por dos subsistemas integrados, el primero un UAV controlado por un sistema computarizado, en tanto que el segundo es un sistema de visión por computadora que detecta la distancia cuando se encuentra una persona. El trabajo futuro de este sistema es la puesta en marcha para ser usado en un bosque, por ejemplo. Con ello se corregirán parámetros que en laboratorio no se han tomado en cuenta.

Agradecimientos. Este trabajo es desarrollado con recursos e instalaciones del Instituto Politécnico Nacional, México por medio de los Proyectos SIP-20160786, SIP-20161053 y SIP-20161713. Además, fue apoyado por la Comisión de Operación y Fomento de Actividades Académicas del IPN (COFAA) y Consejo Nacional de Ciencia y Tecnología de México (CONACyT) por medio del Sistema Nacional de Investigadores con las becas No. 56739 (Dr. Moreno), 32772 (Dr. Morales) y 335839 (Dr. Tejeida). Cabe resaltar que este trabajo es parte de la tesis de Nivel Licenciatura de los alumnos Jesús Cruz, y Edgar Domínguez, dirigidos por el Dr. Jesús Jaime Moreno Escobar.

Referencias

1. Bertrand, S., Hamel, T., Piet-Lahanier, H.: Stabilization of a Small Unmanned Aerial Vehicle Model without Velocity Measurement. In: Proceedings 2007 IEEE International Conference on Robotics and Automation. pp. 724–729 (2007)
2. Lang, S.R., Luerssen, M.H., Powers, D.M.W.: Evolutionary Feature Preselection for Viola-Jones Classifier Training. In: Engineering and Technology (S-CET), 2012 Spring Congress on. pp. 1–4 (2012)
3. Lawlor, O., Moss, M., Kibler, S., Carson, C., Bond, S., Bogosyan, S.: Search-and-rescue robots for integrated research and education in Cyber-Physical Systems. In: e-Learning in Industrial Electronics (ICELIE), 2013 7th IEEE International Conference on. pp. 92–97 (2013)
4. Paiva, E.A., Soto, J.C., Salinas, J.A., Ipanaqué, W.: Modeling and PID cascade control of a Quadcopter for trajectory tracking. In: 2015 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON). pp. 809–815 (2015)

White-donkey: búsqueda de personas con vehículos aéreos no tripulados basada en visión ...

5. Zhang, Z.: A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(11), pp. 1330–1334 (2000)

Un algoritmo multi-objetivo NSGA-II para la programación de los pasos de laminación en caliente de aceros

Carlos A. Hernández¹, Guadalupe Castilla², Alejandro López¹,
Juana E. Mancilla¹

¹ Instituto Politécnico Nacional,
Escuela Superior de Ingeniería Mecánica y Eléctrica Azcapotzalco, Ciudad de México,
México

² Instituto Tecnológico de Ciudad Madero, Tamaulipas,
México

cahernandezc@ipn.mx, gpe_cas@yahoo.com.mx

Resumen. El problema de la programación de los pasos de laminación en caliente de aceros es muy complejo, ya que posee grandes espacios de búsqueda de soluciones. El objetivo de este trabajo fue la optimización multiobjetivo del proceso de laminación en caliente de láminas de acero en un tren acabador de seis pasos, implementando el *algoritmo genético de búsqueda no dominada elitista de segunda generación* (NSGA-II en inglés de *Nondominated Sorting Genetic Algorithm*), con dos cruces diferentes (AG1 y AG2) y dos funciones objetivo. Estas fueron el tiempo total de laminado y la flexión de los rodillos de trabajo. Se realizaron dos clases de experimentos computacionales: a) Búsqueda de la mejor solución, b) evaluación comparativa de las configuraciones del NSGA-II. Para cada experimento se obtuvieron 3000 soluciones. En cada molino se calcularon el tiempo de laminación, la fuerza de laminación requerida y los espesores de salida. Para la fuerza se utilizó un modelo constitutivo que calcula la tensión de fluencia, válido para cualquier tipo de acero de bajo, medio carbono y microaleados al Nb, Ti, V. Se reportan las soluciones, seleccionando la mejor de entre 100 con base en el tiempo de cómputo menor. En tiempos menores a 70 s se generaron 100 soluciones. El AG1 tuvo menor tiempo de cómputo que el AG2. Se midió el tiempo de cómputo necesario para generar cada conjunto de soluciones. El AG1 presentó el menor costo computacional. La implementación del NSGA-II minimizó tanto el tiempo de laminación como la flexión de los rodillos de trabajo, a costos computacionales bajos y asegurando que la planicidad de la lámina siempre será menor a la de estándares internacionales.

Palabras clave: Algoritmos genéticos, NSGA-II, optimización multiobjetivo, laminación en caliente, programación, pasos de laminación, aceros al carbono.

A NSGA-II Multi-Objective Algorithm for Hot Rolling Pass Schedule of Steels

Abstract. The problem of Scheduling Hot Rolling Pass of Steels is very complex because it has large spaces search for solutions. The aim of this work was the Multi-Objective Optimization (MOO) of hot rolling steel sheets in a finishing train of six steps, implementing the second generation of Nondominated Sorting Genetic Algorithm (NSGA-II). It was used two different crossover (AG1 y AG2) operators and two objective functions. These were the total time of rolling and bending of the work rolls. Two kinds of computational experiments were performed: a) Search for the best solution, b) benchmarking of both NSGA-II configurations. For each experiment was obtained 3000 solutions. Rolling time, separation roll force, and exit thickness for each rolling stand are calculated. A flow stress constitutive model for roll force was used, valid for any low, medium carbon and Nb, Ti, V microalloyed steels. The solutions are reported, choosing some of them between 100 based on the lower computing time. The 100 solutions were generated in times lower than 70s. AG1 had lower computing times AG2. The NSGA-II implementation minimized both the rolling time as crown of the work rolls, at low computational costs and ensuring that the flatness of strip will always be lower than international standards.

Keywords: Genetic algorithms, NSGA-II, multi-objective optimization, hot rolling, schedule, pass rolling, carbon steels.

1. Introducción

El proceso de laminación de aceros en caliente es uno de los métodos de conformado más utilizados para la obtención de productos para diferentes industrias metalmeccánicas. En este proceso, el espesor de un planchón de acero se reduce gradualmente aplicando fuerzas de compresión mediante rodillos de laminación [1]. Las fuerzas de laminación aplicadas tienden a incrementarse a medida que el porcentaje de reducción del espesor de la lámina aumenta, pudiendo llegar a provocar flexión en los rodillos de trabajo, pudiendo provocar imperfecciones en la superficie de la lámina o defectos de planicidad por corona de flexión. Esto último produce una lámina que es más delgada en el centro que en las orillas [2]. Existen métodos mecánicos para corregir este tipo de problemas, como el rectificando de los rodillos. Así, cuando el rodillo se somete a fuerzas de separación y se flexiona se obtendrá un espesor constante en la lámina. En el caso de laminación de aceros, el radio del punto máximo de la comba es en general 0.25 mm mayor que en los extremos de los rodillos [3]. Esto implica que el rodillo sólo puede someterse a una carga específica, lo cual en la industria es ineficiente. De no ser así, las láminas producidas por este rodillo presentarían defectos de

planicidad. Además, una mala selección de los pasos de laminación incrementa el tiempo del proceso, lo cual es improductivo.

Cuando la solución de un problema tiene un espacio de búsqueda muy amplio, existen limitaciones de tiempo para llegar a una solución útil y no existe un método analítico mediante el cual se pueda llegar a una solución exacta. Por ello se usan técnicas metaheurísticas para encontrar soluciones óptimas alejadas de óptimos locales; es decir, se orienta la búsqueda acorde a la evolución del proceso. Los algoritmos evolutivos utilizan mecanismos inspirados en la evolución biológica, tales como la selección, reproducción y la mutación. Las soluciones se modelan como individuos de una población. Una función de aptitud determina el entorno donde se encuentran las soluciones. Dentro de los algoritmos evolutivos están los algoritmos genéticos (AG), que son una técnica de búsqueda iterativa, buscando derivar estrategias de optimización basadas en la generación de poblaciones de individuos aptos mediante la reproducción de los padres [4]. Los problemas reales usualmente requieren la búsqueda de soluciones que satisfagan en forma simultánea múltiples criterios de desempeño u objetivos que pueden ser contradictorios, es decir, se desconoce la manera óptima de combinar los diferentes objetivos o que sea inadecuado, cuando no es imposible hacerlo, a lo cual se le conoce como un Problema de Optimización Multiobjetivo (MOP en inglés de *Multiobjective Optimization Problem*) [5].

Ya que el proceso de laminación en caliente obedece a más de una variable, el espacio de soluciones óptimas puede ser infinito y el tiempo para obtener solo una de ellas con métodos analíticos puede tomar demasiado tiempo y disminuir la eficiencia del proceso. Por ello, la aplicación de técnicas metaheurísticas a este proceso ha sido objeto de varias investigaciones cuyos autores han enfocado su atención en diferentes variables. En este trabajo se trata de encontrar los espesores intermedios óptimos (h_1, \dots, h_{f-1}) tales que minimicen tanto el tiempo total de laminación como la flexión máxima de los rodillos de trabajo. Para la flexión, esta deberá ser menor a los límites de planicidad establecidos por algunas normas internacionales como la EN 10 051, ASTM A568 o A653, la JIS G-3193 o G-3116 y la DIN 10051. Por tratarse de una investigación de carácter multiobjetivo, este trabajo se integra a las investigaciones cuyos propósitos son la optimización de las variables intrínsecas del proceso, para reducir el tiempo total de laminación, así como la optimización de la operación y vida útil de la maquinaria para disminuir la máxima flexión de los rodillos de trabajo de la cédula de laminación. Por ello, en las últimas dos décadas, el proceso de laminación ha sido objeto de estudio aplicando diferentes técnicas metaheurísticas y, más recientemente, de algoritmos evolutivos para la optimización multiobjetivo del proceso, por el hecho de que existe más de una restricción, algunas de las cuales se mencionan a continuación.

Nolle y Armstrong [3] realizaron la modelización del proceso de un tren acabador de laminación de siete pasos aplicando redes neuronales [6], dirigido a la optimización de factores cuyos parámetros se reflejaban en la planicidad y la deformación longitudinal de una lámina obtenida mediante un tren laminador. Un desarrollo importante en el campo de la aplicación de los algoritmos

genéticos (AG) a la laminación en frío fue propuesto por Wang y cols. [7] quienes optimizaron la programación de una cédula de laminación en frío. En su investigación obtuvieron los parámetros que optimizaban el proceso infiriendo el incremento en la calidad de la lámina y el aumento de la vida útil de los rodillos de laminación. Wang demostró que los parámetros se mejoraron destacando la reducción de la potencia empleada, las revoluciones por minuto necesarias para el proceso, así como el par requerido. En 2003 Nolle y Zelinka [8] desarrollaron un algoritmo heurístico de optimización denominado algoritmo de migración auto organizado (SOMA del inglés *Self-Organizing Migration Algorithm*). Tanto el objetivo de ese trabajo como la función a optimizar fueron los mismos del trabajo publicado en 1999 [3]. SOMA se aplicó en un programa de laminación para catorce diferentes láminas y se relacionó con la calidad de las láminas obtenidas por el proceso de laminación. Los resultados arrojados se compararon con los de la planicidad de láminas medidas en *I-Units* provenientes de pruebas industriales, obteniendo SOMA un 33% de mejora. Estos autores también desarrollaron un AG para la optimización mediante simulación de los rodillos de trabajo del tren acabador. El factor a optimizar fue una función compuesta por una combinación de parámetros de planicidad. Se comparó el AG con ocho diferentes algoritmos de optimización experimentales, como lo son el recocido simulado, ascenso de colina y la trayectoria aleatoria. Demostraron que el AG fue el que obtuvo los mejores resultados. En 2007, Nolle [9] implementó un algoritmo heurístico de búsqueda de tamaño de pasos auto adaptable (SASS del inglés *self-adaptive step size search*), para optimizar el perfil de los rodillos de trabajo utilizados en el proceso de laminación en caliente. El SASS es un algoritmo basado en los principios aplicados en la técnica de optimización experimental de ascenso de colina, donde únicamente se puede configurar el tamaño de la población y aunque el algoritmo es muy simple, los resultados obtenidos fueron muy buenos. Los resultados arrojados por SASS se valoraron en relación con la planicidad de la lámina y su deformación longitudinal, siendo mejores que los datos industriales contra los cuales se compararon. Recientemente, Pholdee et al. [10] optimizó la planicidad de láminas durante el bobinado utilizando un método de evolución diferencial junto con varios algoritmos evolucionarios. Este método realiza gran cantidad de iteraciones, por lo cual requiere nuevos algoritmos evolucionarios que reduzcan la cantidad de evaluaciones a las funciones.

2. Formulación del problema de optimización

El problema de la optimización de la programación de los pasos de laminación en caliente para la fabricación de lámina plana radica en que la minimización del tiempo de procesamiento no toma en cuenta el efecto que tendría sobre la flexión del rodillo de trabajo y con ello la afectación del perfil de la lámina, tal que sobrepase las tolerancias cada vez más exigentes para la calidad de las láminas. Ya que las minimizaciones del tiempo de laminación y de la flexión del rodillo involucran soluciones en conflicto se requiere resolver el problema de optimización mediante algoritmos multiobjetivos.

Existen muchos métodos de solución para encontrar un conjunto de soluciones lo más cercano posible a un frente óptimo de Pareto de los tiempos de laminación t y la flexión del rodillo ν , en cada paso de laminación. Ya que las fuerzas de laminación están gobernadas por los parámetros del proceso y por la resistencia a la deformación del material, conviene que el tiempo de laminación y la flexión del rodillo óptimos provengan de un NSGA-II, ya que permite hallar un número de soluciones no dominadas que convergen en un frente óptimo de Pareto. Para este trabajo se han considerado tres restricciones, a) la potencia óptima de laminación ($P_{\text{ópt}}$) para obtener un espesor de salida (h_i) sea menor que la potencia nominal (P_{nominal}) de cada molino de laminación, b) el espesor de salida (h_i) sea estrictamente menor que el espesor de entrada (h_{i-1}) y c) la corona de flexión ($\nu_{\text{ópt}}$) sea menor a la establecida por las normas de planicidad (ν_{norma}). Entonces el problema de optimización se formula como

$$\left. \begin{array}{l} \text{minimizar } f_1(t, \nu) = \sum_{n=1}^n t_i, \\ \text{minimizar } f_2(t, \nu) = \frac{R_A x^3}{6EI} - \frac{q(x - l_1)^4}{24EI} + \left(\frac{qx(\frac{L}{2} - l_1)^3}{6EI} - \frac{R_A x L^2}{8EI} \right), \\ \text{sujeto a: } P_{\text{ópt}} < P_{\text{nominal}} \\ \quad h_{i-1} < h_i \\ \quad \nu_{\text{ópt}} < \nu_{\text{norma}}, \end{array} \right\} \quad (1)$$

siendo R_A la reacción en un extremo del rodillo (punto A), x la distancia longitudinal dada a lo largo de la cual ocurre la flexión del rodillo, l_1 la longitud de la primera sección del rodillo de trabajo, L la longitud total del rodillo, E el módulo de elasticidad del material del rodillo e I el momento de inercia de la sección transversal del rodillo. Los parámetros de entrada del planchón se pueden consultar en trabajos previos [11,12].

3. Parámetros del AG

El algoritmo genético se inicializa mediante parámetros establecidos por el usuario previamente (Tabla 1).

Tabla 1. Parámetros del NSGA-II.

| Parámetro | Magnitud |
|------------------------|----------|
| Población inicial | 100 |
| Número de generaciones | 100 |
| Taza de cruzamiento | 0.9 % |
| Taza de mutación | 0.2 % |

4. Instancias empleadas

Las instancias para la experimentación en este trabajo se obtuvieron de trabajos previos [13]. Para determinar la funcionalidad del AG desarrollado, así como sus variantes en el tipo de cruza, se probaron 17 instancias. Cada instancia representa un problema diferente. En cada problema se determinaron las reducciones intermedias para laminar un acero hasta obtener el espesor final deseado en un tren de laminación de 6 pasos. Cada instancia contiene el espesor inicial del planchón (h_0), el espesor final (h_f) deseado, la composición química del material (%C, %Mn, %Si, %Mo, %Nb, %Ti, %V) y el ancho del planchón (w). También, y para cada paso del tren de laminación, se deben especificar diámetros de rodillos de trabajo (D_i), velocidades de rodillos (v_i), distancias entre las cajas de laminación (l_i), potencias de motores (P_i), temperaturas intermedias (T_i) y el tamaño inicial de grano (d_{0_i}).

5. Solución al problema de optimización

Para dar solución al problema de optimización de una cédula de laminación en caliente en esta investigación se empleó el NSGA-II. Se configuró al AG en dos formas diferentes. En la primera se utilizó la cruza en dos puntos (AG1) y en la segunda se utiliza cruza SBX (AG2). En ambos casos se implementó una selección aleatoria elitista para generar la población inicial y se mantuvo una mutación constante.

La población inicial está compuesta sólo por individuos factibles, que evolucionan en busca de la mejor solución al problema. El número de individuos lo define el usuario al inicio de la ejecución del algoritmo, al igual que el número de generaciones, que representa el ciclo del algoritmo.

5.1. Algoritmo NSGA-II

En el Algoritmo 1 se muestra el código básico del NSGA-II. Para que se genere la primera población de padres y cumpla su ciclo de operación, los individuos deben cumplir con todas las restricciones del problema.

En el Algoritmo 2 se muestran las operaciones más importantes, así como los cálculos para determinar las restricciones del problema con las cuales se clasifica a un individuo como viable o no viable.

5.2. Generación población inicial

Dado un espesor inicial h_0 y un espesor final h_f del planchón, considerados constantes del problema, se generan individuos cuya representación es la siguiente:

$$X = [h_0, h_1, h_2, h_3, h_4, h_5, h_f] . \quad (2)$$

A continuación se explica brevemente la solución obtenida para la instancia número 1. Con los datos de la instancia y mediante la ecuación (3), que representa

Algoritmo 1 Ciclo principal de operaciones de NSGA-II.

```

1: Inicio
2: Repite.
3: Inicialización de parámetros
4: Genera individuos aleatorios  $\mu$ .
5: Evalúa a los individuos.
6: Si  $\mu$  es factible entonces
7:      $\mu \rightarrow P_t$ 
8:     Deshecha  $\mu$ 
9:     Hasta Criterios de término.
10:  $F$ =Ordenamiento no dominado ( $P_t$ )
11: Genera  $Q_t$  a partir de  $P_t$ 
12: Mientras (Criterio de parada) Repite
13:     Genera  $R_t = Q_t \cup P_t$ 
14:      $F$ =Ordenamiento no dominado rápido ( $R_t$ )
15:     Mientras  $|P_{t+1}| \geq N$  hacer
16:         Distancia de Agrupamiento ( $F_i$ )
17:          $P_{t+1} = P_{t+1} \cup F_i$ 
18:         Ordena ( $P_{t+1}, \geq n$ )
19:          $P_{t+1} = P_{t+1}[0 : N]$ 
20:          $Q_{t+1}$ =Genera nueva población  $P_{t+1}$ 
21:          $t = t + 1$ 
22:     Fin Mientras
23: Fin

```

la reducción media \bar{r} para n pasos de laminación (=6), se obtienen los individuos de la población inicial:

$$\bar{r} = 1 - \sqrt[n]{\frac{h_f}{h_0}} . \quad (3)$$

Cada individuo representa una solución al problema:

$$X = [48, 36, 25, 18, 10, 6, 38] ,$$

e implica que tal individuo es factible al haber cumplido con las restricciones del problema.

La población de 100 individuos generada representa a la población inicial (P_0), que se clasifica según los niveles de no dominancia en frentes de Pareto (F_1, F_2, \dots, F_n). Una vez que todos los individuos se han ordenado, se les asignó una calificación denominada aptitud. Los individuos que ocupan el frente F_1 reciben la mejor calificación, debido a que son individuos no dominados, como se muestra en la Tabla 2.

5.3. Cruza

Se considera a cada individuo como la representación de una solución para el problema y cada solución tiene un rango de no dominancia asociado (f_i), así como una distancia de apilamiento (d_i).

Algoritmo 2 Ciclo de operaciones principales para determinar si un individuo satisface las restricciones del problema.

-
- 1: **Inicio**
 - 2: **Entrada** Especificaciones del planchón, h_0, h_f, T_i, d_0
Composición química %C, %Mn, %Si, %Mo, %Nb, %V no vacío.
 - 3: **Entrada** Especificaciones de la cédula de laminación, l_i, P_i, v_i, n no vacío.
 - 4: **Entrada** Especificaciones de los rodillos de trabajo, D_i, L_i, E .
 - 5: **Entrada** Restricciones X_n
 - 6: **Salida Individuo viable** si X_n se cumplen
 - 7: **Individuo no viable** en caso contrario.
 - 8: **Repite.**
 - 9: Genera individuos aleatorios μ_i .
 - 10: **Para** μ_i **Calcula**
 - 11: Tensión de Fluencia σ_i
 - 12: Resistencia a la deformación $\bar{\kappa}_i$
 - 13: Fuerza de laminación F_i
 - 14: Calcula par de laminación G_i
 - 15: Calcula potencia de laminación σ_i
 - 16: Calcula flexión en los rodillos de trabajo $Flex_i$.
 - 17: Calcula tiempo total de laminación t_{total} .
 - 18: Evalúa a los individuos.
 - 19: **Si** μ_i es factible
 - 20: $\mu \rightarrow P_i$
 - 21: **Si no** Descarta μ_i
 - 22: **Hasta** Criterios de término.
 - 23: **Fin**
-

Tabla 2. Ordenamiento en frentes no dominados de Pareto.

| Frente | Padres | Fitness |
|---------------------|------------|---------|
| F_1 | 2, 8, 9, 5 | 10 |
| F_2 | 1, 3, 4 | 9 |
| F_3 | 7, 6 | 8 |
| F_4 | ... | 7 |
| F_n | ... | ... |
| \sum Padres = 100 | | |

Una vez completado el ordenamiento de la población inicial de padres P_0 en los frentes de Pareto se procedió a su cruce mediante la técnica de torneo binario. Esta comienza con la selección aleatoria de dos individuos de la población de padres. La mejor solución es por comparación y el ganador es el que tiene la mejor calificación o rango de no dominancia. En el caso de estar situadas en el mismo frente de Pareto, la ganadora del torneo es aquella que tiene mejor distancia de apilamiento, también conocida como distancia de Crowding d_i , que representa la mayor distancia entre las soluciones de la población asegurando la diversidad de soluciones.

La d_i se calcula mediante la ecuación (4),

$$d_{I_j^m} = \frac{f_m^{I_{j+1}^m} - f_m^{I_{j-1}^m}}{f_m^{\text{máx}} - f_m^{\text{mín}}} , \quad (4)$$

siendo $f_m^{\text{máx}}, f_m^{\text{mín}}$ los valores máximo y mínimo de la función objetivo m y $f_m^{I_{j+1}^m}, f_m^{I_{j-1}^m}$ son las soluciones vecinas a la j -ésima configuración para cada una de las funciones objetivo m . La distancia resultante fue la suma de las distancias en cada una de las direcciones de las funciones objetivo. Se dio preferencia a aquellos individuos localizados en una región menos poblada para generar dispersión en las soluciones.

Una vez que los padres ganadores del torneo binario se seleccionaron, se realizó la cruce entre ellos para generar una población de tamaño N . Los métodos de cruce fueron la de dos puntos y SBX. Para el cálculo de la primera generación del hijo₁, sólo se consideran los h_1 de ambos padres. Aplicando la ecuación para la cruce SBX, cuando $\beta = 1.5$, $P_1 = 0.032$ m y $P_2 = 0.034$ m se tiene que:

$$\begin{aligned} H_1 &= 0.5 [(P_1 + P_2 - \beta(P_1 - P_2))] = 0.0315, \\ H_2 &= 0.5 [(P_1 + P_2 + \beta(P_1 - P_2))] = 0.0345. \end{aligned}$$

5.4. Mutación

El operador mutación es necesario para preservar la diversidad genética de la población evitando divergencia prematura. En este trabajo se implementó mutación uniforme para el desarrollo del NSGA-II. La forma en que se desarrolla este tipo de mutación es la siguiente:

$$\begin{aligned} P &= [h_0, h_1, h_f], \\ P' &= [h_0, h'_1, h_f], \end{aligned}$$

siendo $h'_1 = \text{random}(LI, LS)$. Se utilizó una distribución uniforme y (LI, LS) definen los límites mínimos y máximos de la variable h'_1 .

5.5. Unión padres e hijos

En este algoritmo, la población descendiente Q_t de tamaño N se crea en primera instancia utilizando la población de padres P_t de tamaño N . Después, las dos poblaciones se combinan para formar R_t de tamaño $2N$. A esta nueva población $R_t = P_t \cup Q_t$ se le deberán aplicar todas las operaciones para generar la nueva población de individuos. Sin embargo, como la población R_t es de tamaño $2N$ y solamente existen N configuraciones que conforman la población descendiente, no todas las configuraciones de los frentes pertenecientes a la población R_t podrán integrarse a la nueva población, por lo cual aquellos frentes

que no pueden ser acomodados desaparecen, es decir no son considerados en las nuevas poblaciones, asegurando así que sólo los frentes más aptos sigan reproduciéndose.

El ciclo de operaciones del NSGA-II es repetitivo en todas las generaciones, por lo cual sucesivamente se crea la población descendiente Q_{i+1} , a partir de P_{i+1} , aplicando selección por torneo para apilamiento, cruza y mutación, hasta que la instrucción de paro del algoritmo se cumpla. La instrucción de paro es parte de los parámetros de entrada del algoritmo.

6. Evaluación comparativa

Debido a que se implementó el NSGA-II configurado en dos distintas formas, modificando la técnica de cruza en cada uno de ellos, es conveniente establecer si el tipo de configuración afecta los resultados obtenidos. El objetivo de la evaluación comparativa fue identificar si existen diferencias en los resultados obtenidos de las diferentes configuraciones del NSGA-II.

En la Tabla 3 se muestran las dos configuraciones utilizadas en este trabajo.

Tabla 3. Configuraciones del NSGA-II para este estudio.

| Algoritmo | Tipo de selección | Cruza | Mutación |
|-----------|-------------------|----------|----------|
| AG1 | E. por torneo | 2 puntos | Uniforme |
| AG2 | E. por torneo | SBX | Uniforme |

Se consideró al costo computacional para propósitos de la evaluación. Para realmente lograr una comparación objetiva entre soluciones, es necesario definir una medida cuantitativa de la calidad de las mismas. Para problemas de optimización multiobjetivo donde no existe una solución única sino un conjunto de soluciones, no es fácil definir un criterio para determinar cuándo un conjunto es mejor que otro debido a que se espera que un conjunto de soluciones se encuentre compuesto únicamente por soluciones mutuamente no comparables. Es decir, para cualquier par de soluciones \vec{x}_1, \vec{x}_2 no ocurre que $\vec{x}_1 \leq \vec{x}_2$ ni $\vec{x}_1 \geq \vec{x}_2$ debido a que todas ellas son viables y es el usuario quien debe seleccionar una de modo manual o programado.

Es posible definir un indicador de calidad que asigna un valor real a un conjunto de aproximación del frente de Pareto. Un indicador debe definir un orden total en los conjuntos de aproximación del frente de Pareto. Si se tienen dos conjuntos de aproximación A, B tales que $I(A) > I(B)$, entonces el conjunto A es preferible sobre B . Se puede asignar un indicador particular. Por tal motivo puede ocurrir que si $I(A) > I(B)$ utilizando algún indicador, entonces $I'(A) > I'(B)$ tendrá un indicador distinto.

7. Condiciones de la experimentación

Las condiciones fueron las siguientes:

- Se empleó una computadora con procesador Intel Core *i7* con una velocidad máxima de procesamiento de 3.1 GHz y 12 GB RAM.
- Se realizaron 30 corridas por cada una de las 17 instancias para obtener los valores promedios de cada una de ellas así como garantizar la fiabilidad de los resultados. Cada corrida contiene en un archivo de salida con 100 soluciones, todas viables porque cumplieron con las restricciones del problema.

Se implementó el algoritmo bajo los parámetros mostrados en la Tabla 4.

Tabla 4. Parámetros del NSGA-II.

| Parámetro | Magnitud |
|-------------------|----------|
| Población inicial | 100 ind. |
| Núm. generaciones | 100 gen. |
| Cruza | 95% |
| Mutación | 20% |

Para cada experimento se obtuvieron tres mil soluciones que satisfacen a las funciones objetivo de este problema. Además, en cada molino se obtuvo el tiempo de laminación, la fuerza de laminación y los espesores intermedios óptimos. Así mismo se obtuvieron los tiempos de cómputo de cada experimento.

8. Búsqueda de la mejor solución

En otras propuestas de algoritmos publicados en la literatura para el problema de laminación en caliente, el espacio de soluciones quedó restringido a la búsqueda de únicamente una solución por cada experimento que entregan buenos resultados, pero suelen tener altos tiempos de cómputo. Las soluciones obtenidas en este trabajo mediante el AG sirvieron para seleccionar la mejor de ellas. Esta depende directamente de los requerimientos del usuario, es decir es él quien selecciona una respuesta del conjunto como la mejor, basado en su experiencia y conocimiento del proceso, con la seguridad de que cualquiera de ellas cumple con las restricciones del problema. En esta investigación se seleccionó aquella corrida que generó el menor tiempo de cómputo, evaluándose 100 soluciones. Se consideraron dos indicadores principales, tiempo de laminación y flexiones máximas en los rodillos de trabajo; se dio un peso específico a cada indicador. El indicador con mayor peso es el tiempo de laminación. Por ser un problema de minimización se seleccionó como mejor solución la que presentó aquellos indicadores cuyos valores son los mínimos. Con base en este criterio se obtuvieron los siguientes resultados.

8.1. Tiempo de cómputo

En la Fig. 1 se muestra el tiempo de cómputo promedio que le tomó a cada una de las configuraciones del algoritmo propuesto en generar una corrida para la instancia 1. En un lapso de 15 a 70 segundos, el AG es capaz de generar conjuntos de 100 soluciones.



Fig. 1. Tiempo promedio para generar una corrida de la Instancia 1.

Nótese que el AG1 tiene una mejor eficiencia sobre el AG2 pues en todos los casos requiere menor tiempo de cómputo para generar una corrida. Además, para el proceso de selección de la mejor solución de nuestro problema se eligió la corrida que requirió de menor tiempo de cómputo, enmarcadas con circunferencias en dicha Fig. 1. La corrida generada en menor tiempo fue la 14 para el AG1 y la dos para el AG2.

También se seleccionó de entre las 100 soluciones aquella que mejor se adaptó al problema. En la Fig. 2 se muestran los tiempos de laminación para cada una de las 100 soluciones determinadas mediante el AG1, siendo la solución 71 la que tuvo el menor tiempo de laminación.

En la Fig. 3 se muestran los resultados para tiempos de laminación obtenidos mediante el AG2, para la instancia número 1, apreciándose la similitud con los resultados obtenidos mediante el AG1. Sin embargo el tiempo mínimo de laminación obtenido mediante el AG2 es mayor que el del AG1. Se mejoró hasta en un 0.37% el tiempo de laminación industrial.

9. Evaluación comparativa entre las diferentes configuraciones del NSGA-II

Parte fundamental de este trabajo fue determinar el costo computacional de cada una de las variantes del algoritmo propuestas, es decir, el tiempo de cómputo necesario para generar cada conjunto de soluciones. El costo computacional



Fig. 2. Tiempos de laminación para la instancia número 1 producidos por el AG1. Con círculo azul la que tuvo el menor tiempo de laminación.

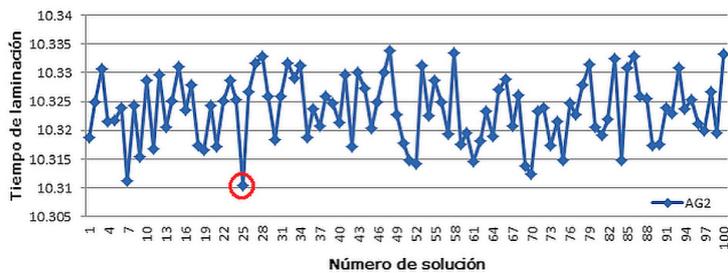


Fig. 3. Tiempos de laminación para la instancia número 1 producidos por el AG2.

se midió desde que inicia la puesta en marcha del algoritmo hasta culminar con la generación del archivo de salida que contiene las soluciones. Ya que el costo computacional es un parámetro utilizado para determinar la mejor solución, fue necesario cronometrarlo.

En la Fig. 4 se muestran los resultados obtenidos aplicando las distintas configuraciones del NSGA-II. Se observa que el AG2, representado en color azul, presentó en todas las instancias mayor costo computacional, es decir tardó mayor tiempo en generar un conjunto de soluciones. Esto se debe en gran medida a que para realizar la cruce SBX es necesario que el algoritmo resuelva ecuaciones para determinar a las nuevas generaciones, a diferencia de la cruce de dos puntos, que emplea un sistema de generación de individuos menos complejo. La diferencia entre los tiempos de cómputo obtenidos en cada una de las corridas aplicando ambas configuraciones del NSGA-II fue muy pequeña, así como en la calidad de la solución obtenida.

En la Fig. 5 se muestra el frente de Pareto con las dos funciones objetivo para el molino 6. Nótese que en la flexión que sufren los rodillos cuando se aplican los parámetros de una cédula de laminación industrial superó los límites establecidos de flexión bajo las normas de planicidad. Por otra parte, las soluciones en el

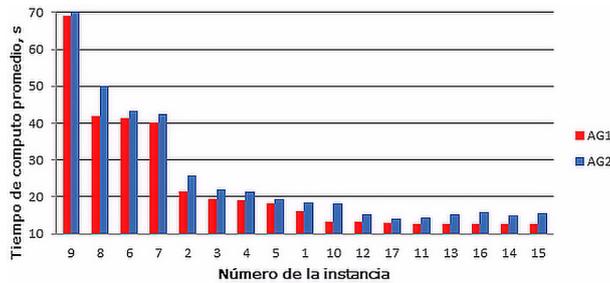


Fig. 4. Comparación entre los tiempos que tarda el AG1 y el AG2 en determinar un conjunto de 100 soluciones.

frente de Pareto encontradas mediante el AG propuesto en todas las ocasiones se mantienen bajo las tolerancias permisibles.

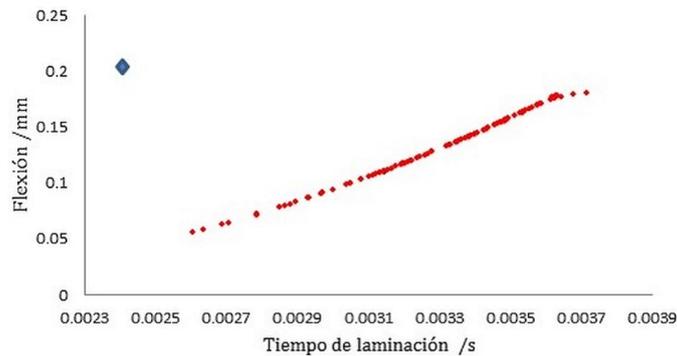


Fig. 5. Frente de Pareto para las flexiones y tiempos calculados mediante el AG (círculos rojos) en comparación con datos industriales (rombo azul).

En la Fig. 6 se muestran las reducciones determinadas por el algoritmo implementado, comparadas con las reducciones reportadas en la cédula de laminación industrial. Se puede decir que las reducciones determinadas mediante el algoritmo están mejor distribuidas en los pasos de laminación, para generar reducciones menos críticas que disminuyan la flexión en los rodillos y la potencia de laminación necesaria. Asimismo, se puede observar que los primeros tres pasos de laminación reducen a más de dos terceras partes el espesor inicial del planchón y los últimos tres pasos son más suaves, ya a que los pasos finales son pasos para dar el acabado que requiere la lámina.

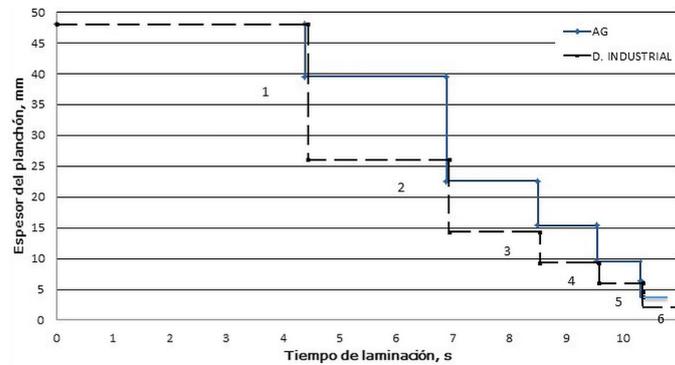


Fig. 6. Comparación entre la cédula obtenida por el algoritmo genético y la cédula industrial. Los números indican el paso de laminación.

