

SEP

SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO
Instituto Tecnológico de La Paz



Robótica y Computación, Retos y Perspectivas



TECNOLÓGICO
NACIONAL DE MÉXICO

Coordinadores de la Edición
Iliana Castro Liera
Mario Cortés Larrinaga



ISBN: 978-607-97128-7-7



Esta página fue dejada en blanco intencionalmente.

SEP

SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO
Instituto Tecnológico de La Paz

Robótica y Computación, Retos y Perspectivas

D. R. ©INSTITUTO TECNOLÓGICO DE LA PAZ
Blvd. Forjadores de B.C.S. No. 4720
Col. 8 de octubre
C.P. 23080
La Paz Baja California Sur, México

Primera edición mayo de 2018
ISBN: 978-607-97128-7-7



Reservados todos los derechos. Ninguna parte de este libro puede ser reproducida, archivada o transmitida, en cualquier sistema –electrónico, mecánico, de fotorreproducción, de almacenamiento en memoria o cualquier otro– sin hacerse acreedor a las sanciones establecidas en las leyes, salvo con el permiso escrito del titular de los derechos de la obra.

Diseño de la portada: Iliana Castro Liera

Esta página fue dejada en blanco intencionalmente.

Coordinador General

Dr. Lorenzo Fidel Cota Verdugo - Instituto Tecnológico de La Paz

Coordinador del Programa Técnico

Dr. Marco Antonio Castro Liera - Instituto Tecnológico de La Paz

Comité Revisor

Dr. Francisco Herrera Fernández - UCLV Cuba

Dr. Alejandro Israel Barranco Gutiérrez - Instituto Tecnológico de Celaya

Dr. Víctor Adrián Santibáñez Dávila - Instituto Tecnológico de La Laguna

Dr. Ricardo Emmanuel Campa Cocom - Instituto Tecnológico de La Laguna

Dr. Alejandro Enrique Dzúl López - Instituto Tecnológico de La Laguna

M.C. Edmundo Javier Ollervides Vázquez - Instituto Tecnológico de La Laguna

Dr. Eduardo Javier Moreno Valenzuela - CITEDI Tijuana

Dr. Eusebio Bugarín Carlos - Instituto Tecnológico de Ensenada

Dra. Ana Yaveni Aguilar Bustos - Instituto Tecnológico de Ensenada

Dr. Jesús Alberto Sandoval Galarza - Instituto Tecnológico de La Paz

M.S.C. Iliana Castro Liera - Instituto Tecnológico de La Paz

Dr. Saúl Martínez Díaz - Instituto Tecnológico de La Paz

Dr. Israel Marcos Santillán Méndez - Instituto Tecnológico de La Paz

Dr. Marco Antonio Castro Liera - Instituto Tecnológico de La Paz

M.C. Jesús Antonio Castro - Instituto Tecnológico de La Paz

M.A.T.I. Luis Armando Cárdenas Florido - Instituto Tecnológico de La Paz

M.S.C. José Luís Gómez Torres - Instituto Tecnológico de La Paz

M.S.C. Diego García Molleda - Instituto Tecnológico de La Paz

M.S.C. César Higuera Verdugo - Instituto Tecnológico de La Paz

Coordinadores de la edición

M.S.C. Iliana Castro Liera - Instituto Tecnológico de La Paz

Dr. Mario Cortés Larrinaga - Instituto Tecnológico de La Paz

Índice general

| | |
|---|-----|
| Portada | I |
| Comités | V |
| Índice | VI |
| Control de seguimiento de trayectorias para un giroscopio subactuado | 1 |
| Dynamic tracking control for mechanical systems using only position measurements | 7 |
| Modeling of a Speed Sensor and its Application in a Sensorless Control System | 11 |
| Sistema de control de camas ortopédicas para pacientes con cuadriplejía por medio de voz | 16 |
| Sistema para el monitoreo y localización de vehículos usando IoT | 23 |
| Estrategias basadas en cómputo ubicuo para el control de la temperatura en un invernadero | 30 |
| MRAC basado en la regla MIT y el método de Lyapunov para el control en altitud de un UAV de ala fija | 35 |
| Síntesis de controladores neuronales para locomoción y auto-ensamblado en robots modulares MARBot | 40 |