10. Conclusiones

- En este trabajo se implementó el NSGA-II para determinar un conjunto de 100 soluciones a un bajo costo computacional.
- La implementación del NSGA-II con sus variantes AG1 y AG2 minimizaron la flexión de los rodillos de trabajo con respecto a la obtenida en datos industriales y mejoró en un 0.37 % el tiempo de laminación industrial.
- La minimización de la corona de flexión asegura que la planicidad de la lámina siempre fue menor a la establecida en normas internacionales.
- Se demostró la factibilidad de la aplicación de algoritmos genéticos multiobjetivo a esta clase de problemas por su fácil adaptación a modelos matemáticos complejos, así como al bajo costo computacional.

Agradecimientos. Este trabajo fue realizado con fondos del Instituto Politécnico Nacional (proyectos clave SIP201612163 y 20161930). Se agradece a Richard Shulkosky de la compañía INTEG Process Group por proporcionar una versión demo del simulador HSMM.

Referencias

1. Petty, D.M.: Application of process modelling—an industrial view. *J Mater Proc Technol* 60, pp. 421–426 (1996)
2. Tseng, A.A., Tong, S.X., Chen, T.C.: Thermal expansion and crown evaluations in rolling processes. *Mater Design* 17, pp. 193–204 (1996)
3. Nolle, L., Armstrong, A., Hopgood, A., Ware, A.: Optimum Work Roll Profile Selection in the Hot Rolling of Wide Steel Strip Using Computational Intelligence. In: *Computational Intelligence: Theory and Applications International Conference*, Dortmund, Germany, Reusch, B. (eds). Springer, Berlin (1999)

4. Holland, J.H.: The dynamics of searchers directed by genetic algorithms. En Lee, Y.C. (Ed.), *Evolution, learning, and cognition*, Teaneck, N. J. World Scientific (1988)
5. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: empirical results. *Evol Comput* 8, pp. 173–195 (2000)
6. Deb, K.: *Multi-Objective Optimization Using Evolutionary Algorithms: An Introduction*. pp. 209–218, Wiley, New York (2001)
7. Wang, D.D., Tieu, A.K., de Boer, F.G., Ma, B., Yuen, W.Y.D.: Toward a heuristic optimum design of rolling schedules for tandem cold rolling mills. *Eng Appl Artificial Intell* 13(4), pp. 397–406 (2000)
8. Nolle, L., Zelinka, I.: SOMA applied to Optimum Work Roll Profile Selection in the hot rolling of wide steel. In: *17th European Simulation Multiconference*, Nottingham, UK (2003)
9. Nolle, L.: SASS applied to optimum work roll profile selection in the hot rolling of wide steel. *Knowledge-Based Systems* 20(2), pp. 203–208 (2007)
10. Pholdee, N., Bureerat, S., Park, W.W., Kim, D.K., Im, Y.T., Kwon, H.C., Chun, M.S.: Optimization of Flatness of Strip during Coiling Process based on Evolutionary Algorithms. *Int J Precis Eng Man* 16(7), pp. 1493–1499 (2015)
11. Hernández Carreón, C.A., Fraire Huacuja, H.J., Espriella Fernandez, K., Valdez Castilla, G., Mancilla Tolama, J.E.: Application of Genetic Algorithms to Strip Hot Rolling Scheduling. In: Corchado, E. y col. (eds.) *Innovations in Hybrid Intelligent Systems*, vol. 44. *Advances in Soft Computing*, pp. 247–254. Springer, Berlin (2007)
12. Hernández Carreón, C.A., Fraire-Huacuja, H.J., Fernandez, K.E., Castilla-Valdez, G., Mancilla Tolama, J.E.: A Model to Minimize the Hot Rolling Time of a Steel Slab Considering the Steel Chemical Composition. In: Corchado, J.M. y col. (eds.) *International Symposium on Distributed Computing and Artificial Intelligence 2008*, vol. 50. *Advances in Soft Computing*, pp. 471–480. Springer, Berlin (2009)
13. Espriella, K: *Optimización mediante algoritmos genéticos: aplicación a la laminación en caliente*. Tesis de Maestría, Instituto Politécnico Nacional, CICATA Unidad Altamira (2008)

Análisis y comparación de diferentes métodos de reconstrucción de árboles semánticos

Ramón Rivera Camacho, Ricardo Barrón Fernández, Hiram Calvo

Instituto Politécnico Nacional,
Centro de Investigación en Computación, Ciudad de México,
México

ulfrheimr@gmail.com,rbarron@cic.ipn.mx,hcalvo@cic.ipn.mx

Resumen. Los procesos modernos enfocados al procesamiento de lenguaje natural basados en aprendizaje profundo se apoyan en varias etapas de pre-entrenamiento. Sin embargo, hay ocasiones en las que la complejidad de las tareas requieren, después de los escenarios antes mencionados, un procesamiento adicional. En la literatura se menciona la estructuración por medio de árboles semánticos como uno de los métodos adicionales más prolíficos. En el presente trabajo comparamos los métodos de construcción de árboles semánticos por el método de duplas ordenadas y, basados en el analizador sintáctico de Stanford. Analizamos los procesos de reconstrucción de tuplas individuales y de estructuras completas para comparar la eficiencia de los métodos a través de la información semántica que cada uno puede mantener.

Palabras clave: Procesamiento de lenguaje natural, construcción de árboles semánticos, autoencoder, Stanford parser.

Analysis and Comparison of Different Methods of Reconstruction of Semantic Trees

Abstract. The addressing of natural language processing tasks by modern techniques, inspired in deep learning, rely on different pre-training stages. Nevertheless, some problems need an additional processing, after the referred scenarios, because of the complexity they exhibit. Structuring of semantic trees is one of the most prolific methods in the literature for this purpose. In this paper we compare the methods of construction of semantic trees by the method of ordered tuples, and based on the Stanford parser. We analyze the reconstruction of individual tuples and the construction of complete structures in order to compare the efficiency of both methods. We focus the measuring process in the amount of semantic information both methods can hold.

Keywords: Natural language processing, semantic tree construction, autoencoder, Stanford parser.

1. Introducción

Los métodos de aprendizaje modernos, entre los que se incluyen los basados en aprendizaje profundo, intentan omitir el tratamiento manual de los rasgos iniciales de algún problema realizando varias fases de pre-entrenamiento no supervisado. Se ha probado que estas fases proveen buenas distribuciones iniciales de los datos capturando las dependencias entre los parámetros de los mismos [3]. El uso de estos métodos dentro del ámbito del procesamiento de lenguaje natural (NLP, por sus siglas del inglés *Natural Language Processing*) no es la excepción.

Una de las técnicas más utilizadas en la fase de pre-entrenamiento es la vectorización de palabras, en la cuál cada palabra dentro del *corpus* es representada como un vector de n dimensiones. Un primer enfoque, basado en lexicones, consiste en tomar una palabra y obtenerla por medio de su respectivo identificador simbólico. Este modelo presenta varias limitaciones, la principal de ellas es la dispersión ya que la representación tendría el número de dimensiones del tamaño del vocabulario y solamente una casilla con información significativa; fenómeno conocido como *one-hot representation*.

Más aún, aquellas palabras que raramente aparezcan en el *corpus* serán escasamente estimadas y el modelo no será capaz de lidiar con palabras que no estuvieron presentes en la etapa de entrenamiento [12].

Es deseable que las palabras nos demuestren la relación con su contexto, los modelos de incrustación de palabras (*Word embeddings models*, en inglés) proveen de dicha información. Después del proceso de incrustación, una palabra a la par de su contexto puede ser representada por un vector denso de dimensión n . Las implementaciones más conocidas de éstos modelos son las basadas en redes neuronales profundas [2] o modelos *skip-gram* [7].

Inclusive aún con la presencia de varios escenarios de pre-entrenamiento, algunos problemas requieren del desarrollo de metodologías de estructuración de la información para tener un comportamiento predecible de los datos sobre la información ya distribuída coherentemente. Éstas metodologías han obtenido resultados favorables con técnicas de construcción de árboles semánticos, los cuales mantienen la máxima semántica posible a partir de reconstrucción de tuplas ordenadas generadas por modelos de incrustación de palabras [4, 10].

En dichos trabajos se presenta la pauta de una posible estructuración por medio analizadores sintácticos (*Parsers*, en inglés) tradicionales, si bien es correcto acentuar que estos mejoran sustancialmente las tareas tradicionales del NLP como el conteo de palabras o n -gramas [9], ninguna comparación formal entre las propiedades de los métodos anteriores ha sido provista.

El presente artículo se divide en las siguiente secciones. En la sección 2 se describen las propuestas de estructuración de árboles semánticos basadas en técnicas de aprendizaje profundo para ambos casos, duplas ordenadas y basadas en analizadores sintácticos. En la sección 3 se presentan los resultados de la evaluación de medidas que representen la cantidad de información semántica retenida. Finalmente, en la sección 4, se presentan las conclusiones de los resultados obtenidos.

2. Propuesta

El objetivo principal del presente trabajo es presentar una técnica para medir y comparar la pérdida semántica en la tarea de construir un árbol semántico por los métodos de analizadores sintácticos y de duplas ordenadas. Al generar estructuras semánticas expresamos la relación de las palabras en base a una función de energía, es posible medir la pérdida de energía de manera no supervisada construyendo estas estructuras y después intentando obtener los datos iniciales, un proceso de reconstrucción.

Aquellas construcciones que presenten menor pérdida también serán las que más fácilmente serán reconstruidas y por ende, las que mayor coherencia o información semántica mantengan. Para la medición nos apoyamos en modelos de energía (EBMs, por sus siglas del inglés *Energy Based Models*) que capturan dependencias asociando una energía escalar a cada configuración de un conjunto de variables y, cuyo entrenamiento y aprendizaje puede ser utilizado como una alternativa a modelos probabilísticos para labores de predicción, clasificación y toma de decisiones [5].

En específico utilizamos *autoencoders* (AE), ya que son entrenados con mayor facilidad y rapidez en comparación a sus contrapartes basados en métodos probabilísticos [11]. Además, son ampliamente utilizados como bloques de construcción para el entrenamiento de redes profundas [1].

Este proceso se realiza en dos etapas [6, 13]: la codificación, en la cual se mapea de manera determinista el vector de entrada a una representación intermedia por medio de una función no lineal y ; el proceso inverso, en el cual se mapea de regreso la representación intermedia a un vector reconstruido, en forma similar por una función no lineal.

Debido a que la pérdida de información semántica está regida por la reconstrucción del árbol, se comparará el desempeño en este escenario para ambos métodos.

En primera instancia, se construyeron los árboles semánticos por medio de tuplas ordenadas con el proceso descrito por Huang [4] y Socher [10], de longitud dos. Las frases son separadas en palabras y después unidas en pares. El resultado es un proceso de presentación-codificación-reconstrucción con vectores de incrustación de dimensiones $[1 \times n] \rightarrow [1 \times 2n] \rightarrow [1 \times n]$, respectivamente. Repetimos el proceso hasta obtener una sola dupla que representa el árbol semántico completo. Este proceso se muestra en la Figura 1-a.

Para el segundo acercamiento utilizamos el Stanford parser¹ para obtener árboles de dependencias. Cabe mencionar que este analizador no produce estrictamente árboles binarios, por lo tanto para la construcción de duplas se utiliza un algoritmo basado de DFS como se muestra en la Figura 1-b. De igual manera, el proceso de codificación-reconstrucción mantiene el formato de dimensionalidad $[1 \times n] \rightarrow [1 \times 2n] \rightarrow [1 \times n]$, para labores de comparación.

En la Figura 2 se contrastan los métodos de construcción de estructuras de máxima semántica. Mientras que en 2-a la construcción de árboles esta

¹ <http://nlp.stanford.edu/software/lex-parser.shtml>

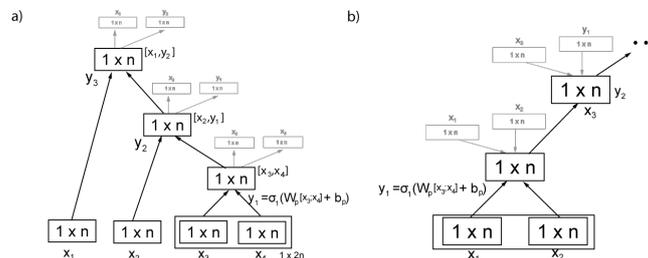


Fig. 1. Construcción de árboles semánticos por métodos para **a)** Duplas ordenadas de longitud 2 y **b)** Analizador sintactico de Stanford basado en algoritmo DFS.

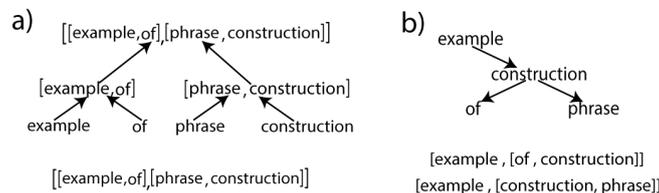


Fig. 2. Ejemplo de construcción semántica para una frase de prueba para los métodos **a)** Duplas ordenadas y **b)** Analizador sintactico.

basada en el ordenamiento semántico de las palabras, dando como resultado una dupla representativa. En 2-b se siguen las reglas pre-establecidas del analizador sintáctico, generando más de una dupla de expresión. A medida que se presenten nuevas frases los modelos de duplas resultantes tendrán mejor desempeño para aquellas construcciones que sean observadas con mayor frecuencia.

3. Experimentos y resultados

Para la construcción de vectores de incrustaciones de palabras optamos por el método basado en el algoritmo *CBOW skip-gram* propuesto por Mikolov, ya que genera mejores resultados en cuanto a exactitud y velocidad de entrenamiento que otros métodos de incrustaciones para labores como POS, NER, SRL, etc. [8].

Como apoyo para este modelo utilizaremos el conjunto de datos pre-entrenado de noticias provisto por Google en su implementación *word2vec*², el cuál contiene 100 mil millones de palabras y su vocabulario es de 3 millones de palabras.

Para el entrenamiento del AE se utilizará el error cuadrático medio (MSE, por sus siglas del inglés *Mean Squared Error*) como función de costo J , para

² <https://code.google.com/archive/p/word2vec/>

cada nodo del árbol resultante T . El error total será la suma de todos los errores de cada nodo de la estructura, como se muestra en (1):

$$J^i(W, b) = \sum_{d \in T^i} \frac{1}{2} \|x_{l,r} - x'_{l,r}\|^2, \quad (1)$$

donde

- d cada dupla de la representación del árbol correspondiente a la frase \mathbf{i} ,
- $x_{l,r}$ dupla de un nodo dado,
- $x'_{l,r}$ representación de la reconstrucción de un nodo dado.

La tarea principal en la etapa de entrenamiento es minimizar la función de costo J para el conjunto de frases I dada una razón de aprendizaje λ , como se muestra en (2):

$$\min_{W, b} \sum_{i \in I} J^i(W, b) + \frac{\lambda}{2} \|W\|^2, \quad (2)$$

donde

- $i \in I$ cada una de las frases del conjunto de datos \mathbf{I} ,
- W, b matriz de pesos sinápticos y de compensadores de la red neuronal, respectivamente.

Para los experimentos utilizamos textos obtenidos del sitio de internet Experience Project³. El conjunto tiene un total de $\sim 3.3K$ frases. Utilizamos este *corpus* ya que presenta errores de sintaxis y gramaticales, además de que contiene etiquetas no textuales. El conjunto contiene 5806 palabras distintas con una relación de palabras desconocidas con respecto al modelo de referencia pre-entrenado de $\sim 9\%$, por lo tanto se espera un rendimiento menor al que se obtendría si estuviera propiamente escrito.

El pre-procesamiento de los datos es de la manera más simple posible, solamente se retiran todos los elementos no textuales y se separan frases de párrafos extensos, ninguna labor extra es realizada.

Al término de la separación de conjuntos en razón 5:1:1, tenemos un total de $\sim 2.4K$ frases para entrenamiento y de ~ 500 frases para las etapas de validación y pruebas.

La comparación está basada en dos principales medidas. La reconstrucción promedio del MSE de duplas y la reconstrucción total promedio de árboles. Mientras que la primera de ellas provee la capacidad del método para obtener valores significativos de reconstrucción a la hora de entrenar, la segunda indica la eficiencia del método para evaluar la estructura completa en etapa de pruebas.

En la Figura 3 se muestran los resultados de la reconstrucción de duplas. En este escenario el método de duplas ordenadas tiene mejor rendimiento que su contraparte. Mientras que éste reporta un mínimo de ~ 0.069 , el basado en analizadores sintácticos reporta un mínimo de ~ 0.081 , ambos restringidos hasta 50 épocas. Con lo que también se infiere una convergencia más rápida.

³ <http://www.experienceproject.com/>

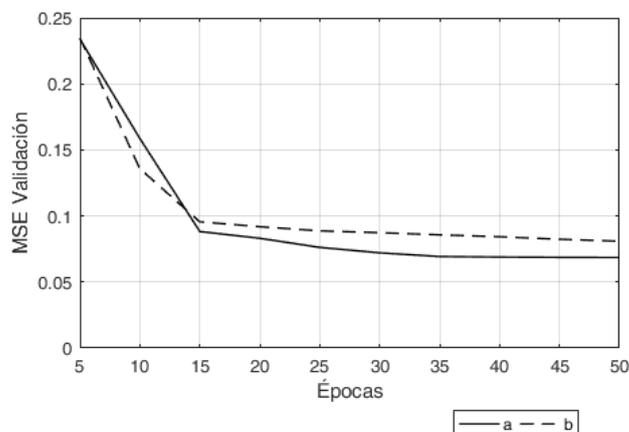


Fig. 3. MSE promedio de reconstrucción de duplas para los métodos a) duplas ordenadas y b) basado en analizadores sintácticos.

Para el escenario de reconstrucción de árboles se realiza un conjunto de 30 experimentos sobre *batches* de palabras. Con el fin de tener un rango más amplio de conjuntos de pruebas, se varían las condiciones de parada temprana y cantidad de nodos en rangos de 20 a 50 nodos. Se reportan los errores de reconstrucción total promedio para cada experimento en la Figura 4.

Después de obtener la media de todos los experimentos expuestos previamente, se aprecia que el método basado en duplas ordenadas ofrece mejores resultados para la reconstrucción total de árboles con una media de ~ 0.127 contra la media de ~ 0.171 para el acercamiento por medio de analizadores sintácticos.

Un mejor rendimiento es obtenido en ambas etapas de comparación para el método de duplas ordenadas. Para la etapa de reconstrucción de duplas se reporta un $\sim 17\%$ de mejoría con respecto al acercamiento de analizadores sintácticos, mientras que para labores de reconstrucción total de árboles la reconstrucción de duplas mejora al reportar $\sim 35\%$ contra el acercamiento de analizadores sintácticos.

4. Conclusiones

En este trabajo de investigación se compararon dos métodos de construcción de árboles, el primero basado en duplas ordenadas y el segundo, por medio del analizador sintáctico de Stanford para los escenarios de reconstrucción de duplas individuales y de error de reconstrucción de todo el conjunto de árboles en el *corpus* de prueba.

Se encontró que para ambos casos el desempeño del primer método es mejor. Estos resultados son justificables debido a la dispersión de las tuplas. Se debe recordar que el objetivo principal de un analizador sintáctico es mostrar la

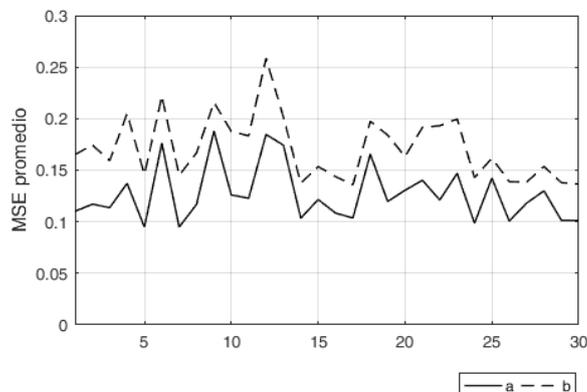


Fig. 4. MSE de reconstrucción total para rangos de parada y nodos de AE para métodos a) duplas ordenadas y b) basado en analizadores sintácticos.

estructura gramatical de una frase en específico. Al tener un conjunto de datos de prueba carente de éstas características la labor se convierte una tarea difícil para un analizador y por lo tanto, los árboles resultantes no siempre serán reconstruidos de la misma manera.

Por otra parte, ya que el método de estructuración por medio de duplas no lidia con clases gramaticales ni parecidos, a medida que los errores aparezcan serán obviados, lo cual se soporta con el conjunto restante de árboles bien estructurados.

Finalmente, se destaca que mientras el método de duplas ordenadas trabaja calculando el error mínimo por nivel para la construcción de los árboles, el método de analizador de Stanford utiliza la estructura procesada y calcula el error de reconstrucción dado el árbol completo.

Agradecimientos. Este trabajo fue realizado gracias al apoyo del Instituto Politécnico Nacional, los proyectos SIP-IPN 20162058 y 20160828 y COFAA-IPN.

Referencias

1. Bengio, Y.: Learning Deep Architectures for AI. *Foundations and Trends in Machine Learning* 2(1) (2009)
2. Collobert, R., Weston, J.: A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In: *Proceedings of the 25 th International Conference on Machine Learning* (2008)
3. Erhan, D., Bengio, Y., Courville, A., Vincent, P., Bengio, S.: Why Does Unsupervised Pre-training Help Deep Learning? *Journal of Machine Learning Research* 11, 625–660 (2010)

4. Huang, E.: Paraphrase Detection Using Recursive Autoencoder (2011)
5. Lecun, Y., Chopra, S., Hadsell, R., Ranzato, M.A., Huang, F.J.: A Tutorial on Energy-Based Learning. Predicting Structured Data, MIT Press (2006), <http://yann.lecun.com>
6. LISA Lab: Deep Learning Tutorial (2015)
7. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient Estimation of Word Representations in Vector Space (2013)
8. Schnabel, T., Labutov, I., Mimno, D., Joachims, T.: Evaluation methods for unsupervised word embeddings. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 298–307. Association for Computational Linguistics (2015)
9. Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernández, L.: Syntactic N-grams as machine learning features for natural language processing. *Expert Systems with Applications: An International Journal* 41(3), 853–860 (2014)
10. Socher, R., Pennington, J., Huang, E.H., Ng, A.Y., Manning, C.D.: Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions. In: EMNLP '11 Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 151–161 (2011)
11. Swersky, K., Ranzato, M.A., Buchman, D., Marlin, B.M., De Freitas, N.: On Autoencoders and Score Matching for Energy Based Models. In: Proceedings of the 28 th International Conference on Machine Learning (2011)
12. Turian, J., Ratinov, L., Bengio, Y.: Word representations: A simple and general method for semi-supervised learning. In: 48th Annual Meeting of the Association for Computational Linguistics (2010)
13. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A.: Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *Journal of Machine Learning Research* 11 (2010)

Modelo de la dinámica de neurotransmisores por medio de caminantes aleatorios

Jorge A. Herrera¹, Juan C. Chimal¹, Norma Sánchez Salas²

¹ Instituto Politécnico Nacional,
Centro de Investigación en Computación,
Ciudad de México, México

² Instituto Politécnico Nacional,
Escuela Superior de Física y Matemáticas,
Ciudad de México, México

jaherrera@yahoo.com.mx, jchimale@gmail.com, norma@esfn.ipn.mx

Resumen. En el presente artículo se propone modelar la dinámica de neurotransmisores por medio de caminantes aleatorios, se muestra que es posible afirmar que existe una relación de la distancia entre neuronas y la cantidad de neurotransmisores que llegan a los receptores de la neurona post-sináptica, con la finalidad de poder entender a grandes rasgos cómo funcionan las conexiones neuronales en términos de las sinápsis químicas. Concretamente, se comparan resultados experimentales de la implementación del modelo en donde se hace una cuantificación del número de moléculas que llegan a los receptores contra los resultados obtenidos por LEONID P. SAVTCHENKO AND DMITRI A. RUSAKOV [1] (corriente en función de la distancia).

Palabras clave: Automatas celulares, caminantes aleatorios, neurotransmisores.

Model of the Dynamics of Neurotransmitters by Means of Random Walkers

Abstract. In the present article we propose to model the dynamics of neurotransmitters by means of random walkers, it is possible to affirm that there is a relation of the distance between neurons and the amount of neurotransmitters that reach the receptors of the post-synaptic neuron, with The purpose of being able to understand roughly how neural connections work in terms of chemical synapses. Specifically, we compare experimental results of the implementation of the model where a quantification is made of the number of molecules that reach the receptors against the results obtained by LEONID P. SAVTCHENKO AND DMITRI A. RUSAKOV [1] (current as a function of distance).

Keywords: Cellular automaton, random walkers, neurotransmitters.

1. Introducción

Para poder entender cómo funcionan las conexiones neuronales, es necesario hablar sobre sinápsis química, misma que figura como una forma por medio de la cual las neuronas se comunican para poder enviar información e interactuar entre ellas [11,12,13].

Aunque el sistema nervioso humano ha sido objeto de estudio a través de los siglos y dentro de diversas culturas [6], aún existen demasiadas preguntas en muy diversos campos relacionados al tema de las neuropatologías, que si bien en un punto determinado podría pensarse lo contrario, pueden estar relacionadas directamente con la manera en que se llevan a cabo conexiones neuronales de forma particular. Dentro de este conjunto de preguntas se encuentran muchas pertenecientes al estudio de enfermedades variadas como el Parkinson, Alzheimer, autismo, adicciones [2], la epilepsia [3,4], etc., las cuales que hoy en día no tienen cura y a su vez no se han terminado de entender.

Después de la publicación del modelo de Hodgkin y Huxley se ha abierto un panorama más amplio en cuanto al estudio de la dinámica neuronal desde el área de las matemáticas y con la ayuda de diversas técnicas de simulación y modelado; sobre todo en términos del papel que la química neuronal juega en la respuesta eléctrica de la misma, pues resultados apuntan a que estas investigaciones parecen tener muchas respuestas por brindar aún, de esta manera, los modelos discretos como el que se presenta en este artículo no son ni serán la excepción al intentar brindar no sólo respuestas, si no métodos más sencillos y computacionalmente menos costosos para realizar experimentación comparados con algunos modelos continuos, así mismo es viable creer que los modelos computacionales que se están desarrollando puedan derivar en nuevos protocolos de tratamiento [5], lo cual figura como un gran avance en el área de las ciencias biomédicas para estudio de neuropatologías particularmente.

2. Antecedentes

2.1. Autómatas celulares

Los automatas celulares son un tipo de modelos discretos, siendo estos una versión idealizada de un sistema complejo [8], estos están conformados por células o celdas dentro de un lattice, en donde a cada paso de tiempo las células pueden mostrar diversos estados, es decir, una célula individual dentro del lattice podría adoptar determinada condición dependiendo, principalmente, del estado (de la célula en cuestión) en t_{n-1} y el estado de las células que están a su alrededor, mismas que conforman su vecindario finito.

De forma simple es posible nombrar a cada célula como c_i y así mismo al autómata como un vector $\mathbf{C} \in \mathbb{Z}$, dentro del cual cada estado de alguna célula dada, en un tiempo dado t_i , podrá ser definido como S_i contenido dentro de un conjunto $\mathcal{S} \in \mathbb{Z}$ de estados posibles para las células.

Por último se define la vecindad de una célula cualquiera en el autómata como $V_i = \{c_1, c_2, \dots, c_n\}$, la cual puede ser de diversos tipos (según las condiciones

del fenómeno que se desee modelar), de entre las más sencillas y conocidas se encuentran, por ejemplo, la vecindad de Vonon Neuman y la vecindad de Moore. A grandes rasgos la vecindad se refiere al conjunto contiguo de celdas y la posición respectiva entre cada una de ellas.

2.2. Caminantes aleatorios y difusividad

La difusión es tal vez el mecanismo más básico de transporte en la materia y se encuentra al origen de muchos procesos. Las caminatas aleatorias ofrecen un marco teórico útil para entender la difusión a un nivel microscópico [9].

De forma simple, un caminante aleatorio es la forma discreta de un proceso difusivo, en el cual la probabilidad juega un papel importante a cada paso de tiempo discreto y en la trayectoria del caminante aleatorio, la cual está conformada por todos los puntos espaciales por donde el caminante pasa desde el tiempo inicial t_0 hasta el tiempo final t_n .

Considérese que una partícula es capaz de permanecer en posiciones discretas y equidistantes que se encuentran en una línea, dicha partícula tiene la posibilidad de moverse a lo largo de esas posiciones discretas en cada paso de tiempo discreto t_i ya sea en dirección izquierda o derecha con igual probabilidad $\frac{1}{2}$, entonces la probabilidad $P_{t_i}(pn)$ de que el caminante se encuentre en el sitio pn al instante t_i se define por medio de:

$$P_{t_i}(pn) = \frac{1}{2}P_{t_{i-1}}(pn - 1) + \frac{1}{2}P_{t_{i-1}}(pn + 1). \quad (1)$$

Assumiendo que el caminante se encuentra en la posición $pn = 0$ en el tiempo inicial $t_0 = 0$ se puede describir la probabilidad de que el caminante haya dado n pasos al tiempo t_i a la derecha y $t_i - n$ pasos a la izquierda mediante la siguiente igualdad:

$$P_{t_i}(pn) = \binom{t_i}{n} \frac{1}{2^{t_i}}. \quad (2)$$

Tomando en cuenta tiempos discretos grandes ($t \gg 1$) y haciendo uso de la aproximación de Stirling, tenemos como resultado la distribución gaussiana:

$$P_{t_i}(pn) = \frac{1}{\sqrt{4\pi Dt}} e^{-\frac{n^2}{4Dt}}. \quad (3)$$

Considerando un tiempo continuo y una probabilidad igual de saltar hacia la derecha o izquierda αdt además de tomar un intervalo finito de tiempo $[t, t + dt]$ y una tasa $\alpha = 1$ se puede escribir una ecuación para la evolución de $P_t(pn)$:

$$\frac{\partial P_t(pn)}{\partial t} = P_t(np + 1) - 2P_t(np) + P_t(np - 1). \quad (4)$$

Obteniendo la transformada de Fourier discreta de la expresión anterior se obtiene la siguiente ecuación:

$$\frac{\partial P_t(pn)}{\partial t} = [e^{ik} + e^{-ik} - 2]P(k, t). \quad (5)$$

Integrando la expresión anterior dadas las condiciones iniciales donde $P_{t=0}(n) = \delta_{n,0}$ implica que: $P(k, t = 0) = 1$ se puede obtener el siguiente resultado:

$$P(k, t) = e^{2(\cos k - 1)t}. \tag{6}$$

El cual puede ser reducido mediante identidades trigonométricas y las funciones de Bessel dando como resultado:

$$\frac{\partial P(x, t)}{\partial t} = D \frac{\partial^2 P(x, t)}{\partial x^2}. \tag{7}$$

Ahora bien, de aquí se puede deducir la expresión para el desplazamiento cuadrático medio $\langle x^2 \rangle$ (segundo momento de la distribución $P(x, t)$) mediante la expresión siguiente, misma que muestra a $\langle x^2 \rangle$ con una tendencia de crecimiento con el paso del tiempo:

$$\langle x^2 \rangle(t) = \int_{-\infty}^{\infty} x^2 P(x, t), dx = 2Dt. \tag{8}$$

Tomando en cuenta una suma de posiciones que contiene N pasos:

$$x = \sum_{np=1}^N u_n. \tag{9}$$

Con u_n generados a partir de una distribución de probabilidad $p(u)$ obtenemos (1) en una forma general:

$$P(x, N) = \int_{-\infty}^{\infty} P(x', N - 1)p(x - x'), dx'. \tag{10}$$

La ecuación anterior puede resolverse mediante la simplificación de la misma por transformada de Fourier y considerando algunas condiciones iniciales de las cuales, se sabe que:

$$P(x, N) = \frac{1}{\sqrt{2\pi N \langle u^2 \rangle}} e^{-\frac{x^2}{2N \langle u^2 \rangle}}. \tag{11}$$

Sustituyendo x por un vector posición r, la expresión anterior puede generalizarse a espacios d-dimensionales, como en la siguiente expresión:

$$P(r, t) = \frac{1}{(4\pi Dt)^{d/2}} e^{-\frac{r^2}{4Dt}}, \tag{12}$$

de donde se obtiene:

$$\langle r^2 \rangle = 2dDt. \tag{13}$$

3. Modelo propuesto

Al ser liberados los neurotransmisores viajan por el espacio sináptico de forma tal que avanzan hacia los receptores mediante un movimiento aleatorio, dada esta condición se propone modelar la dinámica de las moléculas por medio de caminantes aleatorios.

El autómata celular fue implementado a través de código escrito en Matlab®, las gráficas de los resultados obtenidos por medio de experimentación computacional de la cantidad de neurotransmisores que llegan a los receptores en función de la distancia entre las dendritas tienen una forma similar a los resultados de experimentación en [1].

3.1. Lattice

Se modeló el lattice mediante una matriz de $2 \times n$ la cual contiene los límites de este, y la información sobre la posición ocupada por una célula dada. Por consiguiente, los estados de cada célula dentro del lattice simplemente se reducen a las siguientes condiciones: un espacio (x, y) dentro del lattice ocupado por una célula (célula viva) será registrado en la matriz v_{ca} , por consiguiente un espacio descoupado no será tomado en cuenta en los registros de coordenadas de la matriz v_{ca} (como célula viva) y por tanto no desempeñará ningún papel en la dinámica del modelo.

El lattice ha sido pensado como un medio espacial, y las posiciones de las dendritas toman el papel de límites en el medio sináptico de tal forma que las moléculas no pueden moverse, por ejemplo, más allá de la posición $x = d$ de la dendrita post-sináptica ni hacia atrás de la posición en $x = d_0$ de la dendrita pre-sináptica. Es decir, las posiciones en x de las dendritas, juegan un papel de barrera para las células, antes y después de los cuales las moléculas no podrán hacer movimiento alguno a menos que estas logren alcanzar una posición $y = R$ (véase Fig. 1) determinada, más abajo o más arriba de los parámetros propuestos en el programa determinados como en [1], o bien alcancen la posición (d, y) de un receptor.

Es importante señalar que el modelo del lattice no permite que dos o más células c_i del autómata ocupen un mismo lugar (coordenada) (x, y) dentro del lattice definido por la matriz v_{ca} para asegurar dicha condición se lleva un proceso de control el cual permite que una célula c_i conozca su vecindad y tenga conocimiento sobre las posibilidades de movimiento que tiene.

3.2. Movimiento

Seguidas de las consideraciones sobre el lattice, se puntuarán aquellas relacionadas con el movimiento. De forma particular, se hablará sobre cómo las células dentro del lattice eligen una "dirección" hacia la cual avanzar en cada instante de tiempo t_i .

Dado que el fenómeno a modelar sigue un comportamiento de movimiento aleatorio 2-dimensional, el programa se apegará esta dinámica de las células en

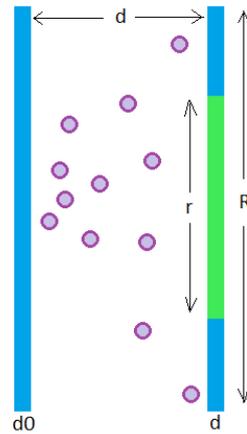


Fig. 1. Representación gráfica del modelo, particularmente el lattice, el cual se extiende desde d_0 hasta d y toda la longitud de R , de la misma manera que en [1] el modelo presentado en este trabajo tiene una zona activa (representada por el área verde), los puntos morados representan a las células del autómata y por último d es la distancia entre las dendritas de las neuronas (distancia de hendidura sináptica).

el autómata mediante un vecindario del tipo Von Neuman, esto implica que las células pueden tener 4 posibilidades de movimiento: arriba, abajo, derecha e izquierda, y la dirección será elegida mediante una gestión de probabilidad, para ello será necesario hacer uso de números pseudoaleatorios generados por la función *rand()* de Matlab®, dicho proceso se verá de forma más detallada más adelante.

En resumen, el algoritmo procede de la siguiente manera: genera un número aleatorio entre 0 y 1 denominado J , si $J \leq 0.25$ entonces la célula en el autómata c_i se moverá hacia arriba pero si $0.25 < J \leq 0.5$ c_i se moverá una posición abajo o, dado $0.5 < J \leq 0.75$ c_i se desplazará hacia la derecha o bien si $0.75 < J \leq 1$ c_i tendrá un desplazamiento hacia la izquierda.

3.3. Receptores

En la barrera de la dendrita post-sináptica encontraremos los receptores, mismos que están distribuidos a lo largo de una región activa como en [1]. Los receptores a grandes rasgos son tomados como coladeras en el modelo (véase Fig. 2), y tienen una posición (d, y) , donde d es la distancia a la cual se encuentra la dendrita post-sináptica.

El papel desempeñado por los receptores es el de absorber a las células que se encuentren en sus mismas posiciones (d, y) , entonces, aunque los caminantes (células) del autómata lleguen al límite espacial d si estos no coinciden estrictamente con alguna posición (d, y) de un receptor en particular, seguirán

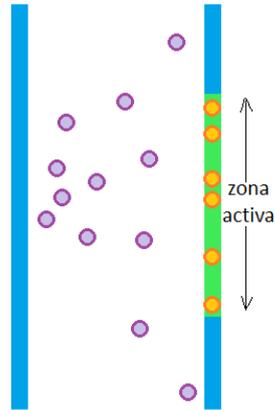


Fig. 2. Representación gráfica de cómo está considerado el límite derecho de lattice (dendrita post-sináptica), el cual contiene receptores distribuidos aleatoriamente a lo largo de la zona activa (zona verde) a la espera de que alguna molécula (puntos morados) llegué a alguna posición (d, y) para ser absorbida.

moviéndose a través de la hendidura y por consecuencia no formarán parte del conteo de células absorbidas por el receptor.