| Modelado, Control y Sintonía de un Péndulo Invertido sobre Ruedas | 46 |
| Modelado y control IDA-PBC de un robot bipedo tipo compás con fricción | 53 |
| Modelado y Comparativa de un Controlador LQR y un Controlador Difuso en un Péndulo de Furuta | 59 |
| Clasificador difuso para la señal de referencia de un controlador PI para regular el inventario en las cadenas de suministro | 66 |
| Reporte Lingüístico Automático para el Proceso de Compostaje | 71 |
| Sistema de clasificación de documentos basado en categorías de Wikipedia | 76 |
| Detección de Alteraciones en Imágenes mediante Watermarking basado en DWT y Cómputo Paralelo | 83 |
| Marca de Agua Visible en Imágenes Digitales a Color utilizando la Distorsión Apenas Perceptible | 90 |
| Pseudocoloración y Segmentación de Tomografías Axiales Computarizadas CT Cráneo-Cerebrales Mediante Operador de Gradiente y Morfológico | 95 |
| Cálculo Inicial de Profundidad para un Sistema de Visión Monocular | 100 |
| Algoritmo genético para la generación de trayectorias de un robot industrial con evasión de colisiones para una interacción segura humano-robot | 104 |
| Desarrollo de un algoritmo genético para análisis espacial de conglomerados con formas irregulares | 110 |
| Implementación de un Cluster Heterogéneo de GPGPU Para Algoritmo Genético | 114 |
| Sensor Linealization using Linear Genetic Programming | 118 |

| | |
|--|-----|
| Propuesta de una arquitectura distribuida orientada a servicios para la adopción de la industria 4.0 en la manufactura de celdas solares | 124 |
| Sistemas de Gestión de Seguridad de la Información, un reto para las universidades de México | 128 |
| Variante del Algoritmo de Proyecciones Afines con Error Codificado y Stop and Go para Comunicación 16-QAM | 133 |
| Modelado Básico con Autómatas Celulares para Epidemias de Gusanos Informáticos | 137 |
| Impacto de dos Mecanismos de Planificación de Tiempo Real con RT-Linux sobre una Computadora 17:40 Embebida | 141 |
| Implementación en FPGA de un sistema de identificación de parámetros de modelos MA utilizando un algoritmo Levinson-Durbin modificado | 151 |
| Modelo cinemático directo basado en cuaterniones duales del manipulador robótico ABB IRB6400 | 156 |
| Diseño e implementación de controlador para aeromodelo por retroalimentación de salida | 163 |
| Modelado Cinemático y Dinámico de un Robot Móvil No Holonómico Considerando Dinámica de Actuadores | 167 |
| Sistema de Visión para Parametrizar la Trayectoria de Extremidades en Marcha Bípeda | 173 |
| Análisis dinámico y simulación de un mecanismo para rehabilitación de miembro inferior | 178 |
| Desarrollo de una GUI para la sintonización de ganancias y simulación de esquemas de control adaptable | 184 |
| Diseño de Librería en Matlab con Enfoque Didáctico para el Robot Móvil Diferencial Khepera IV | 191 |
| Diseño y construcción de un guante portable para medición de flexión-extensión de falanges | 198 |
| XIGUAR: Exoesqueleto para rehabilitación en terapia espejo | 204 |

Control de seguimiento de trayectorias para un giroscopio subactuado*

J. Moreno–Valenzuela

Instituto Politécnico Nacional–CITEDI
Av. Instituto Politécnico Nacional 1310
Col. Nueva Tijuana, Tijuana
Baja California, Mexico, 23435
Email: moreno@citedi.mx

J. Montoya

Instituto Politécnico Nacional–CITEDI
Av. Instituto Politécnico Nacional 1310
Col. Nueva Tijuana, Tijuana
Baja California, Mexico, 23435
Email: jamontoya@citedi.mx

V. Santibáñez

Tecnológico Nacional de México
Instituto Tecnológico de La Laguna
Blvd. Revolución y Cuauhtémoc SN
Torreón, Mexico, 27000
Email: vsantiba@itlalaguna.edu.mx.