Es importante mencionar que todos los receptores tienen diferentes coordenadas en y lo cual implica directamente que no será posible que dos receptores se encuentren en la misma posición (d, y) a lo largo de la ejecución del programa, así mismo el algoritmo se encarga de hacer que estos estén distribuidos de manera aleatoria y normal a lo largo de la zona activa.

Por último, es importante mencionar que el tamaño de la zona activa es una variable fundamental en el desempeño de la tarea de los receptores en el algoritmo ya que las variaciones más simples del tamaño de esta genera resultados totalmente diferentes.

3.4. Pseudocódigo

En términos generales el pseudocódigo presentado en este apartado da una idea general de cómo funciona el programa, además de que mediante el mismo es posible hacer no sólo modificaciones al momento de realizar simulación, sino también nuevas implementaciones a lo largo del mismo con el objetivo de tener al alcance la posibilidad hacer diversos tipos de experimentación según sean requeridos.

Pseudocódigo para la Simulación de la dinámica de Neurotransmisores:

```
limpia memoria
comentario: definicion de variables
define ni; comentario: moleculas iniciales
define D; comentario: pasos por tiempo de evolucion
define d; comentario: espacio de la hendidura sinaptica
define vi; comentario: numero de vesiculas iniciales
define ri; comentario: radio inicial de vesiculas
define recp; comentario: numero receptores en zona activa
comentario: inicializacion de variables y/o arreglos
inicializa x[ci]; comentario:vector para registro de posicion
inicializa y[ci]; comentario:vector para registro de posicion
inicializa ci en un radio ri
inicializa recp a lo largo de zona activa
registra posciones ci en x e y
inicializa recp receceptores aleatorios en zona activa
para i=1 hasta D
  para j=1 hasta ci
    genera un numero aleatorio J;
    si J<=0.7 comentario: movimiento derecha
      si x(i)<=d&&(d-1)<=x(i)
        genera un numero aleatorio J;
        si no se cumple lo anterior
          x(i)=x(i)+1;
          y(i)=y(i);
        end%condiciones de frontera en pos. x
      pero si (J>0.7)&&(J<=0.8) comentario: movimeinto izquierda
        x(i)=x(i)-1;
        y(i)=y(i);
      pero si (J>0.8)&&(J<=0.9) comentario: movimiento arriba
        x(i)=x(i);
        y(i)=y(i)+1;
      si no se cumple lo anterior comentario: movimiento abajo
        x(i)=x(i);
        y(i)=y(i)-1;
      end
    eliminar registro en (x,y) de moleculas que llegan a receptores
  fin para j=1 hasta ci
fin para i=1 hasta D
Muestra datos derivados de la simulacion actual
fin del programa
```

En el pseudocódigo se encuentra una sentencia de control if-else (dentro de la gestión de J para el movimiento hacia la derecha) encargada de dar las condiciones de frontera, a grandes rasgos esta condición está pensada para evitar que los caminantes pasen más allá de la dendrita post-sináptica si es que en el

tiempo t_i no han alcanzado una altura (posición y) suficiente, entonces y como se mencionó anteriormente, la dendrita post-sináptica se modeló como si fuese un muro con algunos huecos (los cuales juegan el papel de receptores) distribuidos a lo largo de una zona activa como en el modelo de [1] y con diferentes posiciones.

Una vez que se cumpla la condición en donde una o más células son absorbidas por los receptores, los registros de las mismas serán eliminados de la matriz v_{ca} , esto implica que tendrá menos elementos en la siguiente entrada de los bucles y las moléculas absorbidas no desempeñarán ningún papel en la dinámica del algoritmo, es decir, serán eliminadas por completo de los registros en el programa y pasarán a ser estrictamente parte de la cuantificación para los resultados del mismo.

4. Experimentos y resultados

A continuación se abordará lo relacionado a los datos obtenidos mediante la experimentación con el modelo ya implementado en Matlab[®], comenzaremos presentando los resultados obtenidos por [1], En los cuales, se puede apreciar en las gráficas una función de la corriente respecto de la distancia.

En general estos datos demuestran que el pico de corriente estará en función de la separación entre dendritas, además de que habrá un intervalo donde la corriente será máxima; por consiguiente también habrá intervalos donde la amplitud de corriente decaiga.

El modelo que se propone en este trabajo muestra que al momento de hacer una cuantificación de los neurotransmisores que llegan a los receptores post-sinápticos también se obtendrán curvas con formas muy similares a las de [1].

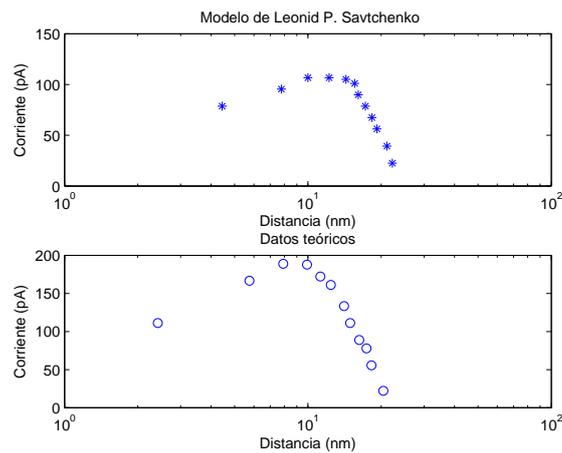


Fig. 3. Datos obtenidos por [1] mediante su modelo propuesto comparados con datos teóricos.

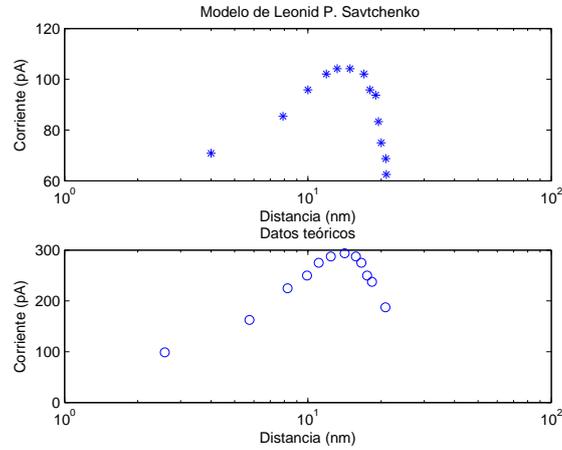


Fig. 4. Datos obtenidos por [1] mediante su modelo propuesto comparados con datos teóricos.

Por otro lado, se presentan los resultados obtenidos por medio de experimentación con el modelo aquí propuesto, en el cual se encontrarán gráficas en donde el número de células que llegan a los receptores está en función de la distancia de separación entre dendritas.

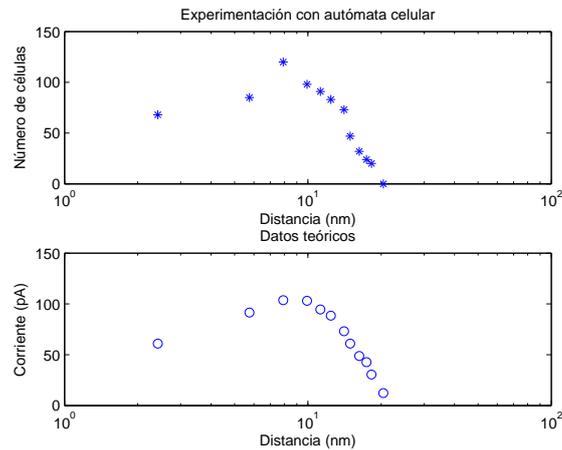


Fig. 5. Datos obtenidos por el modelo propues comparados contra datos teóricos.

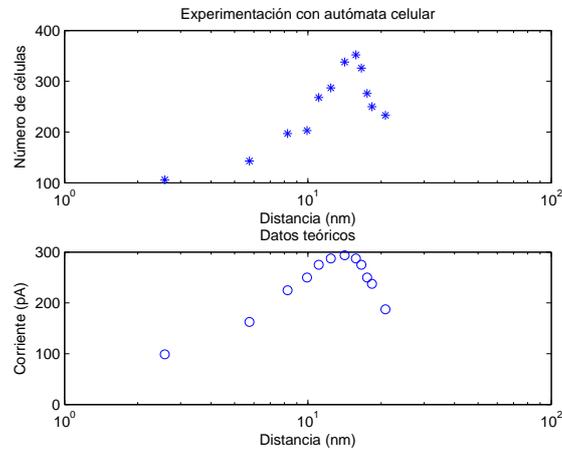


Fig. 6. Datos obtenidos por el modelo propuesto comparados contra datos teóricos.

Estos últimos datos que se presentan en las gráficas anteriores (Fig 5. y Fig 6) y que se obtuvieron a través de la experimentación muestran que aún para la cantidad de moléculas de neurotransmisores que llegan a los receptores post-sinápticos es importante tomar en cuenta la distancia de separación entre dendritas, los datos presentados anteriormente tienen una característica importante: al igual que en las gráficas para los datos del modelo de [1] existe una distancia a la cual el número de moléculas de neurotransmisores que llegan a los receptores es máximo.

Lo anterior bien puede apuntar hacia una relación en la que se podría mostrar que la distancia y el número de moléculas en los receptores juegan un papel determinante en el disparo de corriente producido en una sinápsis.

5. Trabajo futuro

La presente investigación tiene posibilidades abiertas para que se desempeñen más experimentaciones, dentro de las que podrían resaltarse principalmente aquellas que puedan ampliar el modelo y/o generalizarlo, por ejemplo, a una modelación de una serie de neuronas liberando neurotransmisores entre sí.

Por otro lado también queda abierto el estudio sobre la relación entre la concentración de moléculas que llegan a la neurona post-sináptica y el pico de corriente producido por los mismos, de hecho, es viable formar una función que pueda describir este fenómeno a partir de los datos generados por el autómatas.

Así mismo, el modelo ofrece la posibilidad de hacer experimentación con la respuesta eléctrica obtenida mediante diversos parámetros en el autómatas, o agregados a este, con la finalidad de poder estudiar diferentes dinámicas y, derivadas de estas, diferentes respuestas eléctricas.

Computacionalmente existen también posibilidades para trabajo futuro, sobre todo en el tema de optimización del código el cual podrá ser llevado a cabo a través el uso de computo paralelo que ofrece la toolbox dedicada de Matlab®, implementando el algoritmo en algún otro lenguaje o bien hacer uso de la arquitectura CUDA (en algún equipo que lo permita); esto principalmente con la finalidad de realizar pruebas más complejas sobre todo en cuanto al manejo de un mayor número de células en el autómata celular en un menor tiempo de ejecución.

6. Discusión y conclusiones

Los resultados de las experimentaciones muestran que la cantidad de moléculas que llegan a los receptores en la dendrita post-sináptica está directamente relacionada con la separación entre las dendritas (tamaño de hendidura sináptica).

Se logró modelar y simular computacionalmente la dinámica de las moléculas de los neurotransmisores dando como resultado una comparación de datos exitosa, lo cual implica que la implementación del modelo funciona de acuerdo a lo que se esperaba, aunque cabe mencionar que las experimentaciones previas apuntan que el modelo bien podría ser adaptado para modelar un fenómeno sináptico más complejo donde, por ejemplo, un número inicial de vesículas > 1 con radios iguales o diferentes sean las nuevas condiciones iniciales.

Esta investigación derivó en un nuevo modelo discreto mediante el cual es posible hacer experimentación con los parámetros dados y propuestos de manera sencilla y rápida. Además el modelo propuesto está totalmente abierto a cambios con todas y cada una de las variables, lo cual implica que esta flexibilidad al momento de hacer modificaciones pueden permitir modelar no sólo sinápsis más complejas, si no también experimentos donde a este proceso se le puedan agregar perturbaciones las cuales apunten a intentar responder preguntas específicamente sobre enfermedades, que estén relacionadas con procesos neuroquímicos (más específicamente sinápsis) anómalos, ahorrando tiempo y brindando una herramienta de experimentación confiable.

Agradecimientos. Agradecemos a Germán Téllez del CIC-IPN por los comentarios y correcciones hechas a la sección teórica de este artículo.

Referencias

1. Savtchenko, L.P., Rusakov, D.A.: The optimal height of the synaptic cleft. PNAS, 104, pp. 1823–1828 (2006)
2. Spronsen, M.V., Hoogenraad, C.C.: Synapse Pathology in Psychiatric and Neurologic Disease Current. Neurology and Neuroscience Reports, 10(4), pp. 207–214 (2010)

3. Badawya, R.A.B., Harveyc, A.S., Macdonella, R.A.L.: Cortical hyperexcitability and epileptogenesis: Understanding the mechanisms of epilepsy - Part 1. *Journal of Clinical Neuroscience*, 16, pp. 355–36 (2009)
4. Badawya, R.A.B., Harveyc, A.S., Macdonella, R.A.L.: Cortical hyperexcitability and epileptogenesis: Understanding the mechanisms of epilepsy - Part 2. *Journal of Clinical Neuroscience*, 16, pp. 485–500 (2009)
5. Stefanescu, R.A., Shivakeshavan, R.G., Talathi, S.S.: Computational models of epilepsy. *Seizure*, 21, pp. 748–759 (2012)
6. Carod-Artal, F.J., Viquez-Cabrera, C.B.: Paleopatología neurológica en las culturas precolombinas de la costa y el altiplano andino (II). Historia de las trepanaciones craneales, *Rev Neurol*. 38, pp. 886–894 (2004)
7. Bernard, C.: Alterations in synaptic function in epilepsy. <https://www.ncbi.nlm.nih.gov/books/NBK98161/>
8. Mitchel, M.: *Complexity A Guided Tour*, Oxford University press (2009)
9. Boyer, D.: *Fronteras de la Física en el Siglo XXI. Procesos difusivos: de moléculas a animales*, UNAM, México (2013)
10. Jonas, P., Major, G., Sakmann, B.: Quantal components of unitary EPSCs at the mossy fibre synapse on CA3 pyramidal cells of rat hippocampus. *J Physiol*, 472, pp. 615–663 (1993)
11. Purves, D., Augustine, G.J., Fitzpatrick, D.: *Neuroscience*, 2nd edition. Sunderland (MA), Sinauer Associates (2001)
12. Neurons, Synapses, Action Potentials, and Neurotransmission http://www.mind.ilstu.edu/curriculum/neurons_intro/neurons_intro.php
13. Thiele, S., Nash, J.: *Chemical Synapses*. <http://www.els.net/WileyCDA/ElsArticle/refId-a0000037.html>
14. Gilat, A.: *MATLAB: Una introducción con ejemplos prácticos*, Reverte (2006)
15. Strogatz, S.H.: *Nonlinear Dynamics And Chaos: With Applications To Physics, Biology, Chemistry, and Engineering (Studies in Nonlinearity)* 1st edition, AddisonWesley (1994)
16. Weisstein, E.W.: *Elementary Cellular Automaton*. <http://mathworld.wolfram.com/ElementaryCellularAutomaton.html>
17. Rangel-Mondragon, J.: *A Catalog of Cellular Automata*. <http://library.wolfram.com/infocenter/MathSource/505/>

Comparación de clasificadores para el reconocimiento de notas musicales

Omar Velázquez López, José Luis Oropeza Rodríguez, Sergio Suárez Guerra

Instituto Politécnico Nacional, Centro de Investigación en Computación, Ciudad de México, México

{joropeza, ssuarez}@cic.ipn.mx

Abstract. Este artículo describe el conjunto de experimentos realizados para obtener el reconocimiento de 60 notas musicales de un piano digital por medio de técnicas de procesamiento digital de señales y clasificadores. Para la etapa de técnicas de procesamiento digital de señales se utilizaron: Frecuencia fundamental, coeficientes Cepstrales en la Frecuencia de Mel y Cepstrales de Mecánica Coclear. Para los clasificadores se consideraron Modelos Ocultos de Markov usando HTK (Hidden Markov Model Toolkit), Modelos de Mixturas Gaussianas, Redes Neuronales Artificiales y Cuantificación Vectorial. El corpus creado para este trabajo consistió de 720 notas comprendidas en un rango de cinco octavas. Se usaron 12 archivos para cada una de las 60 notas a reconocer, las cuales se grabaron a una velocidad 70 bpm (beats per minute). Los mejores resultados se obtuvieron al aplicar coeficientes CMCC como técnica y Modelos Ocultos de Markov como clasificador (100% de reconocimiento en cada octava). Igualmente se obtuvo el 100% de reconocimiento por octava al usarse coeficientes MFCC como técnica y Cuantificación Vectorial como clasificador.

Palabras clave: Transcripción automática de música, procesamiento digital de señales, reconocimiento de patrones.

Comparison of Classifiers for Recognition of Musical Notes

Abstract. This paper describes the set of experiments performed for recognition of 60 musical notes of a digital piano using digital signal processing techniques and classifiers. In the stage of digital signal processing techniques we used: Fundamental frequency (Pitch), Mel-Frequency Cepstral Coefficients (MFCC) and Cochlear Mechanics Cepstral Coefficients (CMCC). The implemented classifiers were: Hidden Markov Models (HMM) using HTK (Hidden Markov Model Toolkit), Gaussian Mixture Models (GMM), Neuronal Networks and Vector Quantization (VQ). The created corpus for this work consists of 720 notes included in a range of five octaves. We used 12 files for each one of the 60 notes for recognition, which were recorded in a speed of 70 beats per minute. The best results were obtained applying CMCC as technic and HMM as classifier (100%

of recognition in each octave). Also we obtained the 100% of recognition for each octave using MFCC as technique and VQ as classifier.

Keywords: Automatic transcription of music, signal digital processing, pattern recognition.

1. Introducción

El reconocimiento de notas musicales forma parte del problema de realizar transcripción automática de música, el cual es el proceso de convertir una señal de audio a la notación musical convencional mediante métodos computacionales y electrónicos. Para dicha tarea, es necesario realizar la extracción de características de la señal de audio utilizando técnicas de procesamiento digital de señales. La información contenida en una señal de música es de múltiples dimensiones, comúnmente está descrita por su tono, duración y timbre.

El tono de una señal de música en términos de análisis de señales, se conoce como frecuencia fundamental. Existe la posibilidad de que dos instrumentos estén ejecutando la misma nota y por lo tanto el mismo tono, pero lo que la diferencia entre uno y el otro es su timbre.

2. Características y generalidades

Existen tres maneras posibles de resolver el problema de transcripción automática de música desde el punto de vista como plataforma: software (no en tiempo real), hardware (en tiempo real) o híbridamente. En cuanto a software, en [3] se describen las diferentes técnicas de procesamiento digital de señales clásicas y avanzadas; y los tipos de clasificadores utilizados para la solución de este problema. Por otro lado, las soluciones propuestas en el estado del arte para resolver el problema de transcripción en tiempo real mediante hardware han demostrado resultados desde diferentes enfoques, tal como se menciona a continuación.

[4] consiste en un algoritmo de búsqueda del tono fundamental de una señal de audio mediante un microcontrolador. Por otra parte [1] describe un sistema de transcripción de música en tiempo real para guitarras acústicas, éste consiste en tres componentes principales de hardware que se colocan en la guitarra y en una mano del intérprete: estampas sensores adheridas al diapasón, un recolector con pastillas en la boca de la guitarra y un anillo colocado en el dedo de la mano derecha del intérprete.

Tabla 1. Porcentaje de notas transcritas correctamente en [4].

| Duración de Notas Ejecutadas | Tamaño de Ventana Estática | | Tamaño de Ventana Dinámica |
|------------------------------|----------------------------|--------|----------------------------|
| | 20 ms | 100 ms | |
| Corta duración | 67% | 50% | 80% |
| Larga duración | 70% | 75% | 92% |

En el caso de [4] se hace una clasificación de las notas entre corta y larga duración. No se especifica cuales corresponden a una clase, pero se puede suponer que se trata de notas corcheas para el primero, y notas negras para el segundo. Tampoco se hace referencia al alcance de notas reconocidas en cuanto a frecuencia o tono. Por otro lado en [1] al tratarse de un sistema que consta de sensores en cada traste de una guitarra acústica puede suponerse que el alcance de notas comprende todas aquellas que están contenidas en el diapasón de la guitarra. Sin embargo, no se menciona un porcentaje de notas reconocidas por el sistema.

3. Un sistema de transcripción automática de música

Es posible resolver el problema de transcripción automática de música utilizando un transcriptor, el cual consta de dos partes: *front end* y *back end*. La primera contiene los algoritmos encargados de la extracción de características de la señal de música y el segundo contiene los clasificadores que reconocen las notas interpretadas. Como primer paso es necesario entrenar el *back end* por medio de los datos procesados en el *front end*, para ello se lleva a cabo el proceso descrito por el diagrama de bloques de la Fig. 1. Una vez entrenado el clasificador, se procede a probar el transcriptor y reconocer cada nota, siguiendo el procedimiento descrito por la Fig. 2.

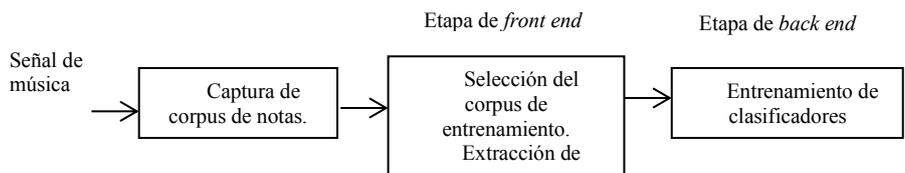


Fig. 1. Diagrama de bloques de la etapa de entrenamiento.

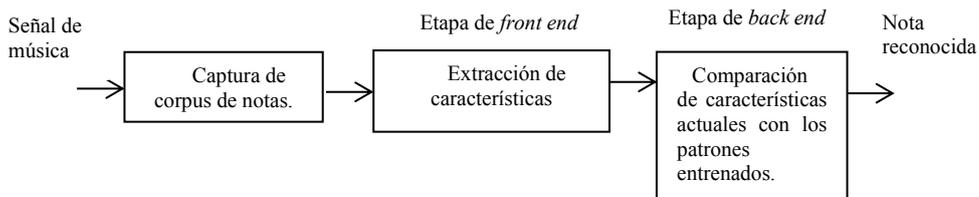


Fig. 2. Diagrama de bloques de la etapa de reconocimiento.

4. Características de la señal de música

Para la implementación del *front end* se utilizaron las siguientes características: Tono fundamental (Pitch), los coeficientes MFCC (cepstrales en la escala de mel) y CMCC (cepstrales de mecánica coclear). A continuación cada una de ellas se describe de forma breve:

4.1. Tono fundamental (*Pitch*)

El tono fundamental (*Pitch* en inglés) es una característica de la señal de música que depende de la frecuencia fundamental (F0) de la misma. Hay diferentes maneras de calcular el F0, para este trabajo se hizo uso de la técnica de autocorrelación parcial.

4.2. Coeficientes cepstrales en la frecuencia de Mel (MFCC)

Los Coeficientes Cepstrales en las Frecuencias de Mel (MFCC en inglés) son otro tipo de características de la señal de música. Trabajan de acuerdo al modelo del sistema auditivo de las personas, es decir, la forma que un ser humano escucha.

El oído humano no percibe las frecuencias de los sonidos linealmente. La manera en que este trabaja demuestra que su escala es aproximadamente lineal hasta 1 kHz y después de ello se comporta de manera logarítmica, y es la Escala de Mel la que caracteriza esa percepción de frecuencia logarítmica por medio de bancos de filtros.

Si se desea obtener más información de la señal de audio además de los MFCC, se pueden calcular más coeficientes tales como los MFCC-Delta (o Δ MFCC) y los MFCC-Delta-Delta (o $\Delta\Delta$ MFCC) [7].

4.3. Coeficientes cepstrales de mecánica coclear (CMCC)

Los coeficientes cepstrales de mecánica coclear (CMCC en inglés) se pueden obtener a partir del funcionamiento de la cóclea [2]. Se utiliza un modelo basado en mecánica de fluidos y su solución se obtiene bajo un análisis de resonancia en el dominio de la frecuencia.

En este tipo de coeficientes la distribución de banco de filtros a los MFCC, es una distribución que depende de la respuesta interna del oído al estímulo que recibe.

5. Clasificadores

Para la parte *back end* se implementaron los siguientes clasificadores: Modelos Ocultos de Markov usando HTK (Hidden Markov Model Toolkit), Modelos de Mixturas Gaussianas, Redes Neuronales y Cuantificación Vectorial. Cada uno de ellos se describe a continuación:

5.1. Modelos ocultos de Markov (HMM)

Un modelo oculto de Markov (HMM en inglés) es un modelo probabilístico aplicado sobre un conjunto de variables aleatorias $(O, Q) = \{O_1, \dots, O_T, Q_1, \dots, Q_T\}$. Las observaciones pueden ser continuas o discretas.

Un HMM resuelve tres problemas básicos:

1. Calcular eficientemente $P(O|\lambda)$, la probabilidad de la secuencia de observaciones $O = \{O_1, O_2, \dots, O_T\}$, dado el modelo $\lambda(A, B, \pi)$.
2. Hallar una secuencia óptima de estados $Q = \{q_1, q_2, \dots, q_T\}$, dada una secuencia de observaciones $O = \{O_1, O_2, \dots, O_T\}$ y un modelo λ .

3. Ajustar los parámetros del modelo $\lambda(A, B, \pi)$ para maximizar $P(O|\lambda)$.

Los tres problemas mencionados anteriormente se aplican para el reconocimiento automático de voz [6]. De manera similar se pueden aplicar en la tarea de reconocer y modelar el problema la transcripción automática de música.

5.2. Modelos de mixturas gaussianas (GMM)

Un modelo de mixtura gaussiana (GMM en inglés) es un modelo probabilístico en el que se hace la suposición de que todos los datos son generados de una mixtura de un número finito de distribuciones gaussianas con parámetros desconocidos, la forma de una mixtura Gaussiana se describe en [5] por:

$$g(\mu, \Sigma)(x) = \frac{1}{\sqrt{2\pi^d} \sqrt{\det(\Sigma)}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}, \quad (1)$$

en dónde las letras μ y Σ representan la media y la covarianza de cada componente respectivamente. Un conjunto de mixturas gaussianas se describe como:

$$gm(x) = \sum_{k=1}^K w_k * g(\mu_k, \Sigma_k)(x). \quad (2)$$

5.3. Redes neuronales (ANN en inglés)

Una red neuronal artificial es un tipo de clasificador perteneciente al área de Inteligencia Artificial. Su modelo puede constar de diferentes elementos dependiendo de su tipo, entre los esenciales se tienen: entradas y salidas, neuronas en sus capas ocultas, pesos y funciones de transferencia.

Las redes tipo feedforward cuentan con capas de neuronas con funciones transferencia no lineales que permiten que la red pueda aprender relaciones lineales y no lineales entre los vectores de entrada y de salida. Si se desea trabajar con valores de salida entre 0 y 1, se debe utilizar una función de transferencia tipo log-sigmoidal. En el presente trabajo se utilizaron las funciones tan-sigmoidal y log-sigmoidal.

5.4. Cuantificación vectorial (VQ)

La cuantificación vectorial (VQ en inglés) es una técnica en la cual un conjunto de vectores de entrada es dividido en un número n de regiones, siendo un vector aquel que define dicha región. El algoritmo que realiza esta tarea se conoce como LBG algoritmo propuesto por [8] este agrupa un conjunto de vectores de entrenamiento L en un conjunto de vectores del libro de códigos M .

6. Experimentos

Para llevar a cabo la evaluación de cada técnica y clasificador, como primer paso fue necesario crear el corpus de las notas a reconocer. Para ello se hizo uso de un piano

digital con un rango de notas comprendido entre C2 y B6, es decir, un total de 60 notas como se muestra en la Fig. 3; cada una de ellas se ejecutó un total de 12 veces a un tempo de 70 bpm para grabarse en un archivo. Posteriormente se recortó cada archivo para separar cada muestra y así tener un total de 720 archivos de muestras.

Cada clasificador se implementó por octavas, y en cada una de estas, hay un contenido total de 12 notas. Debido a que se tienen 12 archivos muestras por cada nota, se tiene un total de 144 archivos muestras por octava.

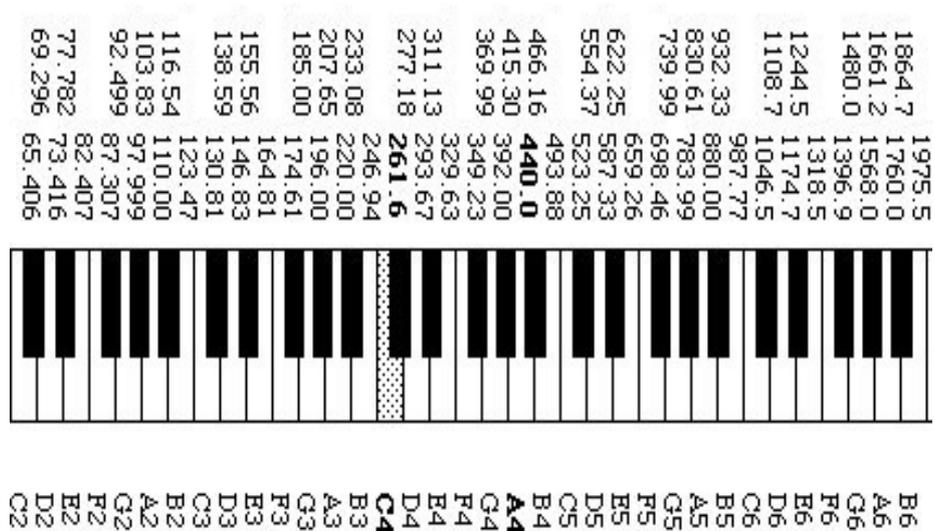


Fig. 3. Rango de notas a reconocer.

6.1. Implementación de HMM con HTK

HTK (Hidden Markov Model Toolkit) es una herramienta utilizada para el reconocimiento de voz. En esta ocasión se ha aplicado para resolver el problema de reconocimiento de notas musicales.

Se crearon modelos para dos tipos de características: MFCC y CMCC. Se emplearon vectores característicos de 39 coeficientes para cada caso. Los primeros 13 coeficientes del vector corresponden a los 12 coeficientes MFCC más el valor de la energía; los segundos 13 son los coeficientes de velocidad (Δ MFCC); y los últimos 13 son los coeficientes de aceleración ($\Delta\Delta$ MFCC). Por lo que cada vector característico v se conformó de la siguiente manera:

$$v = (c_1, c_2, \dots, c_{12}, E, \Delta c_1, \Delta c_2, \dots, \Delta c_{13}, \Delta\Delta c_1, \Delta\Delta c_2, \dots, \Delta\Delta c_{13}). \quad (3)$$

Se hizo un Modelo Oculto de Markov por cada nota y el experimento se realizó para cada una de las cinco octavas. Cada modelo está compuesto por 5 estados y 1 Mixtura Gaussiana por estado. Se dividieron todos los archivos muestras para hacer el entrenamiento con una mitad de ellos y el reconocimiento con la otra. El procedimiento se describe a continuación.

Con la función HParse se crea una red a partir de la gramática propuesta. Esta función recibe un archivo (gramatica.txt) con una descripción sintáctica que sigue un conjunto de reglas, el cual genera la red. El resultado se guarda en un archivo .slf.

Posteriormente se hace el cálculo de coeficientes MFCC y CMCC mediante la función HCopy, la cual copia los archivos de datos a un archivo de salida. Convirtiendo los datos en una forma parametrizada. Esta función originalmente funciona con coeficientes MFCC, por lo cual fue necesario modificarla para calcular coeficientes CMCC.

Con la función HInit es posible realizar las estimaciones iniciales para los parámetros del modelo. Posteriormente la función HRest permite llevar a cabo la reestimación básica Baum-Welch de cada modelo.

Con la función HVite se ejecuta un reconocedor Viterbi de propósito general. Este relaciona un archivo de voz contra una red de modelos HMM y da como salida una transcripción para cada uno. Los resultados se guardan en un archivo .mlf.

Finalmente se aplica la función HResults. Ésta lee un conjunto de archivos de etiqueta y compara con los archivos de transcripción correspondientes. Para el análisis de salida de reconocimiento de voz, la comparación está basada en un procedimiento de alineamiento de cadenas basado en programación dinámica.

6.2. Implementación de GMM

En este clasificador también se utilizó el mismo corpus de datos, una mitad para entrenamiento y la otra para reconocimiento. Para llevar a cabo la implementación de este clasificador se hizo uso de la herramienta VOICEBOX creada por Mike Brookes en MATLAB. En ella están contenidas funciones que fueron empleadas para crear cada modelo de mixtura Gaussiana. El experimento se hizo para 2, 3 y 4 mixturas gaussianas por modelo. Dando como mejores resultados los modelos de 3 mixturas, los cuales se muestran en la Tabla 2.

Se emplearon vectores característicos de 13 coeficientes; compuestos por 12 coeficientes MFCC más el valor del F0, el cual fue calculado por un algoritmo de detección del tono fundamental en cada segmento. Por lo que cada vector característico v se conformó de la siguiente manera:

$$v = (c_1, c_2, c_3, \dots, c_{12}, F0). \quad (4)$$

Este procedimiento se realizó para la mitad destinada al entrenamiento por cada octava de análisis. Una vez entrenados los modelos, se probaron con la mitad restante de los datos. Se hizo un programa capaz de construir una tabla de confusión con la finalidad de una comparación final de resultados.

6.3. Implementación de ANN

Para implementar este clasificador se hizo uso de las funciones mencionadas anteriormente en la implementación de GMM. Se empleó la herramienta Neuronal Network Toolbox de MATLAB. En ella están contenidas funciones que fueron empleadas para crear cada modelo de redes neuronales. El experimento se hizo para 20, 30 y 40 neuronas en la capa oculta. Dando como mejores resultados los modelos de 20 neuronas, los cuales se muestran en la Tabla 2.

6.4. Implementación de VQ

Para implementar este clasificador se hizo uso de las funciones mencionadas anteriormente para GMM. Se creó un libro código por cada nota para cada octava. El experimento se realizó con 16 y 32 vectores de libro código. Dando como mejores resultados el libro código de 32 vectores, los cuales se muestran en las Tabla 2.

Tabla 2. Resultados por modelo utilizado.

| Octava 2 | | | | | | | |
|--------------|------|------------|----|---|----|---|----|
| Clasificador | | %Corr | H | D | S | I | N |
| HMM (MFCC) | [39] | 77.78 | 56 | 0 | 16 | 0 | 72 |
| HMM (CMCC) | [39] | 100 | 72 | 0 | 0 | 0 | 72 |
| GMM | [13] | 100 | 72 | 0 | 0 | 0 | 72 |
| ANN | [13] | 98.61 | 71 | 0 | 1 | 0 | 72 |
| VQ | [13] | 100 | 72 | 0 | 0 | 0 | 72 |
| Octava 3 | | | | | | | |
| Clasificador | | %Corr | H | D | S | I | N |
| HMM (MFCC) | [39] | 90.28 | 65 | 0 | 7 | 0 | 72 |
| HMM (CMCC) | [39] | 100 | 72 | 0 | 0 | 0 | 72 |
| GMM | [13] | 100 | 72 | 0 | 0 | 0 | 72 |
| ANN | [13] | 91.66 | 66 | 0 | 6 | 0 | 72 |
| VQ | [13] | 100 | 72 | 0 | 0 | 0 | 72 |
| Octava 4 | | | | | | | |
| Clasificador | | %Corr | H | D | S | I | N |
| HMM (MFCC) | [39] | 98.61 | 71 | 0 | 1 | 0 | 72 |
| HMM (CMCC) | [39] | 100 | 72 | 0 | 0 | 0 | 72 |
| GMM | [13] | 97.22 | 70 | 0 | 2 | 0 | 72 |
| ANN | [13] | 98.61 | 71 | 0 | 1 | 0 | 72 |
| VQ | [13] | 100 | 72 | 0 | 0 | 0 | 72 |
| Octava 5 | | | | | | | |
| Clasificador | | %Corr | H | D | S | I | N |
| HMM (MFCC) | [39] | 98.61 | 71 | 0 | 1 | 0 | 72 |
| HMM (CMCC) | [39] | 100 | 72 | 0 | 0 | 0 | 72 |
| GMM | [13] | 100 | 72 | 0 | 0 | 0 | 72 |
| ANN | [13] | 98.61 | 71 | 0 | 1 | 0 | 72 |
| VQ | [13] | 100 | 72 | 0 | 0 | 0 | 72 |
| Octava 6 | | | | | | | |
| Clasificador | | %Corr | H | D | S | I | N |
| HMM (MFCC) | [39] | 97.22 | 70 | 0 | 2 | 0 | 72 |
| HMM (CMCC) | [39] | 100 | 72 | 0 | 0 | 0 | 72 |
| GMM | [13] | 75 | 54 | 0 | 18 | 0 | 72 |
| ANN | [13] | 83.33 | 60 | 0 | 12 | 0 | 72 |
| VQ | [13] | 100 | 72 | 0 | 0 | 0 | 72 |

7. Resultados

En la Tabla 2 se muestran los resultados obtenidos por cada clasificador. Como se mencionó anteriormente, se tiene un total de 144 archivos muestras por octava. Dado que la mitad de ellos fue utilizada para el entrenamiento y la otra para el reconocimiento, éste último se aplica sobre 72 archivos, N . El indicador $\%Corr$ es el porcentaje de notas reconocidas correctamente. H es el número de notas correctas en la octava de análisis. D es el número de eliminaciones. S es el número de sustituciones, cuando una nota se reconoce como otra distinta existe una sustitución. Y finalmente I es el número de inserciones. Los valores entre corchetes a lado del nombre de cada clasificador, indican el número de coeficientes utilizado en cada vector de entrenamiento.

8. Conclusiones y trabajos futuros

Después de haber realizado los experimentos anteriores, se ha demostrado que los clasificadores con mejores resultados para realizar una transcripción automática de música han sido los HMM con CMCC y la Cuantificación Vectorial con MFCC como características. Sin embargo, es importante mencionar que cuando se han aplicado los HMM estos no contemplan como necesario el cálculo del F_0 , lo cual representa una gran ventaja ya que se ahorra la implementación y ejecución del algoritmo para su detección.

Cabe decir que a pesar de no ser un clasificador tan complejo como el resto, el cuantificador vectorial ha alcanzado excelentes resultados en este trabajo. Esto probablemente debe a que la tarea de reconocimiento se llevó a cabo únicamente para notas monofónicas y aisladas con una misma duración.

Dados los resultados mostrados, se puede considerar llevar a cabo la implementación de los HMM con CMCC y la Cuantificación Vectorial con MFCC como características sobre una plataforma, para realizar la transcripción automática de música en tiempo real.

Referencias

1. Han-Jong, K., Tek-Jin, N.: Muzlog: Instant Music Transcribing System for Acoustic Guitarists. CIDR Lab, Department of Industrial Design (2014)
2. Oropeza, J., L.: New Parameters for Automatic Speech Recognition Based on the Mammalian Cochlea Model Using Resonance Analysis (2013)
3. Benetos, E.: Automatic Transcription of Polyphonic Music Exploiting Temporal Evolution. PhD thesis, School of Electronic Engineering and Computer Science Queen Mary University of London (2012)
4. Farshad, A., Doraisamy, S.: A Real-Time Note Transcription Technique Using Static and Dynamic Window Sizes. In: ICCSE, Malaysia (2009)
5. Bilmes, J. A.: A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. International Computer Science Institute, Berkeley, CA. (1998)

Omar Velázquez López, José Luis Oropeza Rodríguez, Sergio Suárez Guerra

6. Lawrence, Rabiner, L., Biing-Hwang J.: Fundamentals of Speech Recognition. Prentice Hall (1993)
7. Furui, S.: Comparison of speaker recognition methods using statistical features and dynamic feature (1981)
8. Linde, Y., Buzo, A., Gray, R., M.: An algorithm for Vector Quantizer Design (1980)

Algoritmo para el cálculo del ritmo en una señal de audio digital

Luis Felipe Romero Morales, José Martín Flores Albino

Universidad Autónoma del Estado de México,
Centro Universitario UAEM Valle de México, Estado de México, México

lromero628@alumno.uaemex.mx, jmfloresa@uaemex.mx

Resumen. En este trabajo se presenta un algoritmo para la extracción del ritmo de una señal audio digital. El ritmo está formado por los cambios de intensidad en la señal de audio. El ritmo marca cambios significativos en las señales acústicas. A través de un procedimiento de filtrado para la extracción de la envolvente de la señal de audio, luego a través de una señal cuadrada se marcan los puntos de inicio y su duración de los cambios de ritmo en la señal de audio. El algoritmo se prueba con una señal de un metrónomo de 60pps para validar esta propuesta.

Palabras clave: Extracción del ritmo, señal envolvente, filtro digital.

Algorithm for Rhythm Evaluation in a Digital Audio Signal

Abstract. In this paper presents an algorithm for extraction of audio rhythm of a digital signal. The rhythm is formed by changes of intensity in the audio data signal. The rhythm marks significant changes in the acoustic signals. Through a procedure of filtering for extraction the envelope of the audio signal, then through a square signal marked the start point and duration of changes on the envelope. The algorithm is tested with a signal of a metronome of 60bps to validate this proposal.

Keywords: Rhythm extraction, signal envelope, digital filtering.

1. Introducción

Este artículo presenta un algoritmo para el cálculo del ritmo de una señal digital de audio. El ritmo se expresa en la envolvente de la señal de audio y marca los cambios significativos, sin importar el tono o frecuencia de la misma. El ritmo se experimenta al producirse cambios en el sonido, que ocurren de forma secuencial y organizada durante el paso del tiempo [1]. El algoritmo presentado produce como resultado datos que indican el tiempo y la duración de los cambios asociados al ritmo de la señal.

La detección del ritmo tiene diversas aplicaciones; por ejemplo: En el ámbito médico existe una relación entre el ritmo del corazón y la detección de algunas enfermedades; en los ritmos cerebrales de señales asociadas a la actividad del cerebro, indican su funcionamiento y estado [2]; en la música, el ritmo puede usarse para sincronizar efectos como son de iluminación y movimiento que harían más vívida la experiencia musical. Entonces, se entiende que el ritmo puede ser asociado a diferentes áreas de interés.

Hay trabajos que presentan aplicaciones y técnicas para la extracción del ritmo. En [3], los autores proponen un orden jerárquico de géneros musicales por medio de una clasificación automática de audio; son tres los principios que guían su investigación: la textura del timbre, contenido rítmico y el tono. En [4], se propone un método para representar las periodicidades creadas por un ritmo musical, usan las crestas del sonido para calcular un ritmo de acompañamiento y hacen uso de la transformada de wavelet inversa. En [5], se utiliza la transformada de wavelet y de Fourier para buscar la estructura rítmica y melódica de la música, utilizan un método de escalogramas para percusiones como técnica para analizar ritmos.

El orden de este artículo es el siguiente. En la sección 2, se presenta el algoritmo propuesto y los procedimientos en cada fase. Se inicia con la adquisición de la señal de audio y culmina con la generación de un archivo donde se enlista el tiempo en el que ocurre un cambio significativo en el ritmo y su duración. En la sección 3 se presenta la etapa de pruebas por medio de un ejemplo y los resultados de aplicación del algoritmo. Se utiliza una señal de un metrónomo de 60 pulsos/min. Al aplicar el algoritmo se generan datos que indican los eventos asociados al ritmo de la señal. En la última sección se dan las conclusiones y trabajo futuro de esta investigación.

2. Algoritmo para la detección del ritmo

Para el desarrollo del algoritmo se utiliza el ambiente Matlab™ por ser una herramienta potente de cálculo para actividades de investigación. En la fig. 1, se presenta un diagrama de flujo de los procesos que se ejecutan en el algoritmo.

Proceso 1: El objetivo es cargar los datos de la señal de audio digital en formato mp3. La Frecuencia de Muestreo usada es de $fs=44\ 100$ muestras/segundo [6]. La amplitud está normalizada en el rango de -1 a 1 para los canales izquierdo y derecho. Así entonces un segundo de audio tendrá 88 200 valores de punto flotante en formato de 16 bits.

Proceso 2: La señal obtenida se encuentra dividida en los canales izquierdo y derecho, por ser una señal en “estéreo”. El cambio de ritmo es muy similar en ambos canales, por lo que se procesa una señal monoaural obtenida al promediar los dos canales. Lo anterior además reducirá el número de datos a la mitad.

Proceso 3: La información del ritmo se encuentra en la envolvente de la parte positiva y negativa de la señal monoaural. La envolvente es la representación el ritmo debido a que marca los cambios en la señal sin importar el tono o frecuencia. De manera que se decide eliminar la parte negativa de la señal debido a que, para propósitos prácticos, es la misma que la envolvente de los valores positivos. Para lograrlo hay dos formas: la primera: tomar el valor absoluto; y la segunda: elevar al cuadrado y después

obtener la raíz cuadrada. Se elige el primer método por ser menos demandante de recursos de cómputo.

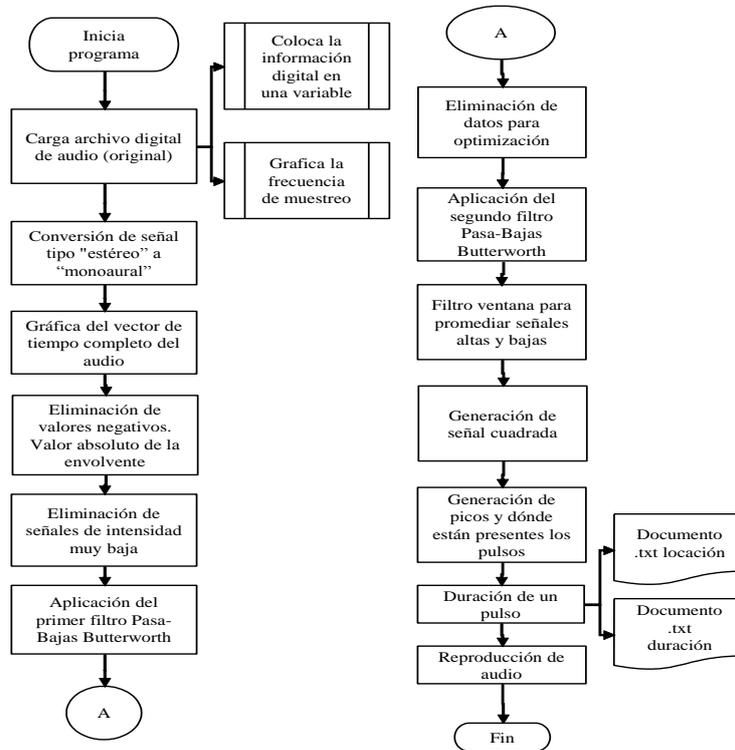


Fig. 1. Diagrama de flujo general del algoritmo para el cálculo del ritmo de una señal digital.

Proceso 4: Dentro de los archivos de audio, en ocasiones se presenta ruido de intensidad muy baja, esto puede afectar el funcionamiento del algoritmo, al detectarlos como parte del audio. Para solucionarlo se utiliza un umbral de nivel bajo para acotar lo que es ruido en la señal.

Proceso 5: La envolvente representa cambios de frecuencia muy baja. La frecuencia de sonido baja es de 20Hz y de 20kHz la frecuencia alta, por lo que se aplica un filtro de elimine los cambios de las señales de audio, y con esto obtener la envolvente. Se aplica un filtro digital tipo Butterworth pasa-bajos con frecuencia de corte de 10 Hz para este propósito.

La función de transferencia un filtro pasa-bajos es:

$$|H(\omega)|^2 = \frac{1}{1+(\omega/\omega_c)^{2N}}, \quad (1)$$

donde ω es la frecuencia angular de la señal, $\omega_c = 2\pi f_c$, f_c es la frecuencia de corte y N orden del filtro. Para llevarlo a su forma digital se utiliza la transformación bilineal.

$$s = \frac{1}{T_s} \frac{z-1}{z+1}, \quad (2)$$

donde T_s es el periodo de muestreo, z es la variable de la transformada z (z^{-1} es un retardo).

Proceso 6: La señal de la envolvente tiene una frecuencia de muestreo de 44 100 muestras por segundo. Se han eliminado las frecuencias por arriba de los 20Hz. Recordando el teorema de Nyquist-Shannon, que indica que para recuperar una señal debe ser muestreada al menos a una tasa de muestreo del doble del ancho de banda de la señal. Se concluye que la señal de la envolvente está en exceso muestreada. El ancho de banda de la señal de envolvente es de a lo más 10 Hz, así que la señal de envolvente se debería submuestrear a 1/2205 veces. Para hacerlo se toma un valor por cada 2205 de la envolvente del audio. Esto además reduce la cantidad de información en la señal.

Proceso 7: Con una menor cantidad de datos se aplica una vez más un filtro pasa-bajas con el propósito de suavizar la envolvente.

Proceso 8: El ritmo está expresado en la forma de la envolvente. Un efecto no deseado de aplicar el filtro pasa-bajas es que se acumula un valor residual que va elevando el nivel promedio de la señal de la envolvente. Para eliminar este aumento de nivel se aplica un filtro del tipo media móvil (*moving average*) para el cálculo del valor medio y posteriormente eliminarlo a través de una simple resta.

Proceso 9: En esta etapa se ha obtenido la señal del ritmo. Pero con el propósito de cuantificar los cambios en la señal del ritmo. Se marca con una señal cuadrada en el momento donde hay un cambio en la envolvente y su duración. Para ello se calculan los máximos y mínimos en la envolvente. Usando los puntos de máximo para indicar el tiempo en el que inicia un cambio en el ritmo y los puntos mínimos marcan cuando termina el cambio detectado, es decir su duración. Cuando la señal cuadrada pasa a nivel alto indica el inicio de cambio en la envolvente. La duración de los cambios serán el tiempo que tarda en alcanzarse los mínimos de la señal de la envolvente. Así se obtendrán los cuándoos y cuánto duran los cambios en el ritmo.

Proceso 10: Para medir los tiempos se cuenta el número de muestras de la señal cuadrada y tiempo el muestreo de la señal. Estos resultados se registran en dos archivos de texto plano (txt). El primero contiene las marcas de tiempo que corresponde al inicio de un cambio en el ritmo y el segundo la duración del cambio en la señal del audio digital inicial.

3. Pruebas y resultados

Se presentan los resultados al aplicar el algoritmo anterior para una señal de audio. La señal seleccionada es la señal de un metrónomo a 60 bpm (bits por minuto), el archivo es de formato mp3 y una frecuencia de muestreo de 44.1k muestras/segundo, la magnitud de los valores tiene un rango entre -1 y 1, con una codificación digital de 16 bits de resolución. La razón por se toma esta señal es que resulta claro el resultado esperado y permitiendo evaluar si el algoritmo entrega resultados coherentes.

En la fig. 2, se presenta la señal sin procesar. En la figura 2.a es el detalle de un solo pulso del metrónomo, se observa su complejidad. En la figura 2.b se presentan 10 pulsos del metrónomo.

En la fig. 3, se presenta la señal después de convertirla en monoaural y tomar su valor absoluto para dejar solo la parte positiva. Esto prepara la señal para el cálculo del ritmo.

En la fig. 4, se muestra el resultado de filtrar las frecuencias altas (tono de audio) de la señal anterior. El filtro usado es de tipo IIR (Respuesta al Impulso Infinita) digital pasa-bajos tipo Butterworth. Para el primer filtro se utilizaron los siguientes parámetros: frecuencia de corte: $f_c=10$ Hz y $N=4$ es el orden del filtro. El resultado es una señal que representa la envolvente de la señal. La señal resultante tiene una cantidad de 44.1K datos por segundo que impactaría en el tiempo de procesamiento, por eso se realiza un submuestreo de la señal, al tomar 1 de cada 2205 datos, reduciendo la cantidad de datos a procesar.

Observar que la envolvente muestra los pulsos de la señal y notar que se va incrementando su nivel al paso del tiempo, esto produce un “*efecto de escalera*”, que se debe a la acumulación del valor promedio de la señal. Para eliminarlo el efecto escalera se calcula el valor medio de la señal de la envolvente a través de un filtro de valor medio, ver fig. 5.

Para eliminar efecto escalera, se cancela el valor medio por simple resta de la envolvente. La señal resultante muestra los pulsos del metrónomo continuo a través del contorno, en la fig. 6, se observan los pulsos del sonido del metrónomo con claridad.

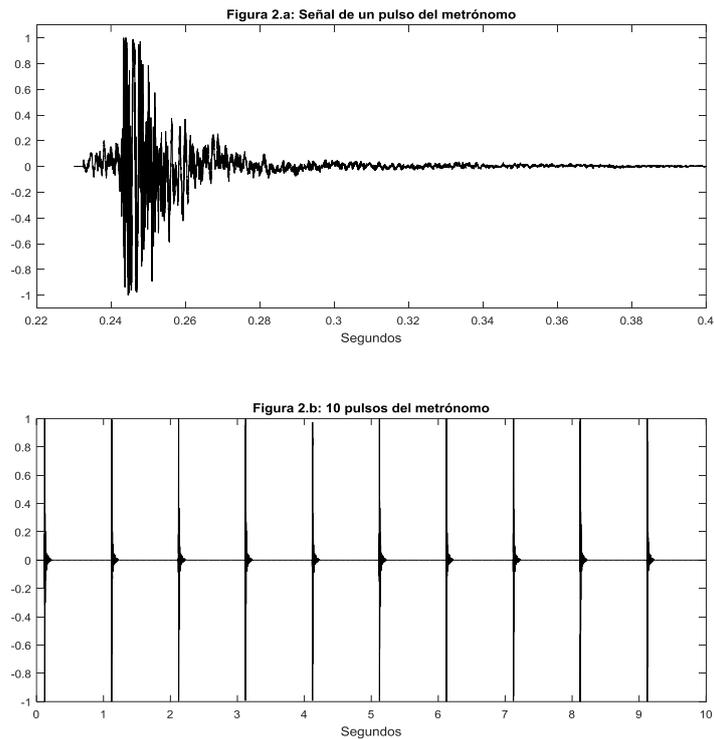


Fig. 2. Señal de audio de sonido de un metrónomo a 60bps. Señal sin procesar: 2.a) es la señal de un solo pulso. 2.b) Trazo de diez pulsos.

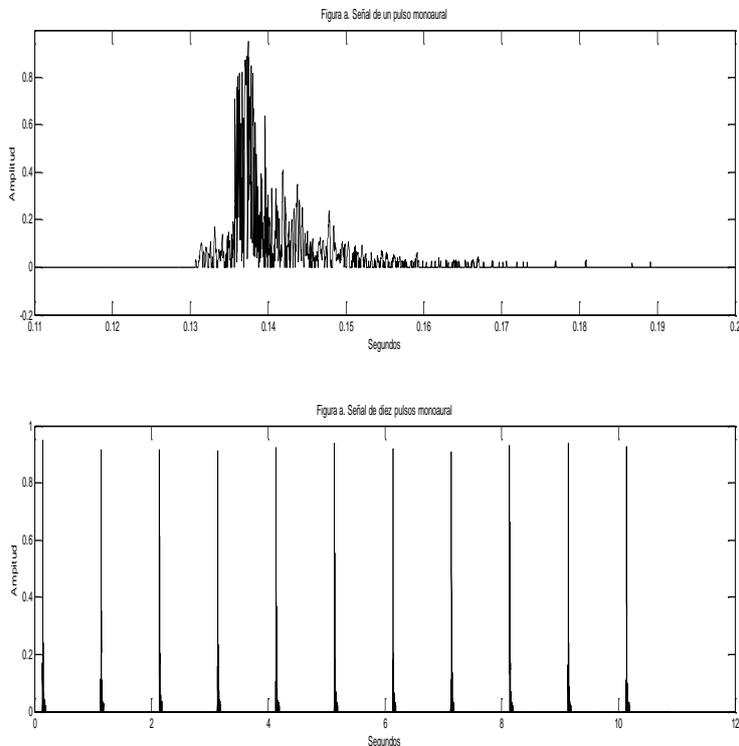


Fig. 3. Señal de audio de sonido de un metrónomo a 60bps. Valor absoluto de la señal, recortada su parte negativa: 3.a) Un pulso. 3.b). Trazo de diez pulsos.

El siguiente paso es generar una señal que sea capaz de indicar el principio y el fin de un pulso rítmico dentro de la señal. Esto es posible con la generación de una señal cuadrada. En el resultado que se observa en la **fig. 7.**, donde se forman los pulsos que indican el tiempo donde el pulso se presenta y su duración. Este paso es muy importante porque la generación de la señal cuadrada concede información del momento en que hay un cambio significativo en el audio. Para ello se evalúan los puntos de máximos y mínimos de la señal de la envolvente. En los máximos se indican cuando hay un cambio en la señal de audio y el punto mínimo cuando la señal cuadrada regresa al nivel de referencia. La duración de un cambio en la señal de la envolvente es equivalente al tiempo que la señal cuadrada está en nivel alto.

Resultados: La señal de audio para la prueba del algoritmo es de un metrónomo a 60 bpm, con duración de 10 minutos y 11 segundos, dando un total de 606.6 pulsos. El algoritmo arroja una cantidad total de 606 pulsos con una duración de 0.4 segundos. Los archivos generados contienen la información digital del tiempo de cuándo hay un cambio de la señal y su duración. En la tabla 1 se muestran los resultados. Se observa que el primer pulso inicia a los 0.1 segundos, desde ese tiempo en adelante, cada segundo se produce otro pulso y la duración de los pulsos es de 0.4 segundos. Estos resultados están de acuerdo a lo esperado.

Algoritmo para el cálculo del ritmo en una señal de audio digital

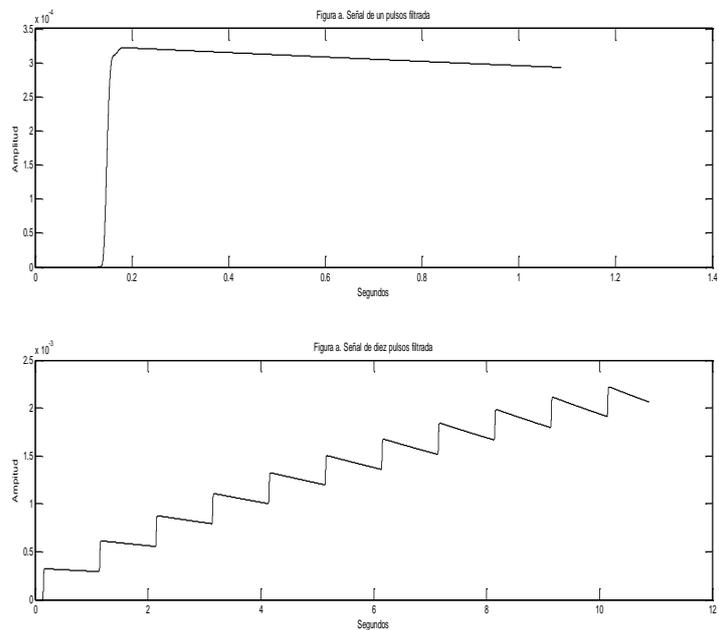


Fig. 4. Aplicación del primero y segundo filtro. 4.a Trazo de un pulso. 4.b Trazo de diez pulsos (*efecto escalera*).

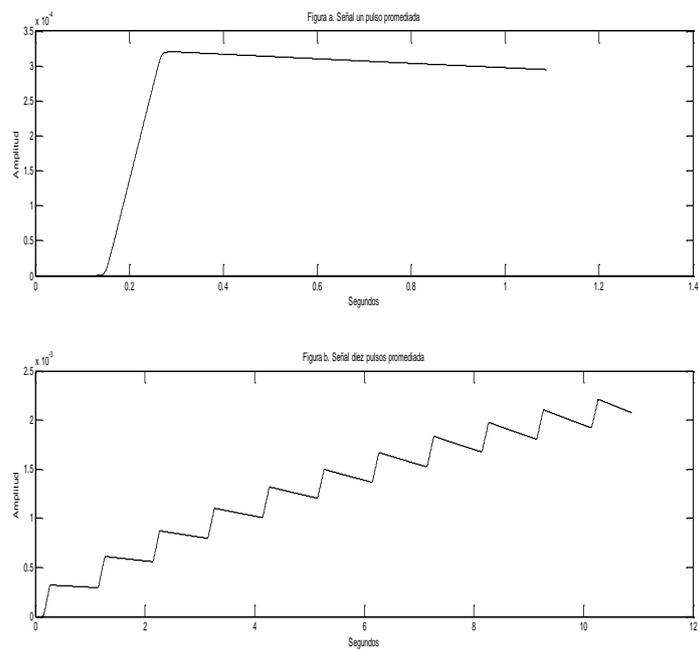


Fig. 5. Valor promedio de la señal: Se obtiene al calcular el valor medio de la señal. 5.a Trazo de un pulso. 5.b Trazo de diez pulsos (*efecto escalera*).

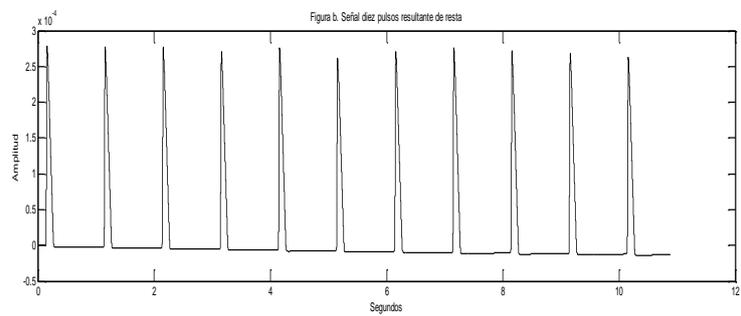
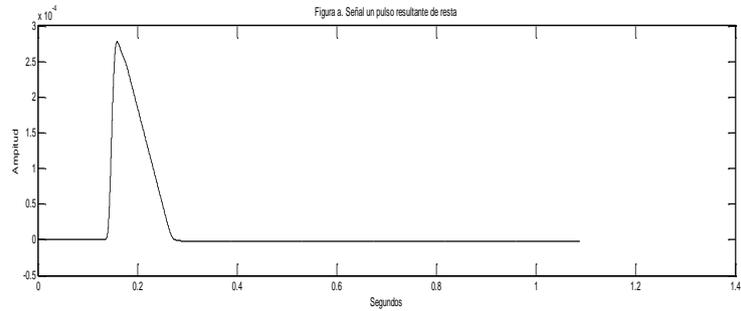


Fig. 6. Señales de ritmo del metrónomo: Resultante de restar el valor promedio de la señal y la filtrada. 6.a Trazo de un pulso. 6.b Trazo de diez pulsos.

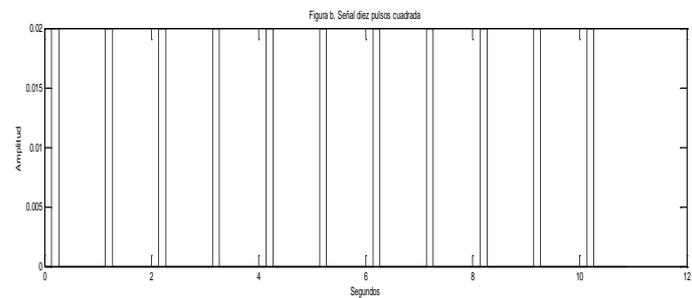
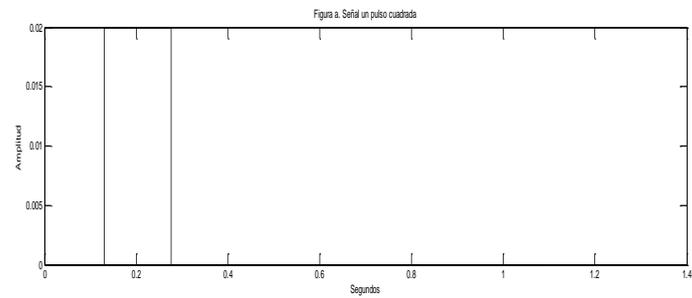


Fig. 7. Señal Cuadrada: Los pulsos cuadrados indican la detección hecha por el algoritmo en la señal. 7.a Trazo de pulso. 7.b. Trazo de diez pulsos.

Tabla 1. Valores en tiempo de la señal de audio procesada (únicamente los primeros 10 pulsos).

| Inicio de pulso (x100 milisegundos) | Duración (x50 milisegundos) |
|-------------------------------------|-----------------------------|
| 1=0.1 segundos | 4=0.4 segundos |
| 11=1.1 segundos | 4=0.4 segundos |
| 21=2.1 segundos | 4=0.4 segundos |
| 31=3.1 segundos | 4=0.4 segundos |
| 41=4.1 segundos | 4=0.4 segundos |
| 51=5.1 segundos | 4=0.4 segundos |
| 61=6.1 segundos | 4=0.4 segundos |
| 71=7.1 segundos | 4=0.4 segundos |
| 81=8.1 segundos | 4=0.4 segundos |
| 91=9.1 segundos | 4=0.4 segundos |

4. Conclusiones y trabajo futuro

Se concluye que los pulsos contenidos en el audio de prueba y su duración corresponden a la señal usada, pues se están presentando en la posición y con la duración correcta de todos los pulsos presentes. Esta señal sirvió para una validación inicial del algoritmo ya que los pulsos se identificaron de manera precisa; sin embargo, es necesario realizar las pruebas para evaluar la capacidad del algoritmo y los ajustes que deba de tener según la aplicación. Al probarse con una señal de audio digital compleja se observaron resultados que mostraban sincronía con los datos obtenidos. Como trabajo futuro se debe probar exhaustivamente el algoritmo con diferentes señales de audio y realizar un estudio comparativo con otras metodologías existentes para la extracción del ritmo, además de usar el algoritmo en aplicaciones donde esta técnica resulte útil.

Referencias

1. Encyclopædia Britannica. <https://global.britannica.com/art/rhythm-music> (2016)
2. Andrea, B.: Técnicas de procesamiento de EEG para detección de eventos. Argentina: Universidad Nacional de la Plata (2013)
3. George T., Perry C.: Musical genre classification of audio signals. IEEE Transactions on Speech and Audio Processing, Vols. 1 de 210, Issue 5, ISSN: 1063–6676, pp. 293–302 (2002)
4. Leigh, S., Henkjan, H.: Frequency Representation of musical rhythm by continuous wavelets. Journal of Mathematics And Music, vol. 2, Issue 2, pp. 81–97 (2008)
5. Cheng, X., Jarod, H., James, W.: Time-frequency Analysis of Musical Rhythm. Notices of the AMS, vol. 56, n. 3, pp. 256–372 (2009)
6. Jesualdo, F.: Tema 3: Audio, Tecnologías para los Sistemas Multimedia. dis.um.es/~jfernand/0405/tsm/tema3.pdf, 1–65 (2004)
7. Kamenov, A.: Digital Signal Processing for Audio Applications. Create Space Independent Publishing (2013)
8. Joshua, R., McPherson, A.: Audio Effects: Theory, Implementation and Application. CRC Press Taylor & Francis Group (2015)

Luis Felipe Romero Morales, José Martín Flores Albino

9. Aggelos, G., Vassilis, K., George, C.: Towards Multi-Purpose Spectral Rhythm Features: An Application to Dance Style, Meter and Tempo Estimaion. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 24, Issue 11, pp. 1885–1896 (2016)
10. Shiratori, T., Nakazawa, A., Ikeuchi, K.: Detecting dance motion structure through music analysis. In: *Automatic Face and Gesture Recognition*, DOI: 10.1109/AFGR.2004.130164 (2004)

Optimización en ubicación de estaciones base de baja potencia LTE

Yair Ali Cerecedo Torres, Miguel Sánchez Meraz, Antonio Valdovinos Bardají,
Ángela Hernández Solana

Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica,
Ciudad de México, México

Universidad de Zaragoza, Escuela de Ingeniería y Arquitectura , Zaragoza,
España

ycerecedot0900@alumno.ipn.mx, mmeraz@ipn.mx, toni@unizar.es, anhersol@unizar.es

Resumen. En este trabajo se desarrolla un Modelo de Programación Lineal Entera (MPLE) para encontrar el emplazamiento óptimo de Estaciones Base de Baja Potencia (EBBP) con el objetivo de solucionar problemas de congestión en una macro celda LTE. Una función de costos representa la función objetivo del modelo de optimización y está restringida a los parámetros técnicos de diseño más importantes en el proceso de planeación de redes: cobertura, demanda de usuario y capacidad. El MPLE es aplicado y evaluado en dos escenarios de análisis. Estos escenarios se configuran en un software de planificación de redes tanto para alimentar al modelo como para determinar si los requerimientos de los escenarios son alcanzados.

Palabras clave: Modelo de programación lineal entera, despliegue de estaciones base de baja potencia, planeación de redes.

Optimization in Placement of Low Power Base Stations LTE

Abstract. This paper develops an Integer Linear Programming Model (ILPM) to find the optimal emplacement of low power base stations in order to resolve traffic congestion problem in hotspots within a LTE macro cell. A cost function represents the objective function inside the ILPM and it is restricted to the most important technical parameters in the planning networks process of design: user demand, coverage and capacity. The ILPM is implemented and evaluated in two analysis scenarios. These scenarios are set up in planning network software to both feed the input of model and determine if the requirements of the scenario are met.

Keywords. Integer linear programming model, deployment of low power base stations, small cell, planning network.

1. Introducción

Desde el punto de vista de capas, una manera de expandir y aumentar las capacidades de una red de acceso LTE mientras se mantiene como una red homogénea, es agregar más sectores o desplegando más Macro Estaciones Base (MEB) en el área de interés. Sin embargo, estas tareas sólo pueden llevarse a cabo en cierta medida debido a que el espectro es limitado y la búsqueda de nuevos emplazamientos es una tarea difícil sobre todo en entornos urbanos [1].

En las especificaciones de LTE avanzado (R10) se propone introducir Estaciones Base de Baja Potencia (EBBP) dentro del área de servicio de macro estaciones base para aumentar de manera más eficiente las capacidades de las redes de acceso LTE [3, 4]. Esto permite aumentar la capacidad de la red en zonas específicas donde se presentan grandes concentraciones de usuarios y/o cubrir áreas que no son cubiertas por las macro estaciones base debido a condiciones de propagación [2].

Un factor muy importante que determina un correcto despliegue es la elección de la ubicación de los emplazamientos donde se instalarán las estaciones base de baja potencia. En los últimos años se han realizado numerosos trabajos que se han apoyado con la investigación de operaciones para determinar el emplazamiento óptimo de EBBP. Dicha área ofrece la posibilidad de desarrollar modelos matemáticos de optimización que son útiles para determinar la ubicación óptima de estaciones base, minimizando o maximizando diferentes parámetros técnicos y económicos [3].

El trabajo se organiza de la siguiente manera: en el apartado 2 se describe el modelo de optimización desarrollado, en el apartado 3 se presentan los resultados de la aplicación del modelo de optimización en dos escenarios de análisis (EA) y, por último, el apartado 4 contiene las conclusiones generales de este trabajo [4, 5].

2. Problema de optimización para la ubicación de estaciones base de baja potencia

El desarrollo del modelo de optimización parte de un escenario de análisis general como se muestra en la Figura 1. Este escenario general consiste en una macro celda dividida en $l \in L$ sectores, que eventualmente presenta un número determinado de zonas de alta densidad de usuarios (ZAD) $h_{i,n} \in H$. En dichas zonas, se disponen de $p_n^{h \in H} \in P$ posibles emplazamientos (PE) para instalar EBBP y se disponen de $b \in B$ anchos de canal para ser asignados dentro de un espectro limitado. Estos canales se identifican con el conjunto $B = \{b_0, b_1, b_2, \dots, b_7\}$ que corresponden a los anchos de canal disponibles en LTE [6]. El primer elemento de este conjunto b_0 está asociado a un canal de 0 MHz y debe ser interpretado como *sin canal asignado*. Para realizar un despliegue ordenado de las EBBP, se proponen diferentes escenarios de despliegue (ED) identificados con el conjunto $F = \{f_1, f_2, f_3, \dots, f_n\}$. Los ED $f \in F$ son evaluados en su desempeño a través de simulaciones en una herramienta computacional de

planeación para determinar el área cubierta del área total A_T de las ZAD $h \in H$ que se encuentren en el sector $l \in L$. El número de escenarios de despliegue $f \in F$ en cada $h \in H$ es determinado por $2^p - 1$. Cada ED $f \in F$ tiene un costo de instalación C_f , de la misma manera, se tiene un costo asociado C_b por la asignación del ancho de banda $b \in B$ tanto para las EBBP como para las sectorizaciones $l \in L$ de la macro celda.

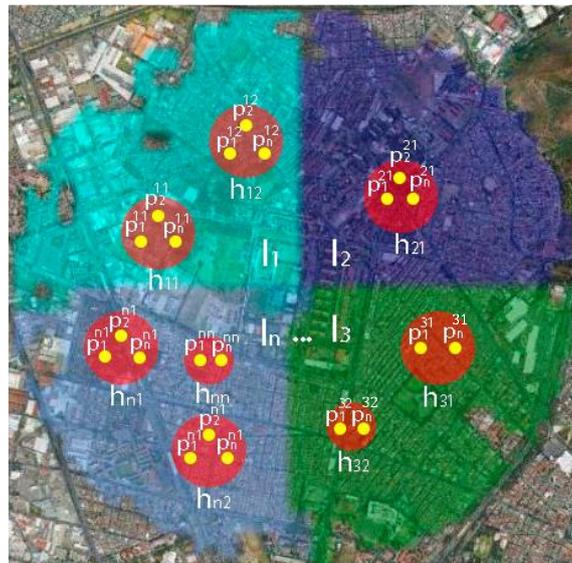


Fig. 1. Escenario general de análisis para el desarrollo del modelo de optimización lineal entero.