Abstract—El control en sistemas mecánicos subactuados ha sido una tendencia importante en los sistemas mecatrónicos y no lineales. Algunos ejemplos son el péndulo de Furuta y el péndulo de rueda inercial. Por otra parte, los giroscopios son importantes en sistemas aéreos y espaciales. En este artículo, se estudia un giroscopio de dos grados de libertad subactuado. Las articulaciones de este sistema se encuentran altamente acopladas y es un sistema difícil de controlar, lo cual lo hace un sistema de referencia importante. Para este sistema se estudia el problema de control de seguimiento de trayectorias y se brinda una solución. Resultados de simulación comprueban los resultados teóricos.

I. INTRODUCCIÓN

El control en sistemas subactuados ha sido un problema prolífico en estos últimos años. Las razones son que estos sistemas aparecen en muchas aplicaciones prácticas y pueden ser usados como sistemas de referencia para probar nuevas leyes de control y aplicaciones. La mayoría de los trabajos en sistemas subactuados de dos grados de libertad están enfocados en control de regulación en configuraciones estables o inestables. Sólo pocos trabajos han abordado el problema de control del seguimiento de trayectorias en sistemas subactuados [1], [2].

Se presenta una breve reseña de la literatura. En [3], se propuso una formulación de control para sistemas no holonómicos para la dinámica de un giroscopio subactuado. En [4], se predijeron ciclos límites en un giroscopio afectado por no linealidades. Además, se propuso un nuevo controlador para anular las oscilaciones. Más recientemente, en [5], se obtuvo un algoritmo de desacoplamiento combinado con la técnica de linealización por realimentación entrada-salida para un giroscopio con velocidad variable montado en dos cardanes.

En [6], se estudió la sincronización de dos giroscopios subactuados. La sincronización se logra usando un control de estructura variable y redes neuronales adaptables.

Cabe a destacar que se han sintetizado giroscopios como sistemas microelectromecánicos subactuados (MEMS, por sus siglas en inglés). En [7], se han estudiado giroscopios MEMS subactuados y se reporta un esquema de control que proporciona robustez inherente.

*Esta investigación fue apoyada parcialmente por SIP–IPN, por el proyecto CONACYT n.º 134534 y Proyectos TecNM.

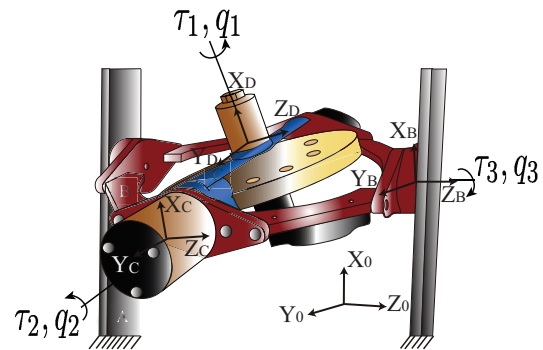


Fig. 1. Giroscopio estudiado en este artículo.

Una aplicación interesante de los giroscopios se reportó en [13]. Específicamente, se utiliza un mecanismo tipo giroscopio para controlar un robot esférico. Todo el sistema se considera como un sistema subactuado.

Este documento presenta un nuevo controlador para un sistema subactuado que consiste en un giroscopio de dos grados de libertad. El nuevo esquema está basado en la teoría de linealización por realimentación. Se propone una nueva función de salida, la cual tiene la ventaja de moldear la dinámica interna. En términos generales, el problema de control consiste en lograr el seguimiento de trayectorias de una articulación (la articulación que no es afectada por ninguna entrada de control), mientras que se trata de mantener regulada la otra articulación regulada. El resultado es un controlador que logra que el error de trayectorias sea uniformemente últimamente acotadas. Resultados en simulación avalan los resultados teóricos.

Este artículo se encuentra organizado de la siguiente manera. La sección II está dedicada al modelo del giroscopio y a la formulación del problema de control. En la sección III se presenta el diseño del control basado en linealización por realimentación. Los resultados de simulación se presentan en la sección IV. Finalmente, las conclusiones derivadas del presente estudio son dadas en la sección V.