El problema de optimización además de determinar la posición de EBBP también consiste en realizar una gestión óptima de los recursos espectrales disponibles, es decir, asignar el menor ancho de banda posible tanto a los sectores $l \in L$ de la MEB como a cada una de las EBBP desplegadas, lo que conlleva a la minimización de los costos del despliegue.

2.1. Variables de decisión

Se introducen tres variables booleanas de decisión para el problema de optimización:

$$x_{h,f}^l = \begin{cases} 1 & \text{Se instala } f \in F \text{ en } h \in H \text{ ubicada en } l \in L \text{ de la macro celda.} \\ 0 & \text{Otro caso} \end{cases} \quad (1)$$

$$y_{p,b}^{l,h} = \begin{cases} 1 & \text{Se asigna } b \in B \text{ a } p_n^{h \in H} \in P \text{ en } h \in H \text{ ubicada en } l \in L \text{ de la macro celda.} \\ 0 & \text{Otro caso} \end{cases} \quad (2)$$

$$z_b^l = \begin{cases} 1 & \text{Se instala el canal } b \in B \text{ al sector } l \in L \text{ de la macro celda.} \\ 0 & \text{Otro caso} \end{cases} \quad (3)$$

2.2. Función objetivo

La función objetivo que se ha planteado para el problema de encontrar la ubicación óptima de las EBBP, es una función de costos que considera los siguientes aspectos: costo de instalación del despliegue de EBBP, costo de la asignación de canal a las EBBP, costo de la asignación de canal a los sectores de la MEB. La función objetivo se muestra en la Ecuación 4:

$$\min \gamma = \sum_{l \in L} \sum_{h \in H} \sum_{f \in F} C_f * x_{h,f}^l + \sum_{l \in L} \sum_{h \in H} \sum_{p \in P} \sum_{b \in B} C_b * y_{p,b}^{l,h} + \sum_{l \in L} \sum_{b \in B} C_b * Z_b^l. \quad (4)$$

2.3. Restricciones

Requerimiento de cobertura. Para asegurar que el $f \in F$ atienda al menos a un porcentaje determinado del área (RC) de la ZAD $h \in H$ ubicada en el sector $l \in L$, se debe cumplir la Relación 5, Donde $S_{f,p}^{l,h}$ es la estimación de cobertura de cada ED:

$$R_1 : \sum_{f \in F} \sum_{p \in P} x_{h,f}^l * S_{f,p}^{l,h} \geq RC. \quad (5)$$

Asignación de canal a las EBBP desplegadas. La asignación del ancho de banda del canal está en función de la demanda de servicio que se tenga presente en la ZAD $h \in H$ ubicada en el sector $l \in L$. Esta demanda está asociada a la densidad de usuarios de la zona U_h^l y su perfil de tráfico Td_h^l . La demanda que existe también depende del área $S_{f,p}^{l,h}$ que esté atendiendo la EBBP en el posible emplazamiento $p_n^{h \in H} \in P$ del ED $f \in F$ del área total de la ZAD.

Por otra parte, la capacidad que se alcanza en la ZAD $h \in H$ depende del número de bloques de recursos W_b disponibles en el ancho de canal $b \in B$ asignado y del tipo de modulación del enlace $Msc_{h,f}^l$. Para asegurar que la demanda sea menor a la capacidad se debe cumplir la Relación 6:

$$R_2 : x_{h,f}^l * U_h^l * Td_h^l * \left(\frac{S_{f,p}^{l,h} * A_h^l}{100} \right) \leq \sum_{b \in B} \frac{120 * y_{p,b}^{l,h} * W_b * Msc_{h,f}^l}{10^{-3}}. \quad (6)$$

Asignación del canal a los sectores de la MEB. De la misma manera que el planteamiento de la asignación de canal a las EBBP, la asignación del canal a los sectores de la MEB depende de la demanda y de la capacidad de acuerdo, en general, a los usuarios atendidos dentro de la macro celda y al canal asignado a los sectores respectivamente. Sin embargo, se deben tomar en cuenta los usuarios que no sean atendidos por las EBBP. Para asegurar que la demanda sea menor a la capacidad en la MEB se debe cumplir la Relación 7:

$$R_3 : Um_l * Tdm_l * \left(\frac{Am_l * A_T}{100} - \sum_{h \in H} A_h^l \right) + U_h^l * Td_h^l * X_{h,f}^l * \left(A_h^l - \sum_{p_n^{h \in H} \in P} \frac{S_{f,p}^{l,h} * A_h^l}{100} \right) \dots \quad (7)$$

$$\dots \leq \sum_{b \in B} \frac{120 * z_b^l * W_b * Msc}{10^{-3}}.$$

El ancho de banda disponible es limitado. La suma de los anchos de canal asignados tanto a los sectores $l \in L$ de la macro celda como los asignados a las EBBP desplegadas, debe ser menor que el ancho de banda disponible por el operador como se muestra en la Relación 8.

$$R_4 : \sum_{b \in B} z_b^l * W_b + \sum_{b \in B} Y_{p,b}^{l,h} * W_b \leq W_{disponible}. \quad (8)$$

Existe un ED válido si existe un canal asignado. Para asegurar que se asigne un canal a cada EBBP instalada en el PE $p_n^{h \in H} \in P$ en las ZAD $h \in H$ ubicadas en el sector $l \in L$, se debe cumplir la Relación 9:

$$R_5 : x_{h,f}^l \leq \sum_{p_n^{h \in H}} \sum_{b \in B} y_{p,b}^{l,h}. \quad (9)$$

Un ED en cada ZAD. Para asegurar que en la ZAD $h \in H$ sólo se asigne un ED $f \in F$, se debe cumplir la Relación 10:

$$R_6 : \sum_{f \in F} x_{h,f}^l \leq 1. \quad (10)$$

Un canal asignado para cada EBBP desplegada. Se debe de asignar sólo un ancho de canal $b \in B$ para cada ED $f \in F$ en la ZAD $h \in H$ ubicada en el sector $l \in L$, para esto, se debe cumplir la Relación 11:

$$R_7 : \sum_{b \in B} Y_{p,b}^{l,h} \leq 1. \quad (11)$$

Un canal asignado para cada sector de la MEB. Para asegurar que sólo se asigne un canal $b \in B$ en cada sector $l \in L$ de la MEB, se debe cumplir la Relación 12:

$$R_8 : \sum_{b \in B} z_b^l \leq 1. \quad (12)$$

3. Resultados de la aplicación del modelo de optimización

3.1. Escenario de análisis A (EA A)

Planteamiento

El modelo es aplicado a una MEB que está operando con una antena omnidireccional transmitiendo a una potencia de 49.5 dBm. Esta MEB se encuentra instalada a una

altura de 25 metros. Con base en la recopilación de información técnica del equipo el problema se traslada a la herramienta computacional de planeación de redes Mentum Planet [7] para determinar el área de análisis de la macro celda mediante un análisis de RSRP como se muestra en la Figura 2. Cabe mencionar que la frecuencia de operación para los dos escenarios de análisis se encuentra en la banda APT 700 MHz. Se propone un esquema de modulación promedio de 3 bits*símbolo con base a una densidad de usuarios uniforme U_m de 50 usuarios por Km^2 [4].

El área de análisis resultante es una macro celda de 800 metros de radio que eventualmente presenta tres ZAD $h_{1,1}, h_{1,2}$, y $h_{1,3}$. Se disponen de dos PE p_1^h y p_2^h para satisfacer la demanda en cada una de estas zonas como se muestra en la Figura 3. Se requiere un 95% de cobertura en las ZAD.

De acuerdo a los dos posibles emplazamientos disponibles en cada ZAD existe un total de tres ED de EBBP, $F = \{f_1, f_2, f_3\}$. Se propone una eficiencia de modulación promedio con base en el número de EBBP consideradas para desplegar [4]. La densidad de usuarios U_l^h es uniforme y se establece para este escenario de 500 usuarios por Km^2 , para las tres ZAD definidas. Además, se define un requerimiento de cobertura en estas zonas de $RC \geq 95\%$. El ancho de banda disponible es de 90 MHz.

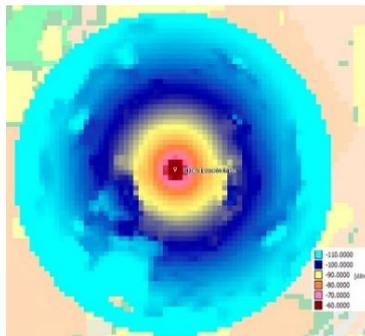


Fig. 2. Definición del área de análisis de la macro celda en el escenario de análisis A.



Fig. 3. Zonas de alta demanda (zonas rojas) y posibles emplazamientos (puntos amarillos).

En este escenario los costos por instalar una EBBP en los PE p_1^h y p_2^h son de 2,000 y 4,000 unidades genéricas de valor respectivamente, para las tres ZAD. Por otra parte, el costo por el ancho de banda de cada canal es $C_b = \{0,1,2,3,4,5,6\} * 1000$.

Resultados

El valor de la función objetivo tiene un valor de $\gamma = 31,000$ [UV]. Cabe mencionar que el modelo de optimización se resuelve en la herramienta computacional CPLEX ILOG [8]. El despliegue optimizado resultante del modelo de optimización se muestra en la Figura 4. Como se puede observar se habilitaron los emplazamientos p_1^{11} , (p_1^{12} y p_2^{12}) y (p_1^{13} y p_2^{13}) en las zonas $h_{1,1}$, $h_{1,2}$ y $h_{1,3}$ respectivamente. Además, los canales que se habilitaron son 10 MHz para p_1^{11} , 5 y 3 MHz para p_1^{12} y p_2^{12} respectivamente y un canal de 3 MHz para cada emplazamiento p_1^{13} y p_2^{13} .

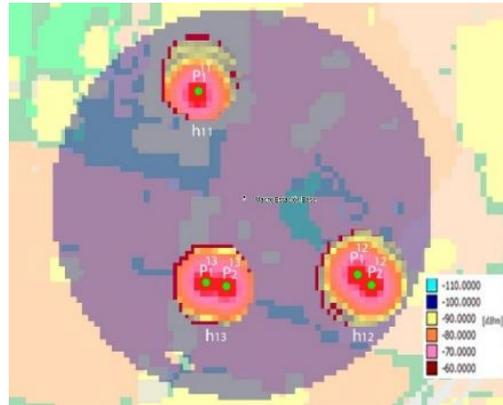


Fig. 4. Ubicación óptima de las EBBP (Despliegue optimizado).

En la Tabla 1 se demuestra que con este despliegue de EBBP se cumple que la demanda es menor a la capacidad en las ZAD.

Tabla 1. Análisis de capacidad en las zonas de alta demanda.

| Zona | Demanda total [Kbps] | Canal asignado. [MHz] | Capacidad alcanzada con el canal asignado [Kbps] | Capacidad total en la zona. [Kbps] |
|-----------|----------------------|-----------------------|--|------------------------------------|
| $h_{1,1}$ | 11,053.09 | 10 | 12,000 | 12,000 |
| $h_{1,2}$ | 14,865.53 | 5 | 12,000 | 19,200 |
| | | 3 | 7,200 | |
| $h_{1,3}$ | 11,128.09 | 3 | 7,200 | 14,400 |
| | | 3 | 7,200 | |

Por otra parte, el análisis a nivel macro celular consiste en determinar la demanda y la capacidad en los sectores de la macro celda antes y después de desplegar EBBP. El análisis se observa en Tabla 2.

Tabla 2. Análisis de capacidad en la macro celda.

| Sector | Demanda sin EBBP [Kbps] | Demanda con EBBP desplegadas [Kbps] | Demanda liberada [Kbps] | Canal asignado por el modelo [MHz] | Capacidad alcanzada con el canal asignado [Kbps] |
|--------|----------------------------|--|----------------------------|---------------------------------------|---|
| l_1 | 60,616.67 | 23,570.96 | 37,045.74 61.11% | 15 | 27,000 |

La diferencia entre la demanda antes y después del despliegue de EBBP es del 61.11% y equivale a liberar a la macro estación base (no sectorizada) de 37 Mbps. Por otra parte, con el canal asignado se alcanza una tasa de transferencia de datos de 27,000 Kbps. Por lo tanto, se satisface la demanda requerida.

3.2. Escenario de análisis B (EA B)

Planteamiento

Se propone una MEB que está operando con tres sectores de 120° , transmitiendo a una potencia (PIRE) de 60.50 dBm y se encuentran a una altura de 25 metros. El resultado del análisis de RSRP realizado en la herramienta de planeación define una celda de 900m de radio. Se propone un esquema de modulación promedio de 3 bits*símbolo con base a una densidad de usuarios uniforme U_{m_i} de 100 usuarios por Km^2 en cada sectorización de la macro celda. Se identifica una ZAD $h_{1,1}$, $h_{2,1}$ y $h_{3,1}$ para l_1 , l_2 y l_3 respectivamente, como se muestra en la Figura 5. Cabe mencionar que en cada zona se presenta un esquema de modulación promedio de 3 bits*símbolo y una densidad de usuarios uniforme de 500 usuarios por Km^2 . Adicionalmente se requiere un 80% de cobertura en estas zonas. El ancho de banda disponible es de 90 MHz.

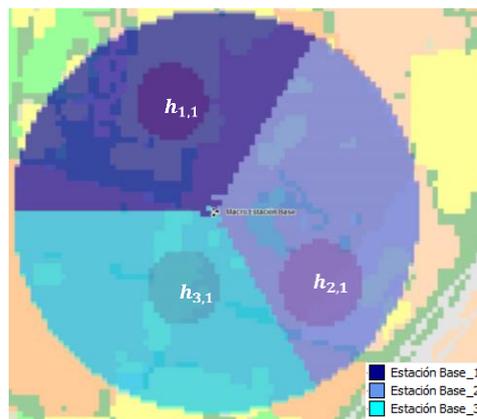


Fig. 5. Ubicación de zonas de alta demanda en los sectores de la macro celda.



Fig. 6. Zonas de alta demanda (zonas rojas) y posibles emplazamientos (puntos amarillos).

Para satisfacer la demanda en las ZAD se dispone, tres PE p_1^h , p_2^h y $p_3^h \forall h \in H$ para desplegar EBBP como se muestra en la Figura 6. De acuerdo a los tres PE disponibles en cada ZAD existe un total de siete escenarios de despliegue de EBBP, $F = \{f_1, f_2, f_3, \dots, f_7\}$. Se propone una eficiencia de modulación promedio con base en el número de EBBP consideradas para desplegar [4]. En este escenario los costos por instalar una EBBP en todos los PE es de $\{1,000$ unidades genéricas de valor. El costo por el ancho de banda de cada canal es el mismo del escenario anterior.

Resultados

El valor de la función objetivo tiene un valor de $\gamma = 31,000$ [UV]. El despliegue optimizado resultante del modelo de optimización se muestra en la Figura 7.

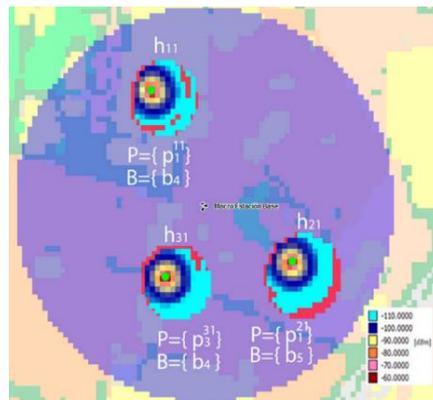


Fig. 7. Ubicación óptima de las EBBP (Despliegue optimizado).

Como se puede observar se habilitaron los emplazamientos p_1^{11} , p_1^{21} y p_3^{31} para $h_{1,1}$, $h_{2,1}$ y $h_{3,1}$ respectivamente. Además, los canales que se habilitaron para dichos

emplazamientos son 10 MHz, 15 MHz y 10 MHz. En la Tabla 3 se muestra que con este despliegue de EBBP se cumple que la demanda es menor a la capacidad en las ZAD.

Table 3. Análisis de capacidad en las zonas de alta demanda.

| Zona | Demanda total [Kbps] | Canal asignado [MHz] | Capacidad alcanzada con el canal asignado [Kbps] |
|-----------|-------------------------|----------------------------|---|
| $h_{1,1}$ | 10,159.09 | 10 | 12,000 |
| $h_{2,1}$ | 12,103.63 | 15 | 18,000 |
| $h_{3,1}$ | 9,990.91 | 10 | 12,000 |

Por otra parte, el análisis a nivel macro celular consiste en determinar la demanda y la capacidad en los sectores de la macro celda antes y después de desplegar EBBP. El análisis se observa en Tabla 4.

Table 4. Análisis de capacidad en la macro celda.

| Sector | l_1 | l_2 | l_3 |
|---|---------------------|---------------------|--------------------|
| Demanda sin EBBP desplegadas [Kbps] | 31,852.57 | 35,568.62 | 32,418.13 |
| Demanda con EBBP desplegadas [Kbps] | 21,693.48 | 23,465 | 22,490.22 |
| Demanda liberada [Kbps] | 10,159.08 31.89% | 12,103.63 34.03% | 9,990.91 30.76% |
| Canal asignado por el modelo [MHz] | 15 | 15 | 15 |
| Capacidad alcanzada [Kbps] | 27,000 | 27,000 | 27,000 |

La diferencia entre la demanda antes y después del despliegue de EBBP es la demanda que es atendida por las zonas de alta demanda en cada sector y equivale, desde este punto de vista, a liberar a los sectores l_1 , l_2 y l_3 del 31.89%, 34.03% y 30.76% de su carga respectivamente. Por otra parte, con el canal asignado a cada uno de los sectores, se alcanza una tasa de transferencia de datos de 27,000 Kbps. Por lo tanto, en todos los sectores de la macro celda se satisface la demanda requerida.

4. Conclusiones

En este trabajo se desarrolla un modelo de optimización lineal entera para mejorar el desempeño en el servicio de una red de comunicaciones móviles LTE a través del despliegue de EBBP. La función objetivo toma un valor de 31,000 [UV] para los dos EA. Cabe mencionar que se habilitaron más EBBP en el EA A con respecto al EA B pero el espectro utilizado es de 39 MHz y 85 MHz respectivamente. Utilizar menor

espectro representa un menor costo económico. El espectro utilizado es menor debido a que el RC es mayor en el EA A con respecto al EA B y la densidad de usuarios en las ZAD es menor en el EA A con respecto al EA B. La solución del modelo de optimización es encontrar la configuración que satisfaga estos requerimientos con el mínimo de recursos económicos. Otro escenario de análisis es una macro celda sectorizada donde se presentan más de una ZAD en cada sector. Este planteamiento es resuelto replicando el escenario de análisis A para las sectorizaciones correspondientes. El objetivo es demostrar que la contribución desarrollada en este trabajo puede ser aplicado a macro celdas con cualquier tipo de configuración. En futuros trabajos se considerará el manejo de interferencias aplicando como primera etapa el esquema ICIC (Inter-Cell Coordination Interference) y como segunda etapa el esquema eICIC (Enhanced Inter-cell Coordination Interference).

Agradecimientos. Los autores agradecen al Instituto Politécnico Nacional y a la Universidad de Zaragoza por el apoyo brindado para la realización de este trabajo.

Referencias

1. M. Ergen: *Mobile Broadband, Including WiMAX and LTE*. Springer (2009)
2. H. Holma, A. Toskala, J. Reunanen: *LTE Small Cell Optimization, 3GPP Evolution to Release 13*. Wiley (2016)
3. Y. Park, J. Heo, H. Kim, H. Wang, S. Choi, T. Yu, D. Hong: *Effective Small Cell Deployment with Interference and Traffic Consideration*. *IEEE Commun. Mag.* (2014)
4. Y. A. Cerecedo: *Optimización en ubicación de estaciones de baja potencia lte*. Tesis, Instituto Politécnico Nacional (2016)
5. Y. A. Cerecedo, M. Sánchez, G. Santillán: *Optimización de cobertura y capacidad en el diseño de redes WiMAX*. Tesis, Instituto Politécnico Nacional (2013)
6. A. Ghosh, J. Zhang, J. G. Andrews, R. Muhamed: *Fundamentals of LTE*. Prentice Hall. (2011)
7. M. Planet, C. Planning, T. Academy: *Coverage Planning using Mentum Planet Creating a Project*.
8. Ibm, *IBM ILOG CPLEX Optimization Studio, User's Manual* (2014)

Modelo dinámico de las afectaciones del tiempo de ciclo kanban en el flujo de producción de una línea de ensamble

Lisaura Walkiria Rodriguez Alvarado, Jesús Loyo Quijada,
Miguel Ángel López Ontiveros

Universidad Autónoma Metropolitana, UAM-Azcapotzalco, Ciudad de México, México

{Lwra, lqj, mlopez}@correo.azc.uam.mx

Resumen. En este artículo se presenta el modelo dinámico del proceso de ensamble de reductores de velocidad como parte de una actividad realizada en un taller de simulación kanban dirigido a alumnos de ingeniería industrial. El principal objetivo del modelo desarrollado es analizar el comportamiento de los niveles de inventario de las estaciones de trabajo y las acumulaciones de las tarjetas kanban (ordenes de producción) derivado de las afectaciones del tiempo de ciclo kanban en los flujos de producción. Los resultados del análisis y evaluación muestran que, cuando el tiempo de ciclo kanban es superior al tiempo de entrega del lote de producción entre estaciones, se genera una acumulación de órdenes de producción. Esto ocasiona que una vez que el flujo de producción alcanza su máximo rendimiento este disminuya. Esa situación provoca que los niveles de inventario en proceso no correspondan al ritmo del cliente, generando órdenes de producción sin entregar a tiempo.

Palabras claves: Dinámica de sistemas, flujo de producción, sistema kanban.

Dynamic Model of Kanban Cycle Time Implications in the Production Flow of an Assembly Line

Abstract. This article describes the dynamic model of assembly process of reducer is presented as part of an activity performed in a workshop kanban simulation aimed at students of industrial engineering. The main objective of the model developed is to analyze the behavior of inventory levels workstations and accumulations of kanban cards (production orders) derived from the effects of the cycle time kanban in production flows. The results of analysis and evaluation show that when kanban cycle time is longer than the time of delivery of batch production between stations, an accumulation of production orders is generated. This causes once that the production flow reaches its maximum performance, this

decrease. This situation causes the WIP levels do not correspond to the rhythm to the client, generating production orders not deliver on time.

Keywords: System dynamics, production flow, kanban system

1. Introducción

El concepto de Manufactura Esbelta (ME) evolucionó a partir de la industria Japonesa, específicamente del Sistema de Producción Toyota (SPT). En [1] la ME es considerada como una filosofía que busca maximizar el valor agregado del producto al reducir o eliminar los desperdicios involucrados en el proceso de producción, mediante la aplicación de una serie de técnicas y metodologías fundamentadas en el SPT. Una de estas técnicas es el sistema kanban usada para la reducción, control de inventarios y el manejo de producción “Justo a Tiempo” (JIT) [2].

Lage y Godinho [3] conducen en su estudio, una revisión en la literatura para identificar, clasificar y analizar las variaciones del sistema kanban propuestos por diferentes investigadores como parte de aplicación de otras metodologías de mejora continua. En su estudio evidencian las dificultades de adaptación del sistema kanban a los nuevos requerimientos productivos El entorno de modelado y simulación ofrece un potencial de gran aplicación en la actualidad para analizar el comportamiento de estos sistemas de producción [4].

En lo que respecta al análisis de modelado y simulación de sistemas kanban utilizando la metodología de Dinámica de Sistemas (DS) se han desarrollado interesantes trabajos en aplicaciones prácticas y planteamientos teóricos [5, 6].

La mayoría de estos trabajos están enfocados en el manejo y control de una línea de producción sencilla por medio de tarjetas de retiro y producción, así como el nivel de respuesta que presenta el sistema ante condiciones de incertidumbre [7, 8, 9, 10, 11, 12]. Por otro lado diferentes investigaciones consideran analizar el desempeño del sistema Kanban en una línea de producción con variedad de productos [13, 14, 15], y tomar en cuenta los efectos de diferentes factores operacionales en el desempeño del sistema productivo controlado por tarjetas kanban [16, 17, 18, 19].

En aplicaciones recientes se ha considerado emplear el sistema kanban en conjunto con otros sistemas control conocidos como sistemas Híbridos. [20, 21, 22, 23]. En este trabajo se presenta un modelo dinámico desarrollado a partir de una actividad didáctica de un taller de simulación kanban dirigido a alumnos de ingeniería industrial. El modelo dinámico está enfocado al análisis de los efectos del tiempo del ciclo kanban en el flujo de producción de una línea de ensamble de reductores de velocidad, en el cual se ejemplifica el comportamiento real de la aplicación del sistema kanban en un ejercicio simulado con los alumnos. El modelo permite explorar el comportamiento de la línea de ensamble ante el efecto de las incidencias ocasionadas por las variaciones en el tiempo de ciclo kanban, tomando como parámetros de análisis el inventario en proceso, tiempo de entrega y acumulación de órdenes de producción.

2. Funcionamiento del sistema kanban

El sistema kanban es definido como un mecanismo de control para el flujo de material [24] mediante la utilización de tarjetas de producción y retiro de producto, las cuales garantizan la producción de la cantidad adecuada en el momento adecuado. Este sistema permite que cada estación de trabajo opere con los niveles mínimos de inventario, al procesar únicamente lo solicitado por la estación siguiente [25, 26].

La explicación del funcionamiento del sistema kanban se explica con ayuda de la Fig. 1. La producción de la última estación de la línea de ensamble, en este caso montaje, es activada una vez que el cliente realiza una solicitud. En ese momento se toma material de esa estación para cubrir el requerimiento del cliente, al mismo tiempo se activa una tarjeta kanban (orden de producción), la cual en función con los niveles de inventario de la estación montaje, las estaciones engrane, cárter, corona A y corona B y los órdenes del cliente, activa la cantidad a cubrir de acuerdo a lo requerido por la estación montaje, lo cual debe ser igual a lo solicitado por el cliente. Esto ocurre cada cierto tiempo determinado por el *tiempo de ciclo kanban*¹,

3. Desarrollo del modelo dinámico

3.1. Entorno de simulación

El modelo desarrollado es una representación de lo que ocurre realmente durante el taller de simulación kanban dirigido a alumnos de ingeniería industrial. En el taller los estudiantes comprenden los mecanismos claves en la gestión de la producción mediante la interacción y juego de roles desempeñado por ellos mismos.



Fig. 1. (a) Taller de simulación kanban. (b) Proceso de ensamble del reductor.

Como parte de la didáctica del taller se establece una línea de ensamble de 4 estaciones de trabajo para fabricar los componentes que componen los reductores de velocidad. Cada componente en cada estación de trabajo se fabrica a partir de materia prima tomada en el stock de las estaciones de engrane, cárter, corona A y corona B (en

¹ El tiempo interno desde que la tarjeta kanban es tomada de la caja de colección

este caso, la estación corona A surte a estación corona B). Posteriormente se reúnen los componentes semiprocesados en la estación montaje para realizar el ensamble del producto final.

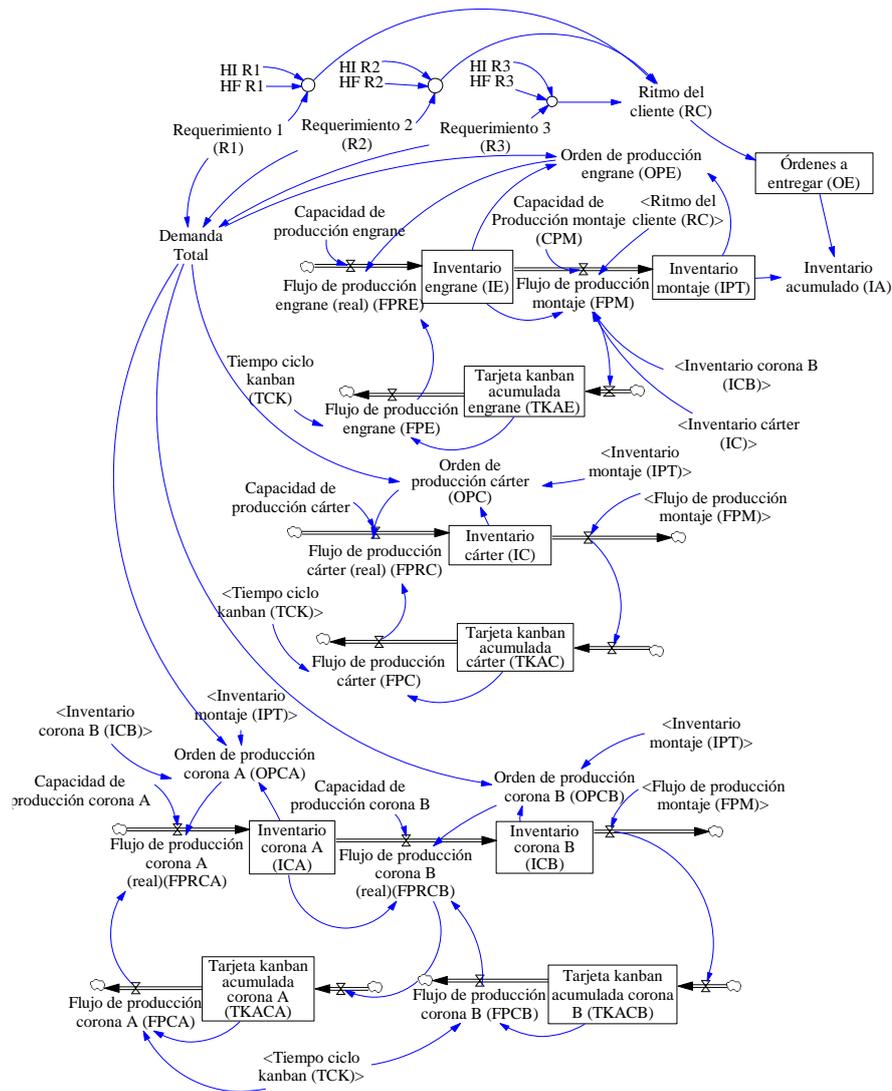


Fig. 2. Modelo dinámica de la línea de ensamble de un reductor.

3.2. Estructura del modelo

En Fig. 2 se presenta el modelo desarrollado, el cual permite analizar el comportamiento dinámico del flujo de producción de las estaciones engrane, cárter,

corona A, corona B y montaje, controladas por las tarjetas kanban. Se toma como principal referencia de análisis para el modelo la acumulación de los niveles de inventario, acumulación de tarjetas kanban, su respectivo tiempo de ciclo y las órdenes de producción pendientes a entregar. La interrelación de estas variables determina la secuencia productiva de cada estación controlada por las tarjetas Kanban.

El análisis de simulación inicia una vez que el cliente establece la *Demanda Total* (comprendida en este caso por tres requerimientos en una jornada laboral). El *Ritmo del cliente* es quien determina la velocidad del flujo de producción de la última estación en el proceso de ensamble del reductor, en este caso corresponde a la estación Montaje. Cuando se presenta una diferencia entre el comportamiento del *flujo de producción de la estación montaje*, (determinada por su *capacidad* y el *ritmo del cliente*) y el tiempo de reacción de cada estación predecesora (determinado por el *tiempo de ciclo kanban*), ocasiona una acumulación de *Tarjetas Kanban*. De esta forma, una estación produce únicamente como respuesta a una orden de reposición inmediatamente posterior de productos consumidos por la estación montaje.

Estas características determinan el *Flujo de producción real* para cada estación, el cual se ve restringido por la *Capacidad de producción* y las *Órdenes de Producción* a cumplir. Por otro lado, cada Orden de producción para cada estación, se determina por la *Demanda Total* establecida por el cliente y por los niveles de *Inventario en proceso* de cada componente, a mayor nivel de inventario existentes, las órdenes de producción son menores. Bajo estas consideraciones, en el comportamiento del proceso productivo se pueden presentar dos situaciones: La primera es que la velocidad de acumulación del inventario de producto terminado sea menor al ritmo establecido por el cliente ocasionado por las restricciones de capacidad de la misma línea. Y la segunda situación es que el sistema de control por medio de las tarjetas Kanban garantice que la velocidad de acumulación del inventario de producto terminado sea igual al ritmo establecido por el cliente.

3.3. Conceptualización de las variables del sistema

Para poder analizar y representar el comportamiento dinámico del proceso de ensamble de los reductores de velocidad en las diferentes estaciones de ensamble se consideraron las siguientes variables principales.

Ritmo del cliente (RC). Establece la velocidad con la cual el cliente solicita una determinada cantidad de producto, *Requerimiento_i*, (R_i). Para determinar el RC se toma en consideración el tiempo disponible que tienen la línea de ensamble (HI_i y HF_i) para dar respuesta a dicha solicitud y la cantidad demandada en dicho periodo. En la ecuación 1 se presenta la función “step²”, sus dos argumentos determinan el punto de inicio y el tiempo en que tarda en completarse el requerimiento solicitado. Por otro lado, la sumatoria de los requerimientos determina la *Demanda Total (DT)*.

² Esta función permite que los datos tengan un comportamiento como entrada a escalón. El valor en esta función tiene un punto de inicio, se mantiene constante y en un punto final vuelve a su condición inicial. Los puntos de inicio y finalización están determinados por HF y HI:

$$Rc = \sum_{i=1}^{I-1} ((\text{Step}((R_i/(HFR_i - HIR_i)), HIR_i) - \text{Step}((R_i/(HFR_i - HIR_i)), HFR_i))). \quad (1)$$

Flujos de producción. Determinan la velocidad de producción de cada una de las estaciones para el ensamble del reductor.

- Flujo de producción montaje (*FPM*). La velocidad del flujo de producción de esta estación está condicionada por la cantidad de inventario mínimo disponibles de producto semiensamblado de las estaciones anteriores (*IE inventario engrane*, *IC inventario cárter*, *ICA inventario corona A*, *ICB inventario corona B*), la capacidad de producción de la estación montaje (*CPM*) y el *RC*. La ecuación 2 establece esta relación:

$$FPM = \text{Min} (\text{Min} (IE, IC, ICB), (CPM, RC)). \quad (2)$$

- Flujo de producción de las demás estaciones de trabajo (*FPE*, *FPC*, *FPCA*, *FPCB*). Estos flujos están determinados por la acumulación de tarjetas kanban (*TKA*) en cada estación y su respectivo tiempo de entrega establecido por el tiempo de ciclo kanban (*TCK*). Donde, la acumulación de las tarjetas se pueden representar como un inventario en proceso de acuerdo con la Ley de Little, la cual establece que, la tasa de producción = inventario en proceso / tiempo de entrega, esta relación se presenta en ecuación 3:

$$FP_{\text{estación}} = TKA_{\text{estación}} / TCK. \quad (3)$$

- Flujo de producción real de las demás estaciones de trabajo (*FPRE*, *FPRC*, *FPRCA*, *FPRCB*). Estos flujos están condicionados por las órdenes de producción a cumplir (*OP*), capacidad de producción (*CP*) y por el flujo de producción de las estaciones de trabajo derivado de la relación entre el número de tarjetas kanban y tiempo ciclo. En el caso del *FPRCB* tiene una restricción adicional correspondiente al nivel de inventario de Corona A. A pesar de la acumulación de las tarjetas Kanban las estaciones de trabajo no pueden producir más de lo establecido por las *OP* y su restricción de *CP* esta relación se presenta en ecuación 4:

$$FP_{\text{Restación}} = \text{If then else} (OP_{\text{estación}} \leq 0, 0, \text{Min} (CP_{\text{estación}}, FP_{\text{estación}})). \quad (4)$$

Orden de producción (*OP_{estación}*). La *OP* de cada estación establece la cantidad máxima a producir, la cual está determinada por la diferencia de la *DT* establecida por el cliente y los niveles de inventarios de las estaciones siguientes y los niveles de su propia estación. La ecuación 5 describe el comportamiento de dicha variable:

$$OP_{\text{estación}} = DT - I_{\text{estación actual}} - I_{\text{estación siguiente}}. \quad (5)$$

Niveles de inventario. Determinan las acumulaciones de inventario en proceso y producto terminado.

- Inventario de producto terminado (*IPT*). Representa la acumulación de material procesado procedente de la estación montaje, su expresión está representada por la ecuación 6:

$$IPT(t) = FPM(t). \quad (6)$$

- Inventario de las demás estaciones (*IE, IC, ICA, ICB*). Representa la acumulación de material semiprocesado procedente de las demás estaciones de trabajo anteriores a la estación montaje. Su comportamiento está determinada en ecuación 7, establecida por la diferencia de flujos de producción de la estación anterior y la estación montaje

$$I_{estación}(t) = F_{Pestación}(t) - FPM(t) \quad (7)$$

- Tarjeta kanban acumulada (*TKA*). Establece la acumulación de las tarjetas Kanban procedentes de estación montaje o estación corona B a lo largo del tiempo. Este nivel de acumulación se presenta debido a la diferencias de los flujos de producción de estas estaciones y el flujo de producción de la estación anterior. La ecuación 8 representa el comportamiento de dicha acumulación:

$$TKA_{estación}(t) = FPM(t) - F_{Pestación}(t). \quad (8)$$

- Inventario Acumulado (*IA*). Establece la diferencia entre los niveles de inventarios procedentes de la estación montaje y las órdenes a entregar (*OE*) determinadas por el ritmo del cliente, la ecuación 9 establece su comportamiento. Los inventarios iniciales de cada estación se balancean considerando el ritmo del cliente por el tiempo de entrega, así si el cliente solicita 100 unidades en la primera hora y el tiempo de entrega de la estación montaje es de una hora, el inventario inicial es igual a 100 Uds:

$$IA(t) = IPT(t) - OE(t). \quad (9)$$

4. Simulación y evaluación

Para analizar el comportamiento dinámico del modelo desarrollado se recolectaron datos derivados de la actividad didáctica del taller de simulación kanban. Los participantes en sus diferentes roles de analistas y operadores, registran las características de los requerimientos del cliente a cumplir (R_1, R_2, R_3), así como el tiempo de entrega del producto final. Los resultados de las corridas de simulación en el taller permitieron obtener los datos que alimentan el modelo dinámico, desarrollado en el software Vensim Ple, para esto se consideraron dos escenarios de análisis. En el primer escenario se considera un tiempo de ciclo kanban (TCK) igual al tiempo de entrega de tamaño de lote de las estaciones de trabajo (1 hora). En un segundo escenario, se considera un tiempo de ciclo kanban igual a dos horas. Los valores de los parámetros y condiciones iniciales seleccionadas y secuencia de producción para verificar la validez de las ecuaciones y la veracidad de los resultados obtenidos, se muestran en Tabla 1.

El tiempo de ciclo kanban afecta directamente el comportamiento del flujo de producción de las estaciones anteriores a la estación montaje, ya que determina el tiempo de solicitud real de cada orden de producción. Esto ocasiona que los niveles de

inventario en proceso varíen de acuerdo a este comportamiento incrementando o disminuyendo de acuerdo al ritmo establecido por el incremento de las tarjetas kanban (ecuación 7).

Tabla 1. Parámetros y condiciones iniciales.

| Parámetro | Valor | Unidad |
|---|-------|-----------|
| Capacidad de producción de engrane (<i>CPE</i>) | 100 | Uds./hora |
| Capacidad de producción de cárter (<i>CPC</i>) | 100 | Uds./hora |
| Capacidad de producción de corona B (<i>CPCB</i>) | 200 | Uds./hora |
| Capacidad de producción de corona A (<i>CPCA</i>) | 200 | Uds./hora |
| Capacidad de producción de montaje (<i>CPM</i>) | 100 | Uds./hora |
| Tiempo de ciclo kanban (TCK) primer escenario | 1 | hora |
| Tiempo de ciclo kanban (TCK) segundo escenario | 2 | horas |
| Requerimiento R1 | 100 | Uds. |
| Requerimiento R2 | 500 | Uds. |
| Requerimiento R3 | 200 | Uds. |
| Hora de inicio R1 (HI_1) y Hora de Finalización R1 (HF_1) | 0-1 | Hora |
| Hora de inicio R2 (HI_2) y Hora de Finalización R2 (HF_2) | 2-6 | Hora |
| Hora de inicio R3 (HI_3) y Hora de Finalización R3 (HF_3) | 6-8 | Hora |

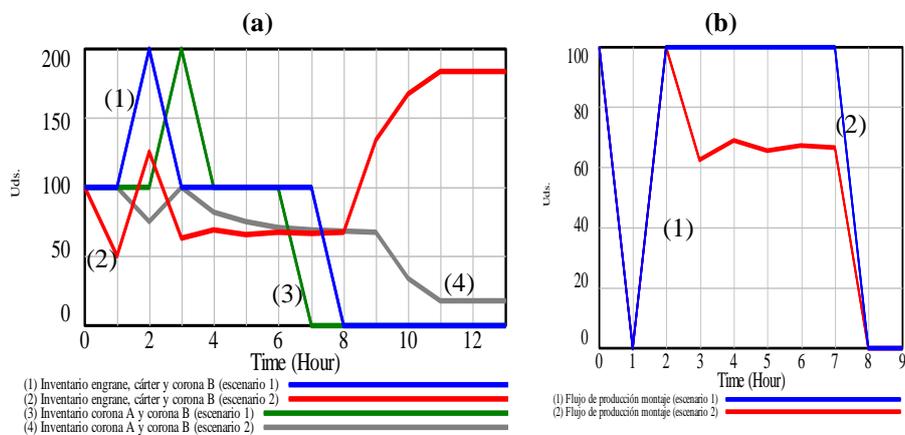


Fig. 2. (a) Comportamiento dinámico de los niveles de inventario en estaciones de trabajo (engrane, cárter, corona A, corona B) escenario 1 y 2. (b) Flujo de producción en estación Montaje.

En Fig. 2 (a) se puede observar que, cuando el tiempo de ciclo kanban es igual a 1 hora los niveles de inventario de engrane, cárter, corona A y corona B se mantienen constantes a un nivel de 100 unidades, con un incremento a 200 unidades desde la hora 1 y 2 hasta la hora 3 y 4. Esto indica que los niveles de inventario de estas estaciones mantienen un nivel mínimo correspondiente al ritmo de producción establecido por el cliente. Por otro lado, al analizar el comportamiento de los niveles de inventario de

estas mismas estaciones cuando el tiempo de ciclo kanban es de 2 horas se observa que dichos niveles se mantienen por debajo de las 100 unidades. Esto indica que los flujos de producción no están produciendo a la máxima capacidad permitida ni al ritmo del cliente establecido, esto se debe a que las órdenes procedentes de las tarjetas kanban presentan un retraso en su tiempo de entrega. Este retraso se transmite a cada flujo de producción de las estaciones engrane, cárter, corona A y B, por lo que su ritmo de producción es inferior al ritmo solicitado por el cliente. Los flujos de producción de estas estaciones, fluctúan de acuerdo al comportamiento de la acumulación de las tarjetas kanban, lo cual se ve reflejado en la acumulación de sus niveles de inventario. En Fig. 2 (b) se ejemplifica este comportamiento con el flujo de producción de estación Montaje para ambos escenarios. Es notorio observar las fluctuaciones del flujo productivo de la estación montaje cuando el tiempo de ciclo kanban es igual a 2 horas. Este comportamiento se debe al retraso ocasionado por el tiempo de entrega de las tarjetas kanban, lo que ocasiona que el flujo productivo de las estaciones predecesoras esté por debajo de la capacidad máxima permitida. Ocurre un caso totalmente diferente cuando el tiempo de ciclo kanban es igual a 1 hora, ya que el flujo productivo de la estación montaje se mantiene estable a lo largo de la jornada laboral, sin interrupciones y trabajando a la capacidad máxima permitida. Esto garantiza que se de un cumplimiento efectivo de las órdenes de producción.

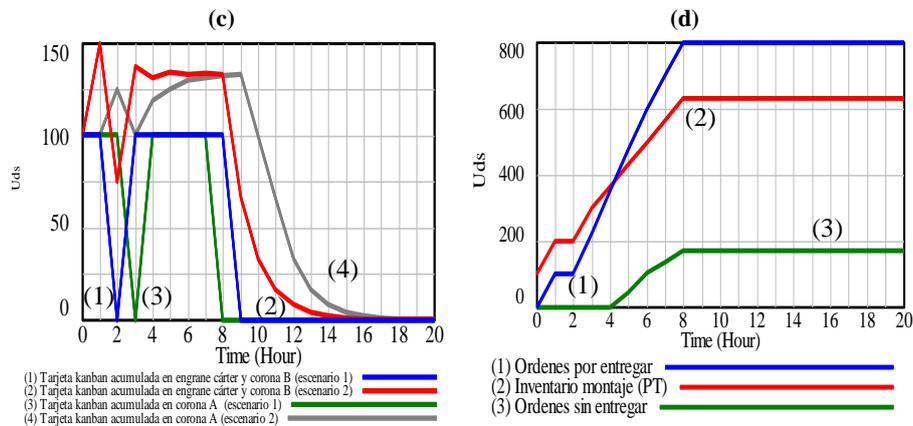


Fig. 3. (c) Acumulación de tarjetas kanban en las estaciones de trabajo. (d) Órdenes pendientes por entregar.

La acumulación de las tarjetas kanban indican las órdenes de producción en cada estación de trabajo. Su nivel de acumulación responde a la diferencia del flujo de producción montaje y el flujo de producción de cada estación anterior a esta. En Fig. 3 (a) se observa que, los niveles de inventario de las tarjetas kanban o bien, la cantidad de órdenes acumuladas cuando el tiempo de ciclo kanban es igual a 1, se mantienen relativamente constantes, igual a 100 unidades lo que corresponde al ritmo establecido por el cliente. Por otro lado, cuando el tiempo de ciclo kanban incrementa a dos horas, es evidente el incremento de las órdenes de producción para dichas estaciones, alcanzando niveles de 130 unidades a producir por hora. Este comportamiento

sobrepasa directamente la capacidad de producción establecida por las estaciones de trabajo, por lo que los flujos de producción únicamente procesarán lo permitido de acuerdo a sus restricciones de capacidad. Esta situación ocasiona que se generen órdenes sin entregar a tiempo. En la Figura 3(d) se presenta que las órdenes sin entregar se presentan a partir de la hora 4 y continúan acumulándose hasta la hora 8 con un total de 180 unidades. Esta situación se presenta a partir de que el inventario en la estación montaje no alcanza los niveles máximos de 800 unidades debido al retraso generado por el tiempo de resurtimiento de las tarjetas kanban entre estaciones lo que ocasiona que no trabajen a su máxima capacidad, por lo tanto la estación montaje no tiene suficiente material para cumplir con las órdenes por entregar solicitadas por el cliente. Este comportamiento responde a la diferencia de los flujos de producción de las estaciones de trabajo afectadas por la variación en los tiempo de ciclo kanban, lo que ocasiona que las órdenes de producción no sean entregadas a tiempo.

5. Conclusiones

La DS permite crear diferentes escenarios para analizar el comportamiento de sistemas productivos bajo diferentes consideraciones. En este caso de estudio se determinan las afectaciones de las variaciones del tiempo de ciclo kanban en una línea de ensamble de reductores, como parte de un ejercicio de simulación para analizar el comportamiento del sistema controlado por tarjetas kanban. Para esto, se analizaron dos escenarios comparativos de las variaciones del tiempo de ciclo kanban. En el primer escenario se consideró que el tiempo de ciclo kanban es igual a 1 hora, aquí se presenta que, los niveles de inventario entre las estaciones involucradas en el proceso mantendrán siempre un nivel máximo permitido (generalmente, es el nivel de producción en una hora de producción), para este caso, para las estaciones Engrane y cárter es igual a 100 unidades, y para las estaciones Corona A y B es igual a 200 unidades. Al mantener un flujo productivo igual al ritmo de producción del cliente y niveles de inventarios mínimos, se garantiza que el flujo de producción de la estación montaje sea estable y se complete las órdenes de producción, bajo consideración de su máxima capacidad productiva. Por otro lado, en el segundo escenario se consideró que el tiempo del ciclo kanban es de 2 horas. Para este caso se evidencia que, se genera una acumulación de órdenes de producción, las cuales son completadas en cada estación de acuerdo a su máxima capacidad permitida. Esto ocasiona que los flujos de producción de las estaciones trabajen a un rendimiento menor al permitido ya que se ven afectado por el retraso en la entrega de las tarjetas kanban que a su vez determinan las órdenes de producción a completar. Por lo tanto los niveles de inventario de producto en proceso en cada estación no corresponden al ritmo establecido por el cliente lo que de igual forma ocasiona acumulación de ordenes pendientes. Para contrarrestar esta situación y satisfacer el creciente número de órdenes de producción se debe incrementar el rendimiento o la capacidad del flujo de producción de las estaciones antecesoras a la estación montaje, ya que una vez que las estaciones alcanzan su máximo rendimiento no se registra ninguna mejora, lo que da paso a que se genere una acumulación de las órdenes sin entregar a tiempo. El análisis de las situaciones presentadas en ambos

escenarios permite una válida representación de la situación real expuesta en el taller de simulación del Sistema kanban. Esto permitió que los alumnos comprendieran la importancia de determinar correctamente el tiempo de ciclo kanban y sus afectaciones en el sistema productivo.

Referencias

1. Sundar, R., Balaji, A. N., Satheesh Kumar, R. M.: A Review on Lean Manufacturing Implementation Techniques. *Procedia Engineering* 97, pp. 1875–1885 (2014)
2. Abdul, N. A., Mohd, S., Mohamed, M.: Lean Manufacturing Case Study with Kanban System Implementation. *Procedia Economics and Finance* 7, pp. 171–180 (2013)
3. Lage, M., Godinho, M.: Variations of the Kanban System: Literature Review and Classification. *Int. J. Production Economics* 125, pp. 13–21. (2010)
4. Mohsen, J., Tillal, E., Aisha, N., Lampros, S., Terry, Y.: Simulation in manufacturing and business: A review. *European Journal of Operational Research* vol. 203, pp. 1–13 (2010)
5. Tregubov, A., Lane, A.: Simulation of Kanban based scheduling for Systems: initial results. *Procedia Computer Science* 44, pp. 224–233 (2015)
6. Halim, N., Yusuf, N., Jaafar, R., Jaffar, A., Kaseh, A., Azira, N.: Effective Material Handling System for JIT Automotive Production Line. *Procedia Manufacturing* 2, pp. 251–257 (2015)
7. O'Callaghan, R.: A System Dynamics Perspective on JIT-Kanban. In: International Conference of the System Dynamics Society, pp. 959–1004 (1986)
8. Patel, K., Thanki, S. J.: System Dynamics Modelling an Analysis of a Single Stage Single Product Kanban Production System. *International Journal of Innovative Research in Science, Engineering and Technology*, Vol. 2, pp. 2262–2270 (2013)
9. Ebrahimpour, M., Fathi, B. M.: Dynamic Simulation of a Kanban Production Inventory System. *International Journal of Operations & Production Management*. Bradford: Vol. 5, Issue 1, pp. 5–10 (1985)
10. Guerra, L., Murino, T., Romano, A.: A System Dynamics Model for a Single-Stage Multi-Product Kanban Production System. *Recent Advances in Automation & Information*, pp. 171–176 (2010)
11. Ahmad, N., Ahmed, B., Noriah, B., Nurul H.: Implementation of Just in Time Production through kanban System. *Industrial Engineering Letters*, vol. 3, No. 6, pp. 11–20 (2013)
12. Lummus, R.: A simulation analysis of sequencing alternatives for JIT lines using kanbans. *Journal of Operations Management*, vol. 13, pp. 183–191 (1995)
13. Chan, F.T.S.: Effect of kanban size on just-in-time manufacturing systems. *Journal of Materials Processing Technology*, 116, pp. 146–160 (2001)
14. Murino, T., Naviglio, G., Romano, E., Zoppoli, P.: Single stage multi product kanban system. Optimization and parametric analysis. In: Proceedings of the 8th WSEAS International Conference on System Science and Simulation in Engineering, pp. 313–318 (2009)
15. Mehmet, S., Abdullah Al-J.: Simulation analysis of just-in-time production system. *Int. J. Production Economics*, vol. 42, pp. 67–78 (1995)
16. Matta, A., Dallery, Y., Mascolo M.: Analysis of assembly systems controlled with kanbans. *European Journal of Operational Research*, vol. 166, pp. 310–336 (2005)
17. Paolo, R.: Dynamic Control Card in a Production System Controlled by Conwip Approach. *Journal of Mechanical and Industrial Engineering*. Vol. 4, Number 4, pp. 425–432 (2010)
18. Sagar, M., Subhash, R.: Development of kanban system for valve manufacturing industries using Just in time principle. *IJRASET*, Vol. 3, Issue III, pp. 294–298 (2015)

19. Gamberini, R., Meli, M., Galloni, L., Rimini, B., Lolli, F.: Alternative refilling policies for an assembly line managed by kanbans. *Management, and Control International Federation of Automatic Control*, pp. 1914–1919 (2013)
20. Qi, H., Weiming, S.: Implementing a hybrid simulation model for a Kanban-based material handling system. *Robotics and Computer-Integrated Manufacturing*, pp. 635–646 (2008)
21. Jagjit, K.: System dynamics modelling and analysis of just-in time manufacturing systems. University of Windsor, *Electronic Theses and Dissertations*, Paper 3713 (1991)
22. Alvin, A.: A performance comparison of single product kanban control systems. *International Journal of Production Management and Engineering*, pp. 57–74 (2014)
23. Chukwunonyelum, E.: Effect of product mix on multi-product pull control. *Simulation Modelling Practice and Theory* 56, pp. 16–35 (2015)
24. Graves, R., Konopka, J.M., Milne, R.J.: Literature review of material flow control mechanisms. *Production Planning and Control* 6 (5), pp. 395–403 (1995)
25. Shaojun, W., Bhaba, S.: Production, Manufacturing and Logistics Optimal models for a multi stage supply chain system controlled by kanban under just-in-time philosophy. *European Journal of Operational Research*, vol. 172, pp. 179–200 (2006)
26. Hussein, M. R.: A Review of Kanban – the Japanese Just in Time Production System. *Engineering Management International*, vol. 4, pp. 143–150 (1987)

Propuesta de un modelo de integración de PSP y Scrum para mejorar la calidad del proceso de desarrollo en una MiPyME

Mauricio Leonardo Urbina Delgadillo, María Antonieta Abud Figueroa,
Gustavo Peláez Camarena, Giner Alor Hernández, Alma Ivonne Sánchez García

Instituto Tecnológico de Orizaba, División de Estudios de Posgrado e Investigación, Veracruz,
México

mauricio.urbina.isc@gmail.com, mabud@ito-depi.edu.mx, sgpelaez@yahoo.com.mx,
galor@itorizaba.edu.mx, alivsaga@hotmail.com

Resumen. El desarrollo de software dentro del ambiente empresarial se caracteriza por ser rápido, cambiante y agresivo en los tiempos de entrega. Las empresas pequeñas necesitan construir un producto de calidad con pocos recursos para incrementar la satisfacción del cliente. La combinación de métodos ágiles y procesos disciplinados es una opción factible que promete mejorar la gestión del desarrollo de software y la calidad del software. Por esta razón, en este trabajo se presenta una propuesta de un modelo de integración entre Scrum y PSP (*Personal Software Process*) para mejorar la calidad del proceso de desarrollo para MiPyMEs donde los equipos de trabajo sean reducidos y se desea integrar la adaptabilidad y predictibilidad a sus prácticas. El resultado de esta investigación es un modelo de mejorar de procesos de desarrollo de software aplicable cualquier proyecto.

Palabras clave: Calidad de software, gestión del proceso de software, mejora de proceso de software, proceso personal de software, PSP, scrum.

PSP and Scrum based Integration Model to Improve Software Quality Process for MiPyME

Abstract. Software development into the enterprise environment is defined by speed, volatile and aggressive delivery times. Small enterprises need to build a quality product with limited resources for increase customer satisfaction. Combination of agile methods and process disciplines is a feasible option to improve the management of software development and software quality. In this paper we propose an integrated process model between Scrum and PSP (*Personal Software Process*) to improve the quality of the development process for MiPyME's, which have small work teams and want to integrate adaptability and predictability into their practices. As a result of this research, a model for improve software development processes for any project is presented.

Keywords: Software quality, software process management, software process improvement, personal software process, PSP, scrum.

1. Introducción

La demanda de la calidad del software es un punto importante para construir una relación de confianza entre la empresa y el cliente. Según [1] el 70% de la fallas son introducidas en las fases de ingeniería de requerimientos, diseño del sistema, arquitectura y diseño de los componentes; el otro 30% corresponde a las fases de codificación, pruebas e integración. Aún cuando el 50.5% de esas fallas sean encontradas en las fases de pruebas e integración, todavía el 21% son encontrados en la fase operativa, es decir, por parte del usuario. Este problema surge de dos puntos de vista; el primero se refiere a la demanda del tipo de aplicaciones y la complejidad que éstas presentan a la hora del desarrollo y, el segundo está relacionado con la competitividad por entregar un mejor producto en poco tiempo, de bajo presupuesto y que cumpla con las características del cliente.

La combinación de métodos ágiles (*SCRUM*, *XP*, *Lean Development (LD)*, *Ágile*) y disciplinados (*CMMI-DEV*, *Six Sigma* y *PSP/TSP*), es una solución cada vez más aceptada por los ingenieros de software, para contrarrestar las deficiencias que traen consigo el utilizar un solo método [2], proporcionar mejoras al desarrollar un producto de calidad, mejorar la gestión del proceso y satisfacer las necesidades del cliente.

Esta investigación está enfocada en la combinación entre Scrum y PSP para mejorar la calidad del desarrollo de software. Por esta razón se propone un modelo de procesos para integrar la agilidad y flexibilidad de Scrum con la disciplina que caracteriza a PSP, con el propósito de mejorar el proceso para desarrollo de software para MiPyMEs. Este documento se estructura de la siguiente manera: en la Sección 2 se mencionan trabajos relacionados con la investigación. En la Sección 3 se presentan la justificación de la selección de Scrum y PSP, además de una pequeña definición de éstos. La Sección 4 presenta la propuesta de integración entre Scrum y PSP. Las conclusiones y las líneas de trabajo futuro se muestran en la Sección 5.

2. Trabajos relacionados

Para conocer una perspectiva clara y sencilla de los trabajos relacionados se presenta en la Tabla 2, un resumen de las diferentes adopciones por parte de empresas de software con modelos disciplinados y/o métodos ágiles.

Tabla 1. Análisis comparativo entre trabajos con diferente adopciones entre métodos ágiles y métodos disciplinados.

| Objetivo / Planteamiento | Solución | Resultados |
|--|---|---|
| Lina, 2012 [3] | | |
| Realizar una correlación entre <i>CMMI</i> y <i>Scrum</i> , mostrando diferencias entre ellos e identificando como las PyMEs están adoptando prácticas en sus proyectos para hacer estos enfoques más dóciles. | La integración de las siguientes actividades: <i>Nivel del Proceso de la Organización</i> , <i>Administración de las Actividades del Proyecto</i> , <i>Gestión de Riesgos</i> , <i>Soporte a la Gestión de Configuración de Procesos</i> , <i>Aseguramiento de la</i> | <i>CMMI</i> y <i>Scrum</i> puede complementarse una a la otra creando una sinergia que beneficia a la organización que los utiliza. Resultando útil para organizaciones que tiene un proceso basado en métodos disciplinarios y planean |

| Objetivo / Planteamiento | Solución | Resultados |
|--|--|--|
| | <i>Calidad y Construir un Nuevo Ciclo de Vida de Software basado en Scrum.</i> | mejorar la agilidad de los procesos. |
| Bougroun, 2014 [4] | | |
| Proyección entre CMMI 3 y tres prácticas ágiles: Scrum, Kanban y XP. Su planteamiento parte de otros autores que realizaron análisis para determinar la compatibilidad entre métodos ágiles y CMMI. | Mapeo extenso y detallado de áreas de CMMI: <i>Desarrollo de requerimientos, Solución Técnica, Integración del Producto, Verificación, Validación, Concentra en el proceso de la organización, Definición del proceso de la organización, Entrenamiento de la Organización, Gestión de la integración del proyecto, Gestión de riesgos, Resolución y Análisis de decisiones.</i> | Se muestran una cobertura, de CMMI Niv. 3, con Scrum del 44% involucrado en las áreas de administración y procesos de la organización, con XP con el 45% en los detalles e implementación de procesos y con Kanban con el 6% en el aspecto de control y decisiones. |
| Brown, 2014 [5] | | |
| Las diferentes adaptaciones de prácticas de PSP con <i>Scrum, PSP y XP</i> dificultan e impiden la reutilización en otras metodologías. Generar un modelo utilizando el núcleo SEMAT para eliminar esta limitante. | Se muestra a PSP como una alpha “ <i>way of working</i> ”. Los 7 niveles de PSP son representados como estados de un sub-alpha “ <i>PSP Compliance</i> ”. Esta representación da a conocer a todo el equipo en qué nivel PSP se encuentran. Igualmente se adapta Scrum utilizando el Backlog, el incremento y el Sprint como alphas dentro de SEMAT. | Esta solución se adapta a Scrum donde su capacidad de adaptación se aprecia, debido a que la representación de las actividades de PSP que tienen que ser realizadas por el equipo. |
| Romano, 2015 [6] | | |
| Aborda el uso de Scrum dentro de una empresa pequeña. Proporciona actividades de monitoreo para la gestión de proyectos de software, lo que implica el uso de recursos humanos, financieros y temporales durante el desarrollo de un proyecto, aumentando la velocidad de sus entregas exitosas. | Una vez definido el método ágil se llevó a cabo su realización, en una empresa de Brasil, fue necesario mencionar los pasos del ciclo: <i>Selección de la Infraestructura, El Equipo de Entrenamiento, Despliegue de Scrum, Refinamiento del Despliegue.</i> | Puntos positivos: mejora en la calidad y tiempo estimado, se fortaleció el trabajo en equipo, los desarrolladores reconocieron las deficiencias por la falta de un método para la toma de decisiones, el uso de <i>Kanban</i> facilitó la concentración y comprensión de los objetivos y tareas, <i>Planning Poker</i> es de utilidad para las estimaciones de |

| Objetivo / Planteamiento | Solución | Resultados |
|---|---|--|
| | | tiempo y el gráfico <i>Burndown</i> muestra el progreso del Sprint y las tareas que se tiene que completar. |
| Soares, 2015 [7] | | |
| Las iniciativas de transición a métodos ágiles fallan debido a que los empleados creen que el proceso no encaja, el cambio es obligado y sin consultarlo antes. Basándose en este escenario de define una estrategia ágil que pueda ayudar a las empresas a implementar prácticas de administración de proyectos. | En total, esta propuesta definió 84 maneras para implementar, de una forma ágil, la práctica de gestión de proyectos y 38 productos de trabajo como parte de la estrategia para implementación de la administración de proyectos ágiles para empresas que buscan CMMI. | Para guiar a las empresas dentro este tipo de escenarios, este trabajo presentó una estratégica que apoya de manera gradual y disciplinada la puesta en marcha de la gestión de proyectos ágiles basado en marcos de trabajo y métodos previamente validados y en su creciente uso ara la comunidad de desarrollo de software. |
| Arauz Ortiz, 2016 [8] | | |
| Describe la experiencia de una empresa mexicana de desarrollo de software que integra métodos ágiles dentro del proceso de desarrollo CMMI-DEV Niv. 5. El objetivo del caso de estudio fue documentar los efectos, cuantitativos y cualitativos, reflejados en el desempeño de sus proyectos. | La motivación fue conocer las técnicas ágiles que integró Praxis (organización en el área de Tecnologías de la Información) con CMMI-DEV para establecer una nueva forma de construir software y mejorar la productividad. Las unidades analizadas fueron: <i>Técnicas ágiles integradas con CMMI-DEV, Métricas de proyectos para determinar beneficios, Adaptación del grupo de trabajo.</i> | Los efectos de la integración tuvieron un impacto sobre los integrantes de los equipos, el cliente y la misma organización. Algunos beneficios obtenidos fueron: <i>Enfoque en producto y no en documentación, mayor control de actividades, corrección oportuna de defectos, aumento de productividad, conservar la calidad, mejoras en el ambiente de trabajo, satisfacción del cliente.</i> |

3. Modelos de desarrollo de software: Scrum y PSP

3.1. Justificación

Como se puede observar en el apartado anterior, la mayoría de las empresas utilizan CMMI como modelo de madurez y Scrum como método para la administración ágil del proyecto. El problema que enfrenta las MiPyMEs para implementar CMMI es la carencia de un proceso y la gran cantidad de áreas que define el modelo de madurez; por otro lado, Scrum es perfecto para cualquier tipo de empresas, en especial las MiPyMEs, para agilizar el proceso de desarrollo sin sobrepasar los tiempos y recursos estipulados por los clientes sin olvidar la gestión y la calidad del producto. Scrum ha sido evaluado como un “muy buena” práctica

para proyectos de tamaño pequeño y mediano con un equipo de trabajo relativamente pequeño (3 a 9) [9,10]. Por otro lado PSP, define prácticas sencillas y ligeras, considera una “excelente” práctica para proyectos de todo tipo de tamaño aplicable a nivel personal (1 a 3) [10] que favorecen la institucionalización de procesos para este tipo de empresas. Scrum y PSP comparten los equipos pequeños, multidisciplinarios y auto-dirigidos para producir software; aquellos que utilizan Scrum usan las buenas prácticas para construir un producto adecuado y PSP ayuda a construir el producto correctamente.

3.2. Scrum

Desarrollador por Ken Schwaber and Jeff Sutherland basado en el concepto de que el desarrollo de software no es un proceso definido, sino un proceso empírico con transformaciones de entrada y salida complejas que pueden o no repetirse en circunstancias diferentes [9]. Es definido como un marco de trabajo por el cual las personas pueden trabajar con problemas complejos adaptativos, al mismo tiempo se entregan productos de gran valor posible [11]. Scrum implementa un ciclo de vida iterativo e incremental, formado por tres fases *Pre-juego*, *Juego* y *Post-juego* el cual involucra tres roles en particular: *Product Owner* (Dueño del Producto), el *Equipo de Desarrollo* y el *Scrum Master* [11, 12] mostrados en la Figura 1.

Scrum hace énfasis en la administración del proceso del desarrollo dirigido por el *Scrum Master* [2], la base de la planeación comienza con la generación de la lista de requisitos (*Product Backlog*) que contiene las mejoras funcionales y tecnológicas para el proyecto. Existe un ciclo de desarrollo correspondiente a 30 días, llamado *Sprint*, el cual es precedido por las actividades de las fases *Pre-juego* y *Post-juego*. Adicionalmente, el objetivo del *Sprint* es establecido, el cual sirve como un criterio mínimo para el éxito del mismo y mantiene al equipo enfocado en panorama global y no estrictamente en tareas específicas.

Se definen tres reuniones para el *Sprint*, la primera es una junta (30 minutos) que permite al equipo monitorear el estatus y los problemas de comunicación del proyecto, la segunda se realiza para mostrar el avance al *Product Owner* para obtener una observaciones, dudas o cambios sobre el producto, la tercera es una retroalimentación sobre lo bueno y lo malo de proceso utilizado en el anterior ciclo y se propone actividades de mejora.

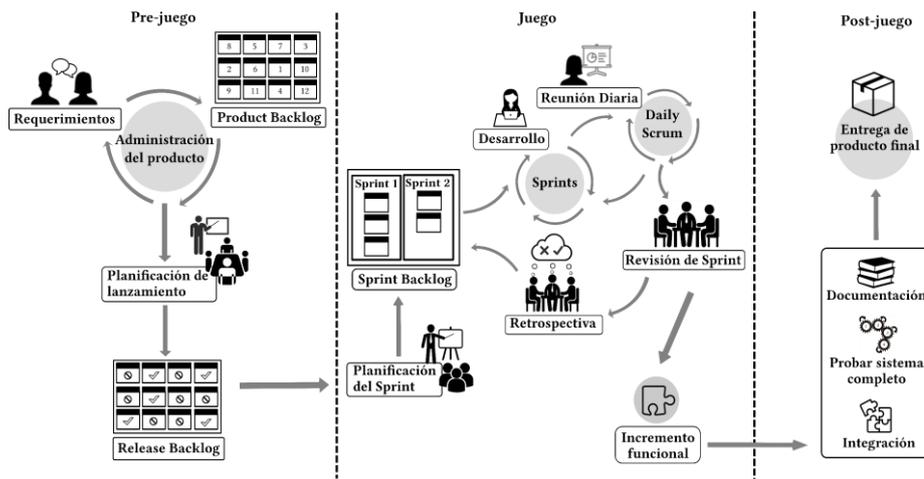


Fig. 1. Ciclo de vida de Scrum.

3.3. Personal Software Process (PSP)

Con base a las experiencias y los datos de cursos iniciales, Watts S. Humphrey escribió y publicó el primer libro sobre *Personal Software Process* (PSP) a finales de 1994. PSP es definido con un marco de trabajo para mejorar los procesos de construcción de software que funciona a nivel personal, su estrategia permite mejorar el rendimiento de los programadores utilizando prácticas sólidas para monitorizar y esforzarse en mejorar el desempeño, y la calidad de los productos. PSP se basa en los siguientes principios [13]:

1. Para una mejorar continua del desempeño, los desarrolladores definen y miden correctamente sus procesos.
2. Para producir productos de calidad, los desarrolladores deben sentirse responsables de la calidad de su trabajo.
3. El costo es menor si se encuentran y reparan defectos en fases tempranas que en fases próximas.
4. Es más eficiente prevenir los defectos que buscarlos y repararlos.
5. El camino correcto el siempre el más rápido y barato de trabajar.

PSP define siete niveles de madurez como se muestra en la Figura 2. Los niveles PSP0 y PSP0.1 son críticos para aprender la disciplina del proceso, considerados punto de partida para la secuencia del proceso. Cada nivel de PSP se divide en procesos descritos por medio de “*scripts*” los cuales no solo especifican los pasos a seguir, también los criterios de entrada y salida.

4. PSP y Scrum: integrando dos enfoques en el desarrollo de software para MiPyMEs

4.1. Análisis de características

En la Tabla 1 se muestra un análisis realizado con el propósito de determinar las características de integración para ambos enfoques considerando factores como: enfoque, tipo de conocimiento, prácticas para el desarrollo, estructura/fases, ambiente de trabajo, tipo de proyecto y cultura.

Tabla 2. Análisis entre factores para Scrum y PSP.

| <i>Factores</i> | <i>Scrum</i> | <i>PSP</i> |
|-----------------------------------|---|---|
| Enfoque | Agilidad, adaptación a cambios e inspecciones de trabajo. Producción rápida. | Mejorar las habilidades personales del ingeniero de software. Estimación basada en datos históricos. Predictibilidad estadística. |
| Tipo de conocimiento | Empírico, Tácito. | Teórico basado en registros. |
| Prácticas en el desarrollo | Gestión del proceso centrado en los requerimientos del cliente. Priorización de los requerimientos y estimación del tiempo. Descomposición de tareas. | Establece un flujo de trabajo para el ingeniero de manera personal. Define guías (<i>scripts</i>) para la administración del proceso. |

| Factores | Scrum | PSP |
|----------------------------|--|--|
| Estructura/Fases | Iterativo e incremental. Define tres fases: <i>Pre-juego</i> , <i>Juego</i> y <i>Post-juego</i> . | Iterativo y escalonado por niveles. Define tres principales procesos: <i>Planificación</i> , <i>Desarrollo</i> y <i>Postmortem</i> . |
| Ambiente de trabajo | Cambiante, inquieto, rápido, enfocado al proyecto. | Estable, pocas modificaciones, enfocado a los procesos. |
| Tipo de proyecto | Para proyectos cortos, medianos. Equipo de trabajo pequeño, poca necesidad de documentación. | Para cualquier tipo de proyectos, se enfoca en el trabajo del ingeniero. Gran cantidad de datos se registran en formularios. |
| Cultura | Trabajo y colaboración en equipo. Permite una mejora continua para todos los integrantes del equipo. | Establece la disciplina y el respeto al proceso de trabajo. Establece una mejora continua de las habilidades personales del desarrollador. |

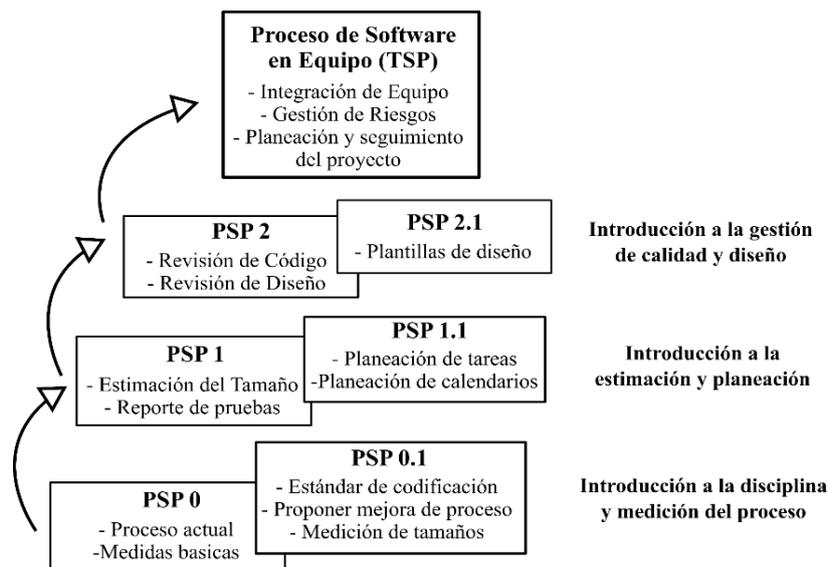


Fig. 2. Niveles de Madurez de PSP (Fuente: PSP: A Self-Improvement Process for Software Engineers, p. 8).

Una desventaja encontrada en Scrum es la carencia de guías para el proceso de desarrollo dirigidas al ingeniero de software en la construcción del incremento en los *Sprints*, PSP ayuda a eliminar esta brecha proporcionando prácticas sencillas para la construcción del incremento siguiendo los requisitos del cliente. Gracias a la recolección de datos, por parte de PSP, se generan registros donde la información se utiliza para predecir un mejor tiempo de trabajo y genera un plan de mejora individual.

La esencia de Scrum es: la agilidad, flexibilidad a cambios, estimación de tiempo de actividades y un proceso iterativo e incremental. Estas características son consideradas parte fundamental del esqueleto de la integración.