II. MODELO DEL GIROSCOPIO Y PLANTEAMIENTO DEL PROBLEMA

A. Modelo del giroscopio

Se considera el giroscopio mostrado en la figura 1, el cual es un giroscopio de dos grados de libertad, se tiene en cuenta que el disco se encuentra girando a velocidad constante, y considerando subactuación, como se muestra en [10], se puede escribir su modelo como

$$\tau_2 = J_3 \dot{q}_2 + P_1 \cos(q_2) \dot{q}_3 - J_a \sin(q_2) \cos(q_2) \dot{q}_3^2 \quad (1)$$

$$0 = (J_2 + J_a \sin(q_2)^2) \ddot{q}_3 - P_1 \cos(q_2) \dot{q}_2 + 2J_a \sin(q_2) \cos(q_2) \dot{q}_2 \dot{q}_3, \quad (2)$$

En la figura 1, la coordenada q_1 expresa la posición del disco. No obstante, esta no aparece en el modelo (1)–(2) dado que es una coordenada cíclica [11] y la cantidad de movimiento P_1 asociada es constante.

Los parámetros asociados a las ecuaciones (1) y (2) se definen como

$$J_1 = I_{C_{xx}} - I_{C_{zz}} - I_{D_{zz}} + I_{D_{xx}}, \quad (3)$$

$$J_2 = I_{B_{zz}} + I_{C_{zz}} + I_{D_{zz}}, \quad (4)$$

$$J_3 = I_{C_{yy}} + I_{D_{yy}}, \quad (5)$$

$$J_a = J_1 - I_{D_{xx}}, \quad (6)$$

Las constantes del lado derecho de las ecuaciones (3)–(6) son los elementos de los tensores de inercia con respecto a los marcos B , C y D que se muestra en la figura 1.

La constante P_1 es la cantidad de movimiento generada por el disco.

B. Planteamiento del problema

En esta formulación de control se considera el vector

$$\mathbf{q}_d(t) = \begin{bmatrix} 0 \\ q_{d3}(t) \end{bmatrix},$$

donde $q_{d3}(t)$ es una función diferenciable dos veces.

Se define el error de posición

$$\mathbf{e}(t) = \mathbf{q}_d(t) - \mathbf{q}(t) = \begin{bmatrix} -q_2(t) \\ q_{d3}(t) - q_3(t) \end{bmatrix} \in \mathbb{R}^2,$$

donde

$$\mathbf{q}(t) = \begin{bmatrix} q_2 \\ q_3 \end{bmatrix}$$

y se define

$$\mathbf{x} = \begin{bmatrix} e_1 \\ e_2 \\ \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} \in \mathbb{R}^4. \quad (7)$$

Similarmente a [12] y [1], el problema de control consiste en diseñar un controlador $\tau_2 \in \mathbb{R}$ tal que el error de trayectorias $\mathbf{x}(t) \in \mathbb{R}^4$ satisfaga la definición de una señal uniformemente últimamente acotada. En otras palabras, el controlador deberá garantizar

$$\|\mathbf{x}(t_0)\| \leq a \Rightarrow \|\mathbf{x}(t)\| \leq b \quad \forall t \geq t_0 + T, \quad (8)$$

con $T = T(a, b) \geq 0$.

III. DISEÑO DEL CONTROL BASADO EN LINEALIZACIÓN POR REALIMENTACIÓN

A. Sistema a lazo abierto

Usando la definición (7) y el modelo (1)–(2), es posible escribir

$$\frac{d}{dt} \begin{bmatrix} e_1 \\ e_2 \\ \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -1/J_3 \\ 0 \end{bmatrix} \tau_2, \quad (9)$$

donde

$$f_1 = \dot{e}_1,$$

$$f_2 = \dot{e}_2,$$

$$f_3 = \frac{P_1[\dot{q}_{d3} - \dot{e}_2] \cos(e_1)}{J_3} + \frac{J_a[\dot{q}_{d3} - \dot{e}_2]^2 \cos(e_1) \sin(e_1)}{J_3},$$

$$f_4 = \ddot{q}_{d3} + \frac{\dot{e}_1 P_1 \cos(e_1)}{J_a \sin(e_1)^2 + J_2} + \frac{2J_a \dot{e}_1 [\dot{q}_{d3} - \dot{e}_2] \cos(e_1) \sin(e_1)}{J_a \sin(e_1)^2 + J_2}.$$

Finalmente, nótese que el sistema a lazo abierto (9) puede ser escrito en la estructura

$$\frac{d}{dt} \mathbf{x} = \mathbf{f}(t, \mathbf{x}) + \mathbf{g}(\mathbf{x}) \tau_2,$$

donde \mathbf{x} está definido en (7).