4.2. Modelo de integración: Scrum y PSP trabajando juntos

El siguiente modelo es una propuesta para integración de Scrum y PSP enfocada hacia las MiPyME's donde los equipos de desarrollo son pequeños. El modelo de integración entre Scrum y PSP está formado por dos capas. Antes de comenzar el desarrollo de un proyecto, se debe tener en cuenta las siguientes actividades preliminares:

- *Identificar la oportunidad apropiada* para el desarrollo de un sistema de software, que representa la comprensión compartida de las necesidades de los involucrados.
- *Conocer el tipo de proyecto* que se llevará a cabo, desarrollo o mantenimiento de software.
- *Reconocer a todos los posibles grupos diferentes de involucrados* que son, o serán afectados por el desarrollo y operación del sistema de software, junto con las responsabilidades de cada uno.

Ciclo de vida. Esta capa está definida por las tres fases: *Preparación*, *Desarrollo* y *Entrega* compaginados con las fases de *Pre-game*, *Game* y *Post-game* de Scrum respectivamente. El cambio en los nombres se realizó para una rápida identificación por parte del personal que lo implementan y sea más fácil su flujo de procesos. En la Figura 3 se muestra el ciclo de vida de la *Integración Scrum-PSP*.

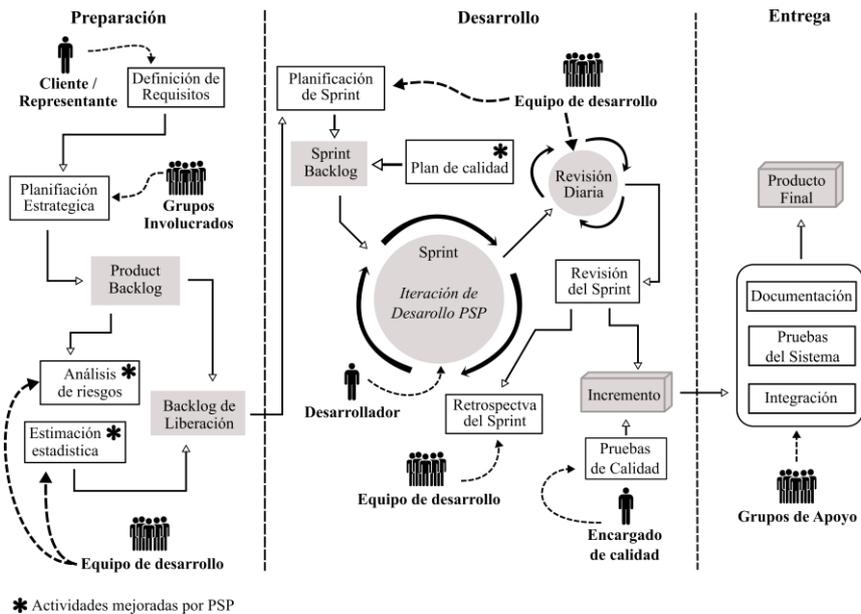


Fig. 3. Ciclo de vida del modelo de integración Scrum-PSP.

La fase de *Preparación* abarca desde la definición de las necesidades del cliente, que posteriormente se convertirán en *Product Backlog*, hasta la creación del *Backlog de liberación*. El proceso de captura de requisitos no exige herramientas y/o formatos para documentar, solo aconseja que los requerimientos sean lo más claros posibles, sin ambigüedades y accesibles por todos los grupos involucrados respetando su perfil de seguridad. La planificación estratégica se realiza respetando la lista de requerimientos y,

conforme los requisitos son priorizados y estimados por el equipo formando el *Backlog de Liberación*. Durante esta misma actividad se realiza el diseño de la “Arquitectura de Alto Nivel” que ayude al equipo como abordar los requerimientos. El “Análisis de Riesgos” será reforzado por las habilidades que PSP desarrolle en los integrantes y las estimaciones estarán apoyadas por el método estadístico PROBE, el cual tiene buena precisión para calcular el costo de las actividades en esfuerzo y tiempo, según los informes del *Software Engineering Institute (SEI)*.

La fase de *Desarrollo*: El equipo de desarrollo tiene que definir el número de *Sprints* que se utilizarán para desarrollar el incremento y tiene que crear el *Sprint Backlog* para cada iteración.

Un “Plan de Calidad” tiene que ser generado por parte del equipo y respetado hasta el final del desarrollo del incremento; en un principio este plan se realizará de manera abstracta y conforme los ingenieros adopten la disciplina de PSP dicho plan será cada vez más detallado. Además, se integran las prácticas de PSP basadas en las tres principales fases: “*Planeación, Desarrollo y Postmortem*”. El *Scrum Daily* continúa siendo una actividad sin cambios proveniente de Scrum.

La fase de *Entrega*: Establece la integración del nuevo incremento con el incremento generado en la anterior iteración. Se realizan pruebas al sistema completo para verificar su funcionamiento en diferentes entornos. Se realizan las configuraciones necesarias para que el nuevo sistema funcione los equipos de cómputo del cliente. Un punto importante de esta fase es que la documentación del proyecto tiene que estar terminada y verificada por las personas involucradas.

Iteración de desarrollo PSP. Es el proceso en el cual cada ingeniero de software utiliza las actividades clásicas para construir los programas o módulos correspondientes a sus actividades asignadas y está ubicada dentro del *Sprint*. La Figura 4 muestra la estructura del flujo de trabajo para PSP la cual consiste en 6 procesos.

- *Planificación personal*: Se produce un plan detallado para trabajar el desarrollo del programa definido por los requisitos del problema, los formatos para escribir el plan de trabajo no son difíciles pero requieren toda la atención del desarrollador. El plan consiste en la obtención y definición de los requerimientos para el programa escritos en documento claramente y sin ambigüedades.
- *Diseño Detallado*: Se realiza un diseño detallado para las especificaciones del programa definido por los requerimientos, las herramientas utilizadas es responsabilidad del desarrollador.
- *Código*: La transformación del diseño a sentencias de lenguajes de programación.
- *Compilación*: Se traducen las sentencias del lenguaje de programación a código ejecutable. La mayoría de los defectos de sintaxis serán removidos durante esta fase. Esta fase es “*opcional*”, determinada por el entorno de desarrollo y el lenguaje de programación.
- *Pruebas Unitarias*: Cada desarrollador realiza pruebas unitarias al programa o módulo para verificar que cumpla con los requerimientos, no se establece un número límite para las pruebas o herramientas para realizarlas, sin embargo se tienen que registrar el tipo de cada prueba realizada.
- *Postmortem*: Considerada como una retrospectiva personal para resumir y analizar los datos generados por el proceso. Estos datos incluyen valores sobre el tiempo estimado y

el tiempo real utilizado, calidad y productividad. Además, la información proporcionada por los productos personales permite dar bases a las retrospectivas del *Sprint*.

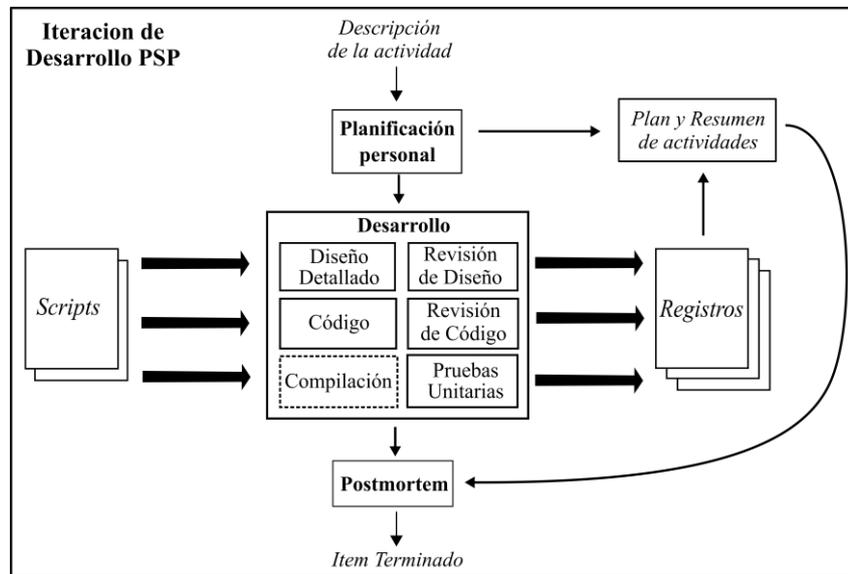


Fig. 4. Iteración de desarrollo PSP.

5. Conclusiones y trabajo a futuro

En esta investigación se ha propuesto un modelo de integración entre Scrum y PSP preservando la esencia de ambos enfoques para mejorar el trabajo individual del equipo, introduciendo prácticas que fomentan la disciplina del proceso y la institucionalización de los procesos de una empresa. Se espera que las prácticas de PSP ayuden a Scrum en generar estimaciones de tiempo más exactas, gracias al registro de datos y defectos. Un análisis de riesgos más preciso gracias a la disciplina creada por el proceso de PSP. Reducir el tiempo invertido en la etapa de pruebas gracias a las revisiones de diseño y código.

Como trabajo futuro se tiene la realización de casos de estudios en diferentes ambientes, un equipo con integrantes con experiencia en Scrum y otro integrado por estudiantes para conocer los puntos positivos y negativos de la propuesta. Comparar los resultados de individuos que utilizan PSP con aquellos que usan otros procesos para verificar si existen mejoras en la calidad del producto y calidad del proceso. Establecer una extensión o modificación al modelo propuesto para aquellas empresas o equipos que se encargan exclusivamente del mantenimiento del software. Enriquecer el modelo con la adición de normas de calidad como: *ISO/IEC*, niveles de *CMMI-DEV*, prácticas de *MoProSoft* o *CompetiSoft*. Enriquecer la dinámica del trabajo en equipo con prácticas de “*Team Software Process*” (TSP).

Agradecimientos. Esta investigación fue apoyada por El Consejo Nacional de Ciencia y Tecnología (CONACYT) bajo el Programa Nacional de Posgrados de Calidad (PNPC) y el Instituto Tecnológico de Orizaba (ITO) División de Estudios de Posgrado e Investigación (DEPI).

Referencias

1. Nielsen, P.: Struggles at the Frontier of Software Engineering. In: TSP Symposium, Going beyond agile, D.F., México (2016)
2. Boehm, B., Turner, R.: Balancing agility and discipline. Addison-Wesley, Boston (2004)
3. Lina, Z., Dan, S.: Research on Combining Scrum with CMMI in Small and Medium Organizations. In: International Conference on Computer Science and Electronics Engineering (2012)
4. Bougroun, Z., Zeaaraoui, A., Bouchentouf, T.: The projection of the specific practices of the third level of CMMI model in agile methods: Scrum, XP and Kanban. In: Third IEEE International Colloquium in Information Science and Technology (CIST) (2014)
5. Brown, D. et al.: PSP Implementations for agile methods: a SEMAT-based approach. *Software Engineering: Methods, Modeling, and Teaching*, 3, pp. 41–45 (2014)
6. Romano, B., Silva, A.: Project Management Using the Scrum Agile Method: A Case Study within a Small Enterprise. In: 12th International Conference on Information Technology - New Generations (2015)
7. Soares, F., de Lemos Meira, S.: An agile strategy for implementing CMMI project management practices in software organizations. In: 10th Iberian Conference on Information Systems and Technologies (CISTI) (2015)
8. Arauz, Ortiz, G., Morales Trujillo, M., Oktaba, H., Ramirez Hernandez, E.: Integrating Agile Methods into a Level 5 CMMI-DEV Organization: a Case Study. *IEEE Latin America Transactions*, 14, pp. 1440–1446 (2016)
9. Sutherland, J., Schwaber, K.: *The Definitive Guide to Scrum: The Rules of the Game*. Scrum. Org and ScrumInc (2014)
10. Jones, C.: *Software engineering best practices*. McGraw-Hill, New York (2010)
11. Palacio, J., Ruata, C.: *Scrum Manager Gestión de Proyectos*. 4th ed. SafeCreative, pp.57–87 (2010)
12. Schwaber, K.: *SCRUM Development Process*. Business Object Design and Implementation. pp. 117–134 (1997)
13. Humphrey, W.: *PSP: A Self-Improvement Process for Software Engineers*. Upper Saddle River, NJ: Addison-Wesley (2005)

Predicción de oxígeno disuelto en acuicultura semi-intensiva con redes neuronales artificiales

Carlos Julián Torres González¹, Víctor Manuel Landassuri-Moreno¹,
José Juan Carbajal Hernández², José Martín Flores Albino¹

¹ Universidad Autónoma del Estado de México,
Centro Universitario del Valle de México, Estado de México,
México

² Instituto Politécnico Nacional, Centro de Investigación en Computación,
Ciudad de México, México

ctorresg167@alumno.uaemex.mx, vmlandassurim@uaemex.mx, jcarbajalh@cic.ipn.mx,
jose.martin.flores.albino@gmail.com

Resumen. Actualmente el cultivo de especies acuícolas para consumo humano es una práctica que se realiza en todo el mundo. De ahí, el oxígeno disuelto es uno de los parámetros más importantes de la calidad del agua en sistemas acuícolas, por lo que mantener buenos niveles de concentración de este gas es esencial para una producción exitosa. Este trabajo presenta un modelo predictivo basado en Redes Neuronales Artificiales (RNAs), las cuales son diseñadas con el algoritmo evolutivo FS-EPNet para estimar la calidad del agua en base a la cantidad de oxígeno disuelto en un cultivo de camarón blanco. Los resultados obtenidos muestran un buen desempeño por parte de las RNAs evolucionadas, por lo que se vuelve una herramienta adecuada en el manejo del oxígeno disuelto y de la calidad del agua.

Palabras clave: Acuicultura, evolución de redes neuronales artificiales, camarón, predicción, tarjeta de adquisición de datos.

Dissolved Oxygen Prediction in Semi-intensive Aquaculture with Artificial Neural Networks

Abstract. Currently, farming of aquatic species for human consumption is a worldwide practice performed. Hence, dissolved oxygen is one of the most important parameters of water quality in aquaculture systems, so maintaining good levels of concentration of this gas is essential for a successful production. This work presents a predictive model based on Artificial Neural Networks (ANNs), which are designed with the FS-EPNet evolutionary algorithm to estimate water quality based on the amount of dissolved oxygen in white shrimp farming. The results show a good performance by the evolved RNAs, so

it becomes a suitable tool in the management of dissolved oxygen and water quality.

Keywords: Aquaculture, evolutionary artificial neural networks, shrimp, prediction, data acquisition card.

1. Introducción

Uno de los factores que más influyen en la producción de los sistemas acuícolas es la calidad del agua del cultivo en cuestión, de modo que su manejo apropiado juega un papel fundamental en el éxito de las operaciones acuícolas [7, 19].

De esta forma, uno de los parámetros de mayor importancia en la calidad del agua es el oxígeno disuelto (OD), mismo que determinará el estado general de la producción acuícola ya que tiene una influencia directa en la ingesta de alimento, la resistencia a enfermedades y el metabolismo, en consecuencia se vuelve crítico mantener valores adecuados de este gas disuelto en el agua [1, 18, 19]. Existen varios factores que modifican las concentraciones de OD, un ejemplo de ello es el ciclo dinámico del oxígeno en estanques, el cual fluctúa en el transcurso del día principalmente debido a la fotosíntesis y respiración del fitoplancton y camarones [1].

En contraparte, la solubilidad del oxígeno se ve afectada en gran medida por la temperatura y la salinidad del agua [1, 4]. Factores secundarios como el pH, amonio, demanda bioquímica y biológica de oxígeno, entre otros, también tienen relación en las concentraciones de este gas [1, 2, 9].

Cabe destacar que durante la noche y en la madrugada los niveles de oxígeno disminuyen de forma natural, pero dependiendo de que tan bajos estén los niveles de éste gas, ya sea durante el día o la noche, puede provocar que los organismos se alimenten menos, presenten una pobre asimilación de alimento, crecimiento más lento, ser más susceptibles a enfermedades infecciosas, estresarse e incluso presentar muertes masivas [1, 7, 12]. Así, existen normas y criterios de cultivo de especies acuícolas las cuales son publicadas por diversos organismos de sanidad: la Secretaría de Agricultura, Ganadería, Desarrollo Rural, Pesca y Alimentación [23], la Canadian Council Ministre of the Environment [6] y la National Sanitation Foundation [22] entre otros.

Dada la importancia del oxígeno disuelto, el presente trabajo propone la creación de un modelo computacional para la predicción del éste gas empleando Programación Evolutivo de Redes Neuronales (EPNet [25]), a través del algoritmo evolutivo FS-EPNet [17], permitiendo así conocer el deterioro de la calidad del agua en un cultivo de camarón blanco. Mediante un conjunto de muestras obtenidas de estanques en funcionamiento, se entrena la red propuesta y se verifica su rendimiento predictivo.

El resto de este trabajo está organizado de la siguiente forma: la Sección 2 muestra la relación del oxígeno disuelto y la calidad del agua, así como se describe la forma en la que se recolectaron los datos de esta investigación. La sección 3 presenta a grandes rasgos el algoritmo evolutivos FS-EPNet y la configuración experimental del trabajo. Los Resultados se pueden encontrar en la Sección 4 y por último las conclusiones en la Sección 5.

2. Oxígeno disuelto para una buena calidad del agua

La acuicultura es definida como el cultivo de organismos acuáticos como peces, moluscos, crustáceos entre otros. El hábitat es controlado con el fin de aumentar la producción, simulando las condiciones generales que dichas especies tienen en su ambiente silvestre [2]. Las concentraciones de oxígeno disuelto son un factor preponderante en la producción de camarón blanco debido a la importancia que tiene en su desarrollo y en la resistencia a enfermedades [24]. Uno de los principales parámetros que influye en la cantidad de oxígeno contenido en el agua es la temperatura; si ésta aumenta, la tasa metabólica será mayor [12] generando una mayor demanda de oxígeno [2, 4]. Asimismo, la solubilidad del oxígeno disminuye a medida que aumentan las sales disueltas [1]. La mayor cantidad de oxígeno en el agua en estanques de cultivo proviene del proceso de fotosíntesis realizado por las plantas y algas microscópicas. Este proceso depende de la cantidad de luz que pueden disponer los vegetales. Debido a ello, los niveles de OD fluctúan durante el día, elevándose a lo largo de la mañana y alcanzando su punto máximo por la tarde [4, 19]. Los niveles aceptables de OD deben ser mayores a 6 mg/l, ya que con niveles inferiores se puede presentar lento crecimiento, llegando incluso a condiciones de hipoxia debajo de los 2 mg/l donde en periodos prolongados es letal para los organismos.

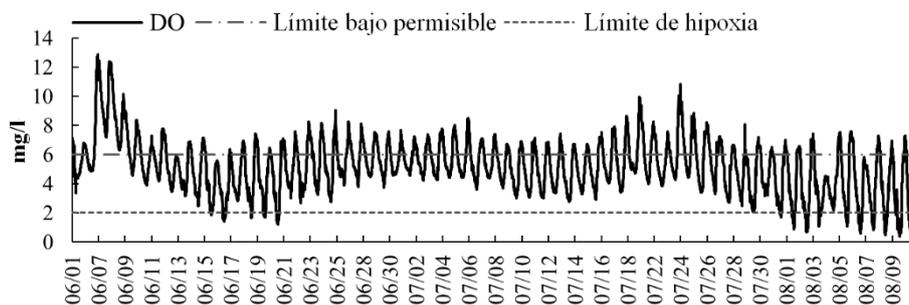


Fig. 1. Comportamiento diario del oxígeno durante un periodo de cultivo.

2.1. Recolección de muestras

Con el fin de crear una base de datos para probar el modelo el oxígeno disuelto se midió éste mediante un sensor en un estanque de pruebas de la finca Rancho Chapo ubicada en Sonora, México. El período de seguimiento de las muestras fue de 15 minutos a lo largo de tres meses de mediciones; esto es, el conjunto de datos está formado por 8736 valores (junio, julio y agosto de 2007). Sin embargo, las fallas en el sensor generaron registros falsos, por lo que fueron borrados de la base de datos de prueba, quedando 5952 registros para este trabajo. El comportamiento de estos registros se puede observar en la Fig. 1, donde se aprecia que los límites permisibles son sobrepasados, generando malas condiciones de calidad del agua. Es importante

remarcar que el oxígeno disuelto presenta concentraciones de hipoxia, lo que es un problema extremadamente peligroso en cualquier tipo de sistema de cultivo acuícola.

3. Evolución de redes neuronales artificiales

En este trabajo, las Redes Neuronales Artificiales (RNAs) se utilizan para evaluar los niveles de riesgo de oxígeno en granjas extensivas de camarón debido a que han demostrado ser un modelo de aprendizaje altamente eficiente. Sin embargo, la construcción de una RNA no es una tarea trivial, siendo en si un problema combinatorio el cual es considerado dentro de los problema NP-Completo, por todas las variables que se tienen que ajustar. Por otro lado, los factores mencionados en la Sección 2 afectan directa o indirectamente las concentraciones de oxígeno disuelto ocasionando un comportamiento no lineal del sistema, donde dicha dinámica puede ser procesada por RNAs encontrando relaciones no lineales y permitiendo estimaciones útiles de éste gas. Así, los algoritmos evolutivos permiten adaptar durante evolución parámetros de las RNAs, como pueden ser el número de entradas, el número de nodos ocultos y su conectividad entre ellos. De esta forma, este trabajo utiliza el algoritmo evolutivo FS-EPNet [17], el cual es una mejora del EPNet [25] propuesto originalmente, donde este último no permite la evolución de las entradas, y éstas tienen que ser ajustadas por el experto. Así, el algoritmo FS-EPNet permite adaptar automáticamente las entradas de la red, ajustándolas automáticamente a través del proceso evolutivo, en donde se cuenta con una población de individuos de RNAs las cuales compiten y evolucionan para mejorar la predicción del OD. A través del proceso de adaptación de nodos y conexiones, es posible encontrar más de una capa oculta, dado el modelo de Generalized Multi-layered perceptron (Generalized MLP-GMLP) el cual elimina la restricción de tener capas adyacentes conectadas, a diferencia del modelo convencional (MLP).

El entrenamiento es llevado a cabo con el algoritmo de Backpropagation modificado (MBP), el cual adapta la tasa de aprendizaje de generación en generación, así posterior a una mutación (eliminar o agregar nodos o conexiones), disminuye o incrementa esta variable de acuerdo al rendimiento de la red. Este algoritmo está diseñado para dar prioridad a RNAs pequeñas. Así, primero da prioridad a las mutaciones de eliminación antes de las mutaciones de incremento o agregación de elementos. Es decir, en cada generación es seleccionada una RNA para ser mutada, donde primero se entrena parcialmente para ver si el error puede ser reducido como se comentó anteriormente (MBP). De no ser el caso, se crean nuevos individuos a partir de él para eliminar entradas, nodos ocultos o conexiones, si alguna de estas mutaciones tiene éxito, lo cual significa que el error se reduce, el nuevo individuo es insertado en el lugar de la red seleccionada. Si ninguna mutación de eliminación funciona, el algoritmo intenta agregar elementos de entradas, nodos ocultos y conexiones, quedándose con la red que mejor rendimiento otorgue (para predicción en este caso), lo que significa que reemplazará a la red escogida inicialmente. Por último, el "bias" es un parámetro que puede ser evolucionado. Por cuestiones de espacio, no

se da una descripción más detallada del algoritmo ni se presenta su diagrama de flujo, sin embargo para mayor detalle se puede referir a [17].

3.1. Configuración experimental

Derivado de diversos experimentos previos, se determinaron los siguientes parámetros usados en la configuración experimental: de las 5296 mediciones disponibles, se tomaron los primeros 4200 valores, donde los primeros 3000 (56% del total) son usados para el diseño y entrenamiento de las redes con el algoritmo evolutivo, y los siguientes 1200 (22% del total) fueron usados para probar las redes resultantes del proceso evolutivo, y seleccionar la mejor red neuronal que resuelva el problema, es decir, al término de la evolución todas las redes encontradas (población de individuos) fueron probadas con este conjunto de prueba de 1200 patrones. En cuanto a las generaciones del algoritmo evolutivo se encontró que 4000 generaciones eran adecuadas para lograr reducir considerablemente el error en comparación con la primera generación (primera población de RNAs), donde más generaciones no representaba un aumento significativo en el rendimiento, pero sí en el tiempo de ejecución. Se utiliza un modelo de predicciones iteradas MSP (Multiple Step Predicción), que consiste en predecir nuevos valores sobre los valores anteriormente predichos. Por otro lado, la determinación del tipo de funciones de transferencia utilizadas en el proceso de construcción de la RNA es importante, debido a que definen las características de salida de todo el modelo neuronal. En este caso, se utilizó una función de transferencia lineal en la capa de salida, $f(x) = x$, y tangente hiperbólica en las capas intermedias, $f(x) = \tanh(ax)$. En cuanto al pre-procesamiento de los datos, el entrenamiento de redes neuronales puede ser más eficiente si eliminamos tendencias, por ejemplo, los parámetros de OD tienen diferentes rangos de valores como se mostró en la Fig. 1. Donde no es conveniente usarlos de esta forma, ya que valores más grandes tienen un mayor impacto y provocan una tendencia en el resultado final [13, 16]. Así, en primer lugar, una estandarización transforma los datos para lograr una media de cero, con la unidad de desviación estándar como sigue:

$$X^k = \frac{X_i - \bar{X}}{\sigma} \quad (1)$$

Después, los parámetros estandarizados fueron normalizados en un rango de [-0.9, 0.9] de acuerdo a la siguiente ecuación:

$$X^\omega = (N_{\max} - N_{\min}) \frac{X_i^k - \min(X^k)}{\max(X^k) - \min(X^k)} + N_{\min} \quad (2)$$

En donde N_{\max} y N_{\min} son los valores máximo y mínimo del rango deseado y X^ω es el valor normalizado con el cual se construirán los patrones de entrenamiento y prueba posteriores. Cabe resaltar que esta normalización tiene efecto a fin de satisfacer los rangos de las funciones de transferencia usadas. La métrica de errores

usada para determinar la precisión en la predicción de las RNAs fue el NRMSE (Normalized Root Mean Square Error).

4. Resultados y discusión

La Fig. 2 muestra la mejor RNA encontrada por el algoritmo evolutivo para la configuración experimental anteriormente descrita. Esta red presenta una topología (6, 6, 2, 3, 1), esto es: 6 nodos de entrada, 6 en la primera capa oculta, 2 en la segunda capa oculta, 3 en la tercera capa oculta y 1 en la capa de salida. Donde es bueno recordar que es una arquitectura obtenida con el algoritmo FS-EPNet usando un modelo GMLP, y las 6 entradas (nodos 1 al 6) de esta red corresponden a 6 valores anteriores al valor a predecir del OD, es decir los nodos del 1 al 6 toman valores pasados para predecir el siguiente. En su contraparte, los nodos del 7 al 17 son ocultos repartidos en 3 capas ocultas y el nodo 18 provee la salida de la RNA. Se indica con una “b” los nodos que tienen un bias asociado, para este caso, del nodo 7 al 18.

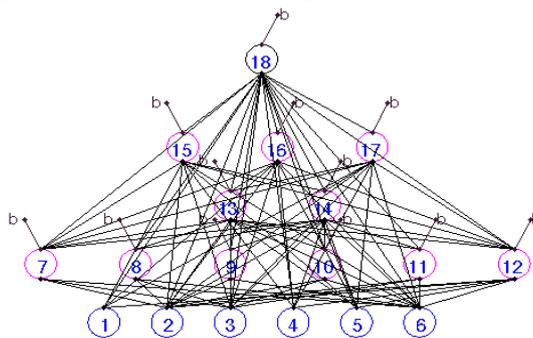


Fig. 2. Mejor RNA encontrada para la predicción de 1200 pasos adelante del OD en el cultivo de camarón blanco.

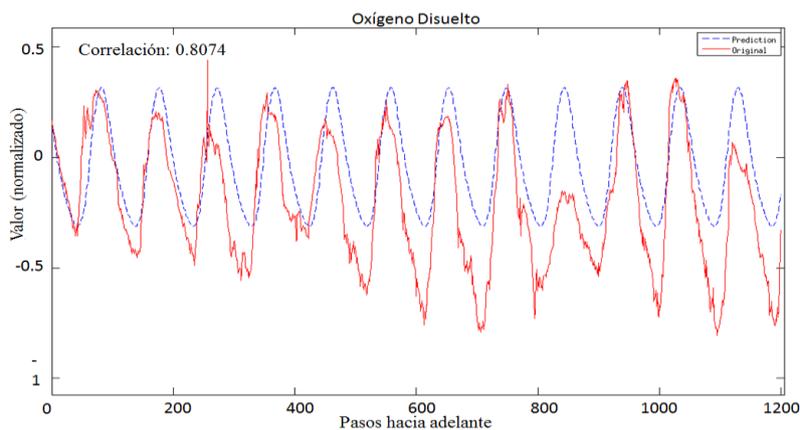


Fig. 3. Comparación de valores medidos de OD y predichos de la red neuronal.

El la Fig. 3 se presenta la predicción realizada con 1200 pasos hacia adelante, la cual muestra un buen acercamiento a los valores reales presentando un $NRMSE = 0.7956$. Asimismo, se muestra una respuesta aceptable y una alta correlación, sin embargo, la curva de predicción parece estar marcando la tendencia del comportamiento de los datos originales, esto es, posiblemente por la gran cantidad de datos de entrenamiento y los pasos adelante a predecir (1200).

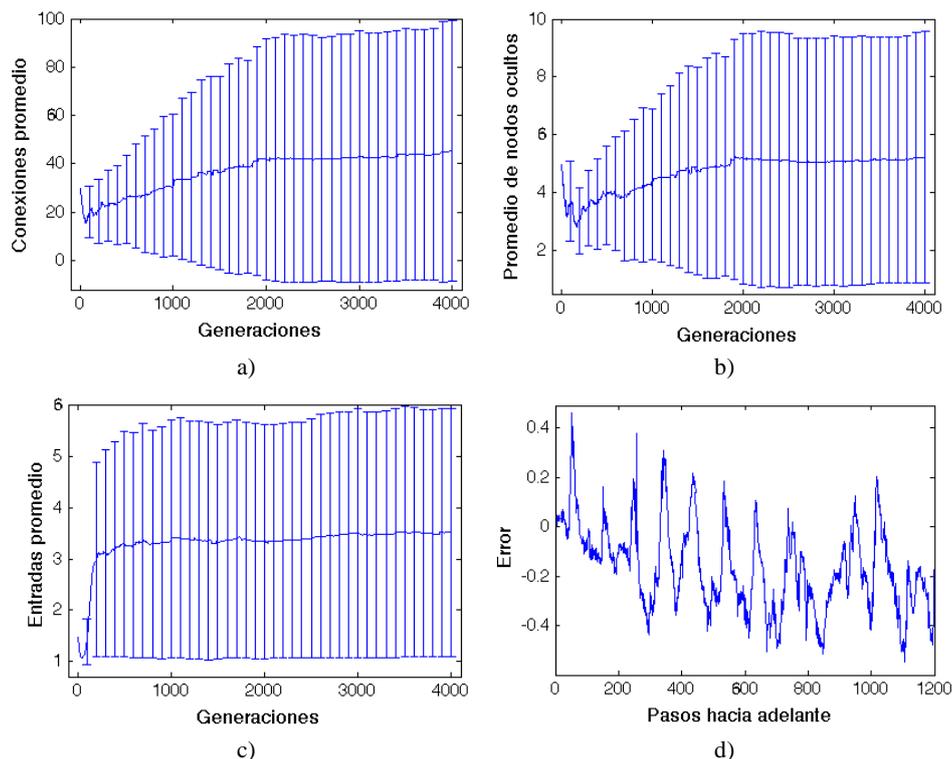


Fig. 4. Barras de Error (desviación estándar) durante evolución de: conexiones (a), nodos ocultos (b), entradas (c). La Figura (d) muestra el error entre las mediciones reales y los predichos por la red neuronal.

En su contraparte, la Fig. 4a – 4c, muestra el promedio de las conexiones (4a), nodos ocultos (4b) y entradas (4c) durante las 4000 generaciones de evolución. A medida que aumentan las generaciones hay un incremento en el número de esas 3 variables, al final la Fig. 4c alcanza 3 nodos de entrada en promedio sobre la población entera de individuos a evolucionar; sin embargo, la mejor RNA presenta 6 neuronas de entrada como se mostró anteriormente. Observando el comportamiento del error durante evolución (figura no mostrada aquí) y las Fig. 4a – 4c, se nota un incremento de todas las variables mientras que el error promedio durante evolución disminuye y no aumenta, lo cual es una ventaja de usar un algoritmo evolutivo como el usado en este trabajo, en lugar de diseñar las arquitecturas de RNAs por el experto humano a mano (Hand design Neural Networks). En la Fig. 4d se muestra el error de

la predicción y de los datos reales. Ahí se puede ver que a medida que van avanzando los pasos a predecir se va presentando más disparidad entre las predicciones y los valores reales, en donde se puede apreciar que el error es cercano a cero al inicio de las predicciones y a medida que se incrementan el horizonte de predicción el error va aumentado, esto es debido al modelo de predicción utilizado (MSP), el cual predice los siguientes valores usando valores predichos previamente.

Para eliminar la aparente tendencia en la predicción con 1200 pasos adelante se realizan pruebas con menos puntos a predecir. La mejor respuesta se obtiene con 21 pasos a predecir en dicho horizonte (Fig. 5), donde la gráfica de datos original corresponde a los últimos 21 datos (1179-1200) de la gráfica de datos original de la figura 3. Diferentes modelos se han desarrollado y utilizado para analizar el OD. Por ejemplo Stefan et al. [26] desarrollaron un modelo determinista, y Boano et al. [27] presenta un modelo estadístico para analizar la relación entre el DO y la demanda bioquímica de oxígeno (DBO), aunque debido a la complejidad y la no linealidad del comportamiento del OD, estos modelos no pueden predecir adecuadamente dichas variables, como los resultados obtenidos en esta sección. En otros estudios [2, 28] se utilizan RNA para realizar estimaciones y predicciones del OD, sin embargo, los métodos de entrenamiento utilizados tienen el inconveniente entre otras cosas, de una lenta convergencia y posibilidades de caer en mínimos locales.

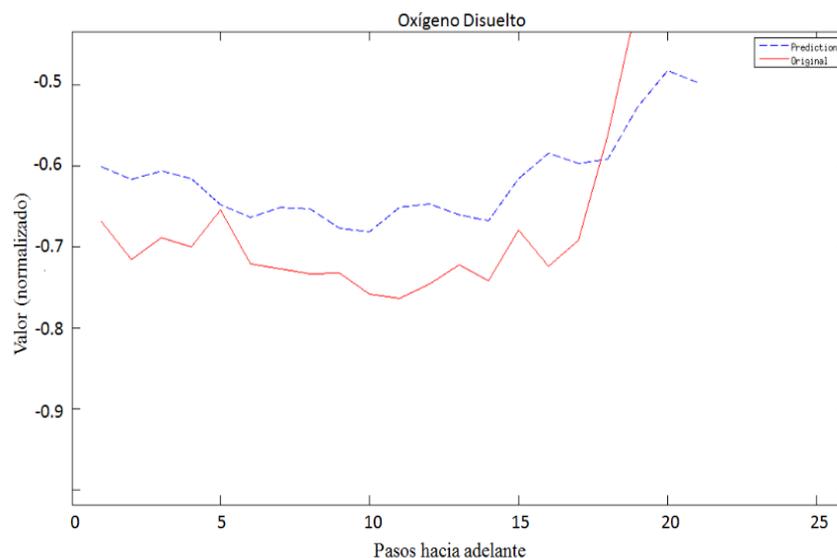


Fig. 5. Desviación estándar de la evolución del número de entradas.

Así, el uso del algoritmo FS-EPNet en el diseño de las RNAs permite obtener una arquitectura optimizada con un rendimiento aceptable, y dado el proceso evolutivo y el entrenamiento parcial con el MBP se evita caer en mínimos locales, todo esto sin la intervención de un experto. No es posible hacer una comparación directa con los trabajos citados anteriormente dado el origen diverso de los datos, sin embargo se espera que en trabajos futuros se puedan comparar.

5. Conclusiones

El estudio de la calidad del agua cobra gran relevancia para los cultivos acuícolas y por ello ha sido estudiada ampliamente. Diversos estudios se han desarrollado para ofrecer indicadores del nivel de la calidad del agua en ríos, saneamiento de agua potable y algunos en sistemas acuícolas. El modelo con RNA propuesto en este trabajo permite realizar una predicción adecuada como se mostró en la Fig. 3, donde dichas predicciones son útiles para tomar decisiones a futuro. En otro aspecto, al usar el algoritmo evolutivo FS-EPNet permite automatizar la creación de ellas (arquitecturas), reduciendo el trabajo y tiempo para encontrar la estructura de la red neuronal adecuada. Así las RNAs evolucionadas permiten abordar de una mejor manera el problema de no linealidad del OD y así anticipar con mayor precisión el estado de la calidad del agua del cultivo, dado que el proceso evolutivo permite ajustar de una mejor forma valores que serían difíciles de ajustar por un experto (Hand Design Neural Networks). Por lo anterior se concluye que es convierte el uso del FS-EPNet al diseñar ANNs como alternativa para controlar el estado de la calidad del agua.

Agradecimientos. Se agradece el tiempo de cómputo brindado por parte del Clúster del Centro Universitario UAEM Valle de México (CLU-CUUAEM-VM), de la Universidad Autónoma del Estado de México (UAEM), al Centro de Investigación en Computación (CIC) del Instituto Politécnico Nacional y al Centro de Investigaciones Biológicas del Noroeste (CIBNOR) para llevar acabo los experimentos y/o simulaciones de éste trabajo.

Referencias

1. La Oxigenación en la Acuicultura. AquaFeed Internacional, <http://aquafeed.co/la-oxigenacion-en-la-acuicultura/> Accedido el 26 de Octubre del 2015 (2012)
2. Areerachakul, S., Junsawang, P., Pomsathit, A.: Prediction of Dissolved Oxygen Using Artificial Neural Network. In: 2011 International Conference on Computer Communication and Management, 5, pp. 524–528 (2011)
3. Bodenhofer, U.: Genetic Algorithms: Theory and Applications. Johannes Kepler Universität Linz, 3 (2004)
4. Boyd, B., C., Handson, T.: Dissolved-Oxygen Concentrations in Pond Aquaculture. Global Aquaculture Advocate. January/February 2010, pp. 40–41 (2010)
5. Bremermann, H., J.: The evolution of intelligence. The nervous system as a model of its environment. Technical Report, Department of Mathematics, University of Washington (1858)
6. Canadian Council of Ministers of the Environment. An assessment of the application and testing of the water quality index of the Canadian Council of Ministers of the Environment for selected water bodies in Atlantic Canada, http://www.ccme.ca/files/Resources/water/water_quality/awi_en_posting.pdf. Accedido el 5 de Febrero del 2015 (2007)
7. Clear Blue Enviromental. <http://clrbu.com/industry/aquaculture-fish-farm-aeration/> . Accedido el 5 de Febrero del 2016

8. Doaa, M., A, Faten, H., F.: Design and control strategy of diffused air aeration system. *The Online Journal of Power and Energy Engineering* 3, 2, pp. 284–290 (2013)
9. Datta S.: Management of water quality in intensive aquaculture. *Researchgate*, pp. 1-17 (2012)
10. Fraser, A. S.: Simulation of genetic systems by automatic digital computers. II: Effects of linkage on rates under selection. *Austral. J. Biol.*, 10, pp. 492–499 (1957)
11. Freeman, J., Skapura, J.: *Redes neuronales. Algoritmos, aplicaciones y técnicas de programación.* <http://unilectura.blogspot.mx/2013/08/redes-neuronales-algoritmos.html> (1991)
12. Goyenola, G.: Oxígeno Disuelto, Junio. *RED MAPSA*, 1.0, 3 (2007)
13. Haykin, S.: *Neural networks. A comprehensive foundation.* Pearson, 2 (2005)
14. Holland, J., H.: *Adaptation in Natural and Artificial Systems.* University of Michigan Press, Ann Arbor, MI (1975)
15. Ibañez, J.: *Conceptos y tipos de modelos científicos.* Madrid <http://www.madrimas.org/blogs/universo/2008/05/10/91441>, Accedido el 20 de Julio del 2016 (2008)
16. Kriesel, D.: *Neural Networks. A brief introduction.* Dkdriese.com (2005)
17. Landassuri, V., M., Bustillo, C., L., Carbajal, J., J., Sánchez, L., P.: Single-Step-Ahead and Multi-Step-Ahead Prediction with Evolutionary Artificial Neural Networks. *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. Lecture Notes in Computer Science*, 8252, Springer, pp. 65–72 (2013)
18. Lee, G., P.: *Process control and artificial intelligence software for aquaculture.* *Aquaculture Engineering*, 23 (2000)
19. Lekang, O., I.: *Aquaculture Engineering.* Blackwell Publishing (2007)
20. Mendez, P., Ibarra, J.: Implementación de una red neuronal de convolución para el reconocimiento de poses en imágenes de rostros. *Avances en ciencias e ingeniería*, 6, 2, A1–P6 (2004)
21. Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs.* Springer, Heidelberg (1996)
22. NSF Internaional, <http://www.nsf.org> Accedido el 26 de Junio del 2015
23. SAGARPA. Normatividad, http://www.sagarpa.gob.mx/agricultura/Programas/proagro/Normatividad/Paginas/Normatividad_2014 Accedido el 24 de Junio del 2015 (2014)
24. Tucker, C.: *Pond Aeration.* Southern regional aquaculture center SRAC. 3700, 8 (2005)
25. Yao, X., Liu, Y.: A new evolutionary system for evolving artificial neural networks. *IEEE Transactions on Neural Networks* 8, 3, 694–713 (1997)
26. Stefan, H., G., Fang, X., Wright, D., Eaton, J., G., McCormick, J., H.: Simulation of dissolved oxygen profiles in a transparent, dimictic lake, *Limnology and Oceanography*, 4, pp. 105–118 (1995)
27. Boano, F., Revell, R., Ridolfi, L.: Stochastic modelling of DO and BOD components in a stream with random inputs. *Advances in Water Resources*, 29, pp. 1341–1350 (2006)
28. Faruk, D. O.: A hybrid neural network and ARIMA model for water quality time series prediction. *Engineering Applications of Artificial Intelligence*, pp. 586–594 (2010)

Filtro de restauración de imágenes basado en la transformada discreta del coseno y el análisis de componentes principales

Alejandro I. Callejas Ramos, Edgardo M. Felipe-Riverón,
Pablo Manrique Ramírez, Oleksiy Pogrebnyak

Instituto Politécnico Nacional, Centro de Investigación en Computación, Ciudad de México,
México

¹ivan_grin22@hotmail.com, ²edgardo@cic.ipn.mx, ³pmanriq@cic.ipn.mx,
⁴olek@cic.ipn.mx

Resumen. En este trabajo se propone un algoritmo de filtrado de imágenes contaminadas por el ruido aditivo Gaussiano. Los conjuntos de bloques a procesar se determinan por medio del algoritmo propuesto para la búsqueda de la similitud de complejidad reducida en el dominio de la transformada del coseno. Para tratar imágenes con ruido de varianza pequeña, se propone usar la etapa adicional del filtrado con base en el análisis de componentes principales, para una mejor preservación de los detalles de la imagen, y en el caso de tener ruido de alta intensidad, se realiza el filtrado de Wiener utilizando la estimación de la imagen resultante de la primera etapa. Los resultados obtenidos se comparan con los filtros del estado del arte en términos de la relación pico de señal/ruido e índice de similitud estructural.

Palabras clave: Restauración de imágenes, análisis de las componentes principales, transformada discreta del coseno, búsqueda rápida de bloques similares.

Filter for Image Restoration Based on Discrete Cosine Transform and Principal Component Analysis

Abstract. An algorithm for restoration of the images contaminated by additive white Gaussian noise is proposed. The groups of patches are found by the proposed block similarity search algorithm of reduced complexity performed on block patches in transform domain. When the noise variance is small, the proposed filter uses an additional stage based on principal component analysis; otherwise the experimental Wiener filtering is performed. The obtained filtering results are compared to the state of the art filters in terms of peak signal-to-noise ratio and structure similarity index.

Keywords: Image restoration, principal component analysis, block matching.

1. Introducción

El ruido es uno de los principales factores que afecta la calidad de las imágenes [1,2] y con frecuencia es el ruido aditivo Gaussiano (AWGN). Actualmente se conocen varios métodos de supresión de ruido en imágenes digitales [2-8], pero las investigaciones continúan para diseñar nuevas técnicas más eficientes. La razón es que los resultados en la supresión de ruido todavía no son completamente aceptables para el análisis posterior de las imágenes.

Las técnicas modernas del estado de arte para la supresión del ruido se pueden dividir en dos grupos: 1) filtros no locales [2] basados en la búsqueda de bloques similares y su procesamiento conjunto, tales como BM3D [3] y SA-DCT [4] y 2) aquellos basados en agrupamiento de las imágenes, kernels de regresión, descomposición en valores singulares o análisis de componentes principales para el aprendizaje de diccionarios y representación dispersa de imágenes [5-8].

En la familia de los filtros no locales se destaca el filtro BM3D [3] que demostró ser más eficiente para procesar la mayoría de las imágenes en escala de grises [5, 9] y las componentes de las imágenes en color [10] contaminadas por el ruido AWGN. Por otra parte, los filtros basados en la representación escasa muestran buenos resultados y en algunos casos son superiores a BM3D [7], pero este tipo de filtros tiene muy alta complejidad computacional causada por etapas de agrupamiento, aprendizaje de diccionarios, cálculos de diferentes rasgos locales y una búsqueda recurrida para la representación más dispersa.

Otro aspecto es la evaluación de la eficiencia en la supresión del ruido. Con frecuencia, se usa el error mínimo cuadrático en la forma de la relación pico de la señal al ruido (PSNR) para estimar la calidad de las imágenes filtradas. Desafortunadamente, con este criterio no siempre se obtienen buenos resultados en el sentido de la calidad visual de las imágenes, que sean más apropiados para su posterior análisis y reconocimiento [12]. Por otra parte, existen unos criterios de calidad de las imágenes relativamente nuevos, basados en las propiedades del sistema de visión humano (PSNR-HVSM) [13], en el índice de similitud de rasgos (FSIM) [14], en el índice de similitud estructural (SSIM) [15] o en el índice de similitud estructural multiescala (MSSIM) [16].

En este artículo tratamos de unir las ventajas de ambos grupos de filtros mencionados. La técnica de filtrado desarrollada utiliza la transformada discreta del coseno (DCT); se propone el algoritmo de búsqueda de bloques similares de imagen en el dominio de la transformada con complejidad computacional reducida; la transformada de Hadamard para el umbralización dura de grupos de bloques [11] en la primera etapa, y la transformada de Karhunen-Löve (KLT) con umbralización dura o filtrado de Wiener en la etapa final, dependiendo del nivel de ruido.

2. Filtrado de Wiener y umbralado en el dominio DCT

El modelo de observación para la señal de entrada es (Ec. 1):

$$u(x, y) = s(x, y) + n(x, y), \quad (1)$$

donde $u(x, y)$ es un pixel de la imagen de entrada ruidosa, x, y son sus coordenadas, $s(x, y)$ denota la imagen sin ruido, y $n(x, y)$ es el ruido aditivo blanco Gaussiano. El problema es encontrar una estimación $\hat{s}(x, y)$ de la imagen sin ruido tal, que minimice el error cuadrático promedio (MSE, *Mean Square Error*). El filtro óptimo lineal que minimiza el MSE es el filtro de Wiener, que en el dominio espectral puede ser formulado como [9] (Ec. 2):

$$H_W(\omega_x, \omega_y) = \frac{P_s(\omega_x, \omega_y)}{P_s(\omega_x, \omega_y) + P_n(\omega_x, \omega_y)}, \quad (2)$$

donde $P_s(\omega_x, \omega_y), P_n(\omega_x, \omega_y)$ son las densidades espectrales de potencia de la señal y del ruido, respectivamente. En la práctica, los espectros de potencia exactos $P_s(\omega_x, \omega_y), P_n(\omega_x, \omega_y)$ no están disponibles, por lo que se sustituyen por sus estimaciones. Con ello, el filtro de Wiener estimado (en el caso de AWGN) es (Ec. 3):

$$\hat{H}_W(\omega_x, \omega_y) = \frac{\hat{P}_s(\omega_x, \omega_y)}{\hat{P}_s(\omega_x, \omega_y) + \sigma^2} = \frac{P_u(\omega_x, \omega_y)}{P_u(\omega_x, \omega_y) + \sigma^2}. \quad (3)$$

Como referencia, tomamos el punto de banda de paso del filtro de ganancia de -3dB donde $\hat{H}_W^{-3dB}(\omega_x, \omega_y) = 1/\sqrt{2}$. En este punto, el espectro de la señal pura debe estar en función de la varianza del ruido σ^2 (Ec. 4):

$$\hat{P}_s^{-3dB}(\omega_x, \omega_y) = \frac{1}{\sqrt{2}-1} \sigma^2 \approx 2.414 \sigma^2. \quad (4)$$

Por otra parte, según el modelo de observación (Ec. 1), la densidad espectral de potencia de la señal ruidosa se puede expresar en función del punto de -3dB de ganancia del filtro de Wiener como (Ec. 5):

$$P_u(\omega_x, \omega_y) = \tilde{P}_s^{-3dB}(\omega_x, \omega_y) + 2\sigma \sqrt{\hat{P}_s^{-3dB}(\omega_x, \omega_y) + \sigma^2}, \quad (5)$$

donde $\tilde{P}_s^{-3dB}(\omega_x, \omega_y)$ denota la estimación de la densidad de potencia de la señal verdadera en el punto de -3dB que se puede derivar del periodograma $P_u(\omega_x, \omega_y)$. Usando $\hat{P}_s^{-3dB}(\omega_x, \omega_y)$ de la ecuación (4) en ecuación (5), se puede derivar la estimación de $\tilde{P}_s^{-3dB}(\omega_x, \omega_y)$ en términos de $P_u(\omega_x, \omega_y)$ (Ec. 6):

$$\tilde{P}_s^{-3dB}(\omega_x, \omega_y) = P_u(\omega_x, \omega_y) - 2\sigma^2 \left(\sqrt{1/(\sqrt{2}-1)} + 1 \right). \quad (6)$$

De las ecuaciones (4) y (6) es posible encontrar las condiciones para el corte de la respuesta en frecuencia del filtro de Wiener al nivel de -3dB en términos de la varianza del ruido (Ec. 7):

$$P_u(\omega_x, \omega_y) = \sigma^2 \frac{2 \left(\sqrt{1/(\sqrt{2}-1)} + 1 \right) (\sqrt{2}-1) + 1}{\sqrt{2}-1} \approx 6.5217615 \cdot \sigma^2. \quad (7)$$

Con ello, el filtro debe suprimir los coeficientes espectrales menores por su valor absoluto que $\sqrt{6.5217615} = 2.55377397$; la respuesta en frecuencia del filtro propuesto de umbralado es entonces (Ec. 8):

$$H_T(\omega_x, \omega_y) = \begin{cases} 1, & \text{si } |U(\omega_x, \omega_y)| \geq 2.55377397 \cdot \sigma \\ 0 & \text{en otro caso} \end{cases}, \quad (8)$$

donde $U(\omega_x, \omega_y)$ es el espectro de voltaje de la imagen observada $u(x, y)$, con valor $U(\omega_x, \omega_y) = F\{u(x, y)\}$, $F\{\cdot\}$ denota la transformada de Fourier. El filtro de Wiener de umbralización dura (8) es un filtro adaptativo que puede adaptar sus características a las propiedades del espectro local. En este trabajo, se propone realizar el procesamiento en el dominio espectral utilizando DCT en lugar de la transformada de Fourier, dentro de los bloques de imagen solapados. La transformación se realiza mediante la multiplicación por la matriz (Ec. 9):

$$\mathbf{U}^{(m)} = \mathbf{T}_{DCT}^{(m)} \mathbf{u}^{(m)} \left(\mathbf{T}_{DCT}^{(m)} \right)^T, \quad (9)$$

donde $\mathbf{u}^{(m)}$ es el bloque de la imagen de entrada de tamaño $m \times m$; se elige $m = 8$ para propósitos de comparación con otros filtros. El bloque de la imagen filtrada se puede obtener tomando la transformada inversa: $\mathbf{u}^{(m)} = \left(\mathbf{T}_{DCT}^{(m)} \right)^T \left(H_T(\mathbf{U}^{(m)}) \right) \mathbf{T}_{DCT}^{(m)}$.

3. Técnica de filtrado propuesta

En este trabajo presentamos una técnica avanzada de filtrado basada en la estrategia de filtrado BM3D [11] que supone la búsqueda de bloques similares en las cercanías del bloque actual de la imagen, formando listas de parches con los coeficientes DCT, su procesamiento mediante la transformada de Hadamard, umbralización dura y el agregado de los parches procesados para formar la imagen filtrada. La técnica propuesta comienza con la etapa de preprocesamiento para formar los parches DCT para cada píxel de la imagen en el rango de $(M - m) \times (N - m)$ píxeles, donde $M \times N$ es el tamaño de la imagen donde se realiza la búsqueda de los bloques de imagen similares con los datos transformados. Después del preprocesamiento y el primer paso de filtrado, se realiza el filtrado final con la imagen pre-filtrada; para ruido de varianza pequeña se usa el filtrado con KLT; en el caso contrario alto, se utiliza el filtrado de Wiener de acuerdo con (3).

3.1. Algoritmo de búsqueda de bloques similares

El algoritmo propuesto realiza la búsqueda de bloques similares dentro de un rango especificado de entre los parches de bloques transformados por la DCT (9). Es una variante de la búsqueda jerárquica que calcula las distancias desde el valor medio del bloque corriente i, j -ésimo de la imagen y los valores medios de los parches de bloques

dentro del rango especificado; estos valores corresponden a la componente de corriente directa (DC) con $\omega_x = \omega_y = 0$ del espectro (Ec. 10):

$$\mathbf{d}^1(i, j) = \left\{ \left(U_{1,1}(i, j) - U_{1,1}(i \pm shift, j \pm shift) \right)^2 \right\}, \quad (10)$$

donde el parámetro *shift* define el rango de la zona de búsqueda. Luego, se ordenan según las distancias, y la mitad de los bloques encontrados se consideran para la búsqueda más precisa usando los coeficientes DCT de los bloques encontrados en el paso siguiente (Ec. 11):

$$\mathbf{d}^2(i, j) = \left\{ \left(U_{p,q}(i, j) - d_k^1(i, j, U_{p,q}^{\mathbf{d}^1(i,j)}) \right)^2 \mid p, q \in \overline{1,5}; k \in \overline{1, (shift+1)^2/2} \right\}, \quad (11)$$

donde $d_k^1(\cdot)$ denota el parche k -ésimo de los coeficientes DCT encontrado en el primer paso. Nótese que p, q varían de 1 a 5, excluyendo en este paso la componente DC, $U_{1,1}$. La complejidad computacional de la técnica de búsqueda propuesta es considerablemente menor que en el caso del cálculo de todas las distancias entre los píxeles del bloque actual $\mathbf{u}^{(m)}$ y los píxeles de todos bloques dentro del rango:

$$\mathbf{d}(i, j) = \left\{ \left(u(i, j) - u(i+k, j+k) \right)^2 \mid i, j \in \overline{1, m}; k \in \overline{1, (shift+1)^2/2} \right\}.$$

En la etapa final de la búsqueda de bloques, se forma y se guarda la lista de las coordenadas de los parches, $\mathbf{I}_U(i, j) = \{ l_1(i, j) \cdot l_{b_{\max}}(i, j) \}$, donde $l_1(i, j) \cdot l_{b_{\max}}(i, j)$, b_{\max} es el número máximo de parches similares, y el vector con las distancias correspondientes, para su uso en el proceso de filtrado final.

3.2. Filtrado de los parches de DCT usando la transformada de Hadamard

En la segunda etapa, se lleva a cabo un filtrado preliminar para formar la estimación de la imagen para su uso en la tercera etapa. Antes de este filtrado, la estimación del espectro \hat{P}_s del actual i, j -ésimo bloque de imagen se calcula usando la lista de parches $\mathbf{I}_U(i, j)$ (Ec. 12):

$$\hat{P}_s(i, j) = \sum_{k=1}^{b_{\max}/2} U_k(l_k(i, j)) \cdot w_k^P(i, j), \quad (12)$$

donde $w_k^P(i, j)$ es un coeficiente de ponderación normalizado calculado a partir de las distancias entre los parches (Ec. 13):

$$w_k^P(i, j) = \frac{\exp\{-d_k(i, j)/200\}}{\|\mathbf{w}^P(i, j)\|}, \quad (13)$$

donde $\|\mathbf{w}^P(i, j)\|$ denota la suma de los coeficientes no normalizados. Con la estimación \hat{P}_s , la respuesta en frecuencia del filtro Wiener provisional en la posición del i, j -ésimo bloque $\hat{H}_{i,j}(\omega_x, \omega_y)$ se forma en concordancia con (3). Después, la transformada de Hadamard en tercera dimensión se aplica al grupo de parches $\mathbf{U}(i, j) = \{\mathbf{U}_k(l_k(i, j)), k = \overline{1, b_{\max}}\}$, $\mathbf{U}_{Hadamard} = Hadamard\{\mathbf{U}(i, j)\}$. Se propone aplicar a $\mathbf{U}_{Hadamard}$ el siguiente umbralado duro (Ec.14):

$$\tilde{\mathbf{U}}_{Hadamard} = \begin{cases} \mathbf{U}_{Hadamard}(\omega_x, \omega_y) & \text{si } \omega_x = \omega_y = 0 \\ \mathbf{U}_{Hadamard}(\omega_x, \omega_y) & \text{si } |\mathbf{U}_{Hadamard}(\omega_x, \omega_y)| \geq T(\omega_x, \omega_y), \\ 0 & \text{en otro caso} \end{cases} \quad (14)$$

donde los umbrales se forman usando $\hat{H}_{i,j}(\omega_x, \omega_y)$ como se ve en la ecuación (15).

$$T(\omega_x, \omega_y) = \begin{cases} \beta_{\min} \cdot \sigma & \text{si } \hat{H}_{i,j}(\omega_x, \omega_y) > 0.87 \\ \beta_{\max} \cdot \sigma & \text{si } \hat{H}_{i,j}(\omega_x, \omega_y) < 0.3 \\ \beta \cdot \sigma & \text{en otro caso} \end{cases} \quad (15)$$

y $\beta_{\min} = 1$, $\beta_{\max} = 2.9$; $\beta = 2.55377397$ de la ecuación (8).

Al realizar el umbralado y la transformada inversa de Hadamard, $\tilde{\mathbf{U}}(i, j) = Hadamard^{-1}\{\tilde{\mathbf{U}}_{Hadamard}\}$, se realiza el agrupamiento de todos los parches del grupo $\mathbf{I}(i, j)$. El agregado promediado se calcula, como

$$\begin{aligned} \hat{\mathbf{s}}_{\Sigma}(\mathbf{I}(i, j)) &= \sum_{k=1}^{b_{\max}} (\hat{\mathbf{U}}_k(l_k(i, j))) \cdot w_k^A(i, j), \quad R_{\Sigma}(\mathbf{I}(i, j)) = \sum_{k=1}^{b_{\max}} w_k^A(i, j), \\ w_k^A(i, j) &= \exp\{-d_k(i, j)/500\}, \\ (\tilde{\mathbf{s}})_{*} &= \frac{(\hat{\mathbf{s}}_{\Sigma})_{*}}{(R_{\Sigma})_{*}} \end{aligned} \quad (16)$$

donde $\hat{\mathbf{s}}_{\Sigma}(\mathbf{I}(i, j))$ es la suma de los bloques definidos por la lista $\mathbf{I}(i, j)$, $R_{\Sigma}(\mathbf{I}(i, j))$ es la suma de los coeficientes de ponderación $w_k^A(i, j)$, $\tilde{\mathbf{s}}$ es la estimación de la imagen que resulta de la etapa de prefiltrado y $(\)_{*}$ indica las operaciones para cada elemento.

3.3. Filtrado final usando minimización de representación escasa

En la tercera etapa, si la varianza del ruido es mayor o igual que 100, $\sigma^2 \geq 100$, se aplica el filtro de Wiener empírico a los grupos de bloques encontrados en la primera etapa, $\mathbf{I}(i, j)$ (Ec. 17):

$$\mathbf{U}_{Hadamard} = Hadamard\{\mathbf{U}(i, j)\},$$

$$\begin{aligned} \mathbf{S}_{Hadamard} &= Hadamard \{ \mathbf{S}(i, j) \}, \mathbf{S}(i, j) = \{ \mathbf{S}_k(l_k(i, j)), k = \overline{1, b_{\max}} \} \\ \hat{\mathbf{U}}(i, j) &= Hadamard^{-1} \left\{ (\mathbf{U}_{Hadamard})_* \frac{(\tilde{\mathbf{S}})_*}{(\tilde{\mathbf{S}} + \sigma^2)_*} \right\}, \end{aligned} \quad (17)$$

donde $\tilde{\mathbf{S}} = \mathbf{T}_{DCT} \tilde{\mathbf{s}} (\mathbf{T}_{DCT})^T$ es la transformada DCT del bloque de la imagen pre-filtrada $\tilde{\mathbf{s}}$.

En el caso de ruido de pequeña varianza, $\sigma^2 < 100$, se propone el tratamiento siguiente: si la varianza de los datos del bloque es pequeña, i.e, menor que $2\sigma^2$, se toma como el resultado la estimación previa, $\hat{\mathbf{s}} = \tilde{\mathbf{s}}$; en cualquier otro caso, se realiza la minimización de la representación dispersa. Para ello, primero se calcula la matriz de covarianza del bloque (i, j) -ésimo (Ec. 18):

$$\mathbf{C}(i, j) = \frac{(\mathbf{s}(i, j) - \tilde{\mathbf{s}}(i, j)) \cdot (\mathbf{s}(i, j) - \tilde{\mathbf{s}}(i, j))^T}{m \times m - 1}. \quad (18)$$

En este cálculo, proponemos usar los datos prefiltrados en calidad de la estimación de valores promedio de los píxeles para obtener mejores resultados en el filtrado.

Después, los eigenvectores \mathbf{Q} de $\mathbf{C}(i, j)$ se determinan y se calcula la KLT de los primeros dos bloques de la lista en $\mathbf{I}(i, j)$, se calcula: $\mathbf{S} = \mathbf{Q}(\mathbf{s}(i, j) - \tilde{\mathbf{s}}(i, j))$ y se realiza la umbralización dura (Ec. 19):

$$(\hat{\mathbf{S}})_* = \begin{cases} (\mathbf{S})_* & \text{if } (\mathbf{S})_* < 2.7\sigma \\ 0 & \text{en otro caso} \end{cases}, \quad (19)$$

Con ello se obtiene la estimación de los datos del bloque (Ec. 20):

$$\hat{\mathbf{s}} = \mathbf{Q}^T \hat{\mathbf{S}} + \tilde{\mathbf{s}}. \quad (20)$$

Finalmente, el agregado de los dos bloques procesados es similar a (16), la que se calcula usando los pesos $w_k(i, j) = \exp \{-d_k(i, j)/40\}$.

4. Resultados

Se realizaron las simulaciones numéricas con imágenes de prueba estándares de regiones planas “Lena”, “F-16”, “Pimientos” e imágenes texturales “Aérea”, “Babuino”, “Puente”. Para efectos de comparación, se consideraron los tres mejores filtros del estado de arte: K-SVD, BM3D (con la misma área de búsqueda definido por el parámetro $shift = 21$) y NCSR.

Los resultados del filtrado en términos de valores obtenidos de PSNR y SSIM se presentan en la tabla 1 y la tabla 2, respectivamente. La calidad visual de las imágenes procesadas por los filtros considerados es muy similar, aunque NCSR presenta algunos efectos de suavización excesiva y BM3D en algún momento introduce artefactos visibles.

De los datos presentados en las tablas 1 y 2 se desprende que la técnica propuesta de filtrado compite en sus características con las de los mejores filtros actuales en el sentido de PSNR, y casi en todos los casos es superior en términos del índice de similitud estructural.

Tabla 1. Resultados del filtrado de imágenes estándares de prueba con diferentes técnicas en términos de PSNTR. Los mejores resultados están marcados con negrita.

| Imagen | σ^{-2} | Ruidosa | K-SVD | BM3D | NCSR | Propuesta |
|-----------|---------------|---------|---------------|---------------|---------------|---------------|
| Lena | 25 | 38.554 | 38.686 | 38.674 | 38.674 | 38.801 |
| | 100 | 35.428 | 35.906 | 35.866 | 35.866 | 35.888 |
| | 225 | 33.544 | 34.263 | 34.175 | 34.175 | 34.169 |
| | 400 | 32.21 | 33.056 | 32.867 | 32.867 | 32.895 |
| F-16 | 25 | 39.067 | 39.258 | 39.232 | 39.232 | 39.286 |
| | 100 | 35.479 | 35.854 | 35.849 | 35.849 | 35.837 |
| | 225 | 33.40 | 33.886 | 33.887 | 33.887 | 33.855 |
| | 400 | 22.191 | 31.928 | 32.508 | 32.443 | 32.468 |
| Pimientos | 25 | 34.146 | 37.657 | 37.609 | 37.866 | 37.774 |
| | 100 | 28.167 | 34.766 | 35.024 | 35.081 | 35.009 |
| | 225 | 24.681 | 33.239 | 33.732 | 33.69 | 33.621 |
| | 400 | 22.22 | 32.073 | 32.723 | 32.613 | 32.564 |
| Aérea | 25 | 34.145 | 36.673 | 37.08 | 37.04 | 36.98 |
| | 100 | 28.142 | 32.289 | 32.792 | 32.789 | 32.637 |
| | 225 | 24.644 | 29.879 | 30.438 | 30.481 | 30.329 |
| | 400 | 22.199 | 28.187 | 28.835 | 28.812 | 28.765 |
| Babuino | 25 | 34.141 | 35.177 | 35.247 | 35.264 | 35.277 |
| | 100 | 28.135 | 30.451 | 30.61 | 30.598 | 30.696 |
| | 225 | 24.618 | 27.960 | 28.217 | 28.232 | 28.318 |
| | 400 | 22.129 | 26.307 | 26.642 | 26.64 | 26.772 |
| Puente | 25 | 34.159 | 35.578 | 35.765 | 35.695 | 35.698 |
| | 100 | 28.169 | 30.94 | 31.207 | 31.143 | 31.157 |
| | 225 | 24.682 | 28.544 | 28.822 | 28.824 | 28.842 |
| | 400 | 22.23 | 26.98 | 27.284 | 27.259 | 27.377 |

5. Conclusiones

Se ha presentado una técnica novedosa de filtrado para procesar imágenes contaminadas por el ruido aditivo blanco Gaussiano. El algoritmo propuesto utiliza la transformada discreta del coseno y grupos de parches similares al bloque actual de la imagen, los cuales se hallan utilizando el algoritmo de búsqueda propuesto de complejidad reducida. Los componentes ruidosos son rechazados según la teoría de filtrado de Wiener mediante el umbralado de la transformada de Hadamard y el agregado ponderado. Para obtener la imagen final se utiliza una etapa adicional de análisis de las componentes principales o filtro de Wiener experimental. Los resultados obtenidos de filtrado en comparación con los resultados de los filtros del estado del arte, tales como los conocidos como K-SVD y BM3D, muestran que el algoritmo propuesto es

competitivo en términos de la relación (pico de señal/ ruido) y en casi todos los casos, es superior en términos de la similitud estructural. A su vez, la calidad visual de los filtros considerados es muy similar al nuestro, aunque el filtro propuesto no suaviza excesivamente los detalles de la imagen ni introduce artefactos visibles.

Tabla 2. Resultados del filtrado de las imágenes estándares de prueba con diferentes técnicas en términos de SSIM. Los mejores resultados están marcados con negrita.

| Imagen | σ^{2l} | Ruidosa | KSVD | BM3D | NCSR | Propuesta |
|-----------|---------------|----------|-----------------|-----------------|-----------------|-----------------|
| Lena | 25 | 0.650071 | 0.729343 | 0.706594 | 0.702052 | 0.729986 |
| | 100 | 0.434195 | 0.612688 | 0.620161 | 0.616173 | 0.635876 |
| | 225 | 0.321075 | 0.54803 | 0.570386 | 0.564288 | 0.582004 |
| | 400 | 0.250943 | 0.503566 | 0.532345 | 0.516749 | 0.538954 |
| F-16 | 25 | 0.573694 | 0.677342 | 0.67417 | 0.664878 | 0.687661 |
| | 100 | 0.407647 | 0.571205 | 0.583485 | 0.574901 | 0.595023 |
| | 225 | 0.322104 | 0.516994 | 0.531316 | 0.520495 | 0.540428 |
| | 400 | 0.266426 | 0.477619 | 0.493573 | 0.470992 | 0.500174 |
| Pimientos | 25 | 0.698161 | 0.742456 | 0.704669 | 0.736183 | 0.735144 |
| | 100 | 0.460156 | 0.588543 | 0.579873 | 0.588272 | 0.60069 |
| | 225 | 0.334244 | 0.526644 | 0.534485 | 0.527292 | 0.548582 |
| | 400 | 0.259053 | 0.490874 | 0.503144 | 0.487441 | 0.514074 |
| Aérea | 25 | 0.84257 | 0.899446 | 0.907226 | 0.900802 | 0.906303 |
| | 100 | 0.697669 | 0.81245 | 0.832167 | 0.826998 | 0.832912 |
| | 225 | 0.587101 | 0.740628 | 0.771846 | 0.768144 | 0.772791 |
| | 400 | 0.500772 | 0.671283 | 0.71822 | 0.707465 | 0.720075 |
| Babuino | 25 | 0.91839 | 0.929301 | 0.926673 | 0.924035 | 0.927581 |
| | 100 | 0.790511 | 0.824671 | 0.828144 | 0.81302 | 0.841045 |
| | 225 | 0.678611 | 0.733180 | 0.751595 | 0.734304 | 0.76799 |
| | 400 | 0.585645 | 0.654664 | 0.685388 | 0.65984 | 0.706898 |
| Puente | 25 | 0.913643 | 0.938448 | 0.941021 | 0.937837 | 0.939819 |
| | 100 | 0.780924 | 0.849257 | 0.859999 | 0.854613 | 0.863266 |
| | 225 | 0.658758 | 0.755813 | 0.777584 | 0.774548 | 0.787016 |
| | 400 | 0.556196 | 0.666819 | 0.701251 | 0.691471 | 0.718897 |

Agradecimientos. Este trabajo fue parcialmente apoyado por el Instituto Politécnico Nacional como parte del proyecto de investigación SIP #20161173.

Referencias

1. Pratt, W.K.: Digital Image Processing. 4th Edition, NY, USA, Wiley-Interscience (2007)
2. Buades, A., Coll, B., Morel, J., M.: A review of image denoising algorithms, with a new one. J. SIAM, vol. 2, no. 4 (2005)
3. Daboy, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3D transform-domain collaborative filtering. IEEE Transactions on Image Processing, Vol. 16, No. 8, pp. 2080–2095 (2007)

4. Foi, A., Katkovnik, V., Egiazarian, K.: Pointwise Shape-Adaptive DCT for High-Quality Denoising and Deblocking of Grayscale and Color Images. *IEEE Trans. Image Process.*, vol. 16, no. 5, pp. 1395–1411 (2007)
5. Chatterjee, P., Milanfar, P.: Is Denoising Dead. *IEEE Trans. on Image Processing*, Vol. 19, No. 4, pp. 895–911 (2010)
6. Aharon, M., Elad, M., Bruckstein, A., M.: K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Processing*, vol. 54, no. 11, pp. 4311–4322 (2006)
7. Weisheng Dong, Lei Zhang, Guangming Shi, Xin Li: Nonlocally Centralized Sparse Representation for Image Restoration. *IEEE Trans. on Image Processing*, vol. 22, no. 4, pp. 1620–1630 (2013)
8. Ning He, Jin-Bao Wang, Lu-Lu Zhang, Guang-Mei Xu, Ke Lu: Non-local sparse regularization model with application to image denoising. *Multimedia Tools and Applications*, Vol. 75, Issue 5, pp. 2579–2594 (2016)
9. Oleksiy Pogrebnyak, Vladimir V. Lukin: Wiener discrete cosine transform-based image filtering. *Journal of Electronic Imaging*, Volume 21, Issue 4, USA (2012)
10. Fevraleev, D., Lukin, V., Ponomarenko, N., S., Abramov, K., Egiazarian, J. Astola: Efficiency analysis of color image filtering. *EURASIP Journal on Advances in Signal Processing*, Vol. 41 (2011)
11. Marc Lebrun: An Analysis and Implementation of the BM3D Image Denoising Method. *Image Processing On Line*, pp. 175–213, <http://dx.doi.org/10.5201/ipol.2012.1-bm3d> (2012)
12. Vladimir Lukin, Sergey Abramov, Sergey Krivenko, Andriy Kurekin, Oleksiy Pogrebnyak: Analysis of classification accuracy for pre-filtered multichannel remote sensing data. *Expert Systems With Applications*, Vol. 40, Issue 16, pp. 6400–6411 (2013)
13. Egiazarian K., Astola J., Ponomarenko N., Lukin V., Battisti F., Carli M.: New full-reference quality metrics based on HVS. *CD-ROM Proceedings of the Second International Workshop on Video Processing and Quality Metrics*, Scottsdale, USA (2006)
14. Lin Zhang, Lei Zhang; Xuanqin Mou; David Zhang: FSIM: A Feature SIMilarity index for image quality assessment. *IEEE Transactions on Image Processing*, pp. 2378–2386 (2011)
15. Wang, Zhou; Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, pp. 600–612 (2004)
16. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, Vol.2 , pp. 1398–1402 (2003)

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
noviembre de 2016
Printing 500 / Edición 500 ejemplares