En este artículo, se propone la siguiente función de salida

$$\begin{aligned} y &= \boldsymbol{\alpha}^T \mathbf{e} + \boldsymbol{\beta}^T \dot{\mathbf{e}} \\ &= \alpha_1 x_1 + \alpha_2 x_2 + \beta_1 x_3 + \beta_2 x_4, \end{aligned} \quad (10)$$

donde α_1 , α_2 , β_1 y β_2 son constantes definidas por el usuario y serán definidas posteriormente.

Se utiliza la metodología de linealización por realimentación para resolver el objetivo de control (8).

B. Diseño

La derivada temporal de la función de salida esta dada por

$$\begin{aligned} \dot{y} &= \boldsymbol{\alpha}^T \dot{\mathbf{e}} + \boldsymbol{\beta}^T \ddot{\mathbf{e}} \\ &= \alpha_1 x_1 + \alpha_2 x_2 + \beta_1 (f_3 - \frac{1}{J_3} \tau_2) + \beta_2 f_4. \end{aligned} \quad (11)$$

El grado relativo es $r = 1$.

De la expresión (11), se propone la siguiente ley de control

$$\tau_2 = \frac{J_3}{\beta_1} (k_p y + \alpha_1 x_1 + \alpha_2 x_2 + \beta_1 f_3 + \beta_2 f_4), \quad (12)$$

la cual convierte a la dinámica de la función de salida en el siguiente sistema lineal

$$\dot{y} = -k_p y.$$

Por lo tanto, $y(t) \rightarrow 0$ conforme $t \rightarrow \infty$ de forma exponencial.

C. Sistema de lazo cerrado

El sistema de lazo cerrado se obtiene usando la siguiente transformación

$$\begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ y \end{bmatrix} = H \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix},$$

donde $H \in \mathbb{R}^{4 \times 4}$ y

$$\frac{\partial n_i}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x}) = 0, \quad i = 1, 2, 3.$$

Se obtiene

$$\begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ y \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \alpha_1 & \alpha_2 & \beta_1 & \beta_2 \end{bmatrix}}_H \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}, \quad (13)$$

cuya inversa esta dada por

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -\frac{\alpha_1}{\beta_1} & -\frac{\alpha_2}{\beta_1} & -\frac{\beta_2}{\beta_1} & \frac{1}{\beta_1} \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ y \end{bmatrix}. \quad (14)$$

Usando las relaciones (13)-(14), y la definición

$$\rho = \frac{P_1 \cos(\eta_1) + 2J_a \cos(\eta_1) \sin(\eta_1) [\dot{q}_{d3} - \eta_3]}{J_a \sin(\eta_1)^2 + J_2}, \quad (15)$$

la dinámica interna se obtiene de la siguiente manera

$$\frac{d}{dt} \eta_1 = -\frac{\alpha_1}{\beta_1} \eta_1 - \frac{\alpha_2}{\beta_1} \eta_2 - \frac{\beta_2}{\beta_1} \eta_3 + \frac{1}{\beta_1} y, \quad (16)$$

$$\frac{d}{dt} \eta_2 = \eta_3, \quad (17)$$

$$\frac{d}{dt} \eta_3 = \ddot{q}_{d3} + \rho \dot{\eta}_1 \quad (18)$$

La dinámica interna (16)–(18) también se puede escribir como

$$\frac{d}{dt} \boldsymbol{\eta} = A \boldsymbol{\eta} + \boldsymbol{\psi}_1(\boldsymbol{\eta}, y) + \boldsymbol{\psi}_2(t), \quad (19)$$

donde

$$A = \begin{bmatrix} -\frac{\alpha_1}{\beta_1} & -\frac{\alpha_2}{\beta_1} & -\frac{\beta_2}{\beta_1} \\ 0 & 0 & 1 \\ -\Delta \frac{\alpha_1}{\beta_1} & -\Delta \frac{\alpha_2}{\beta_1} & -\Delta \frac{\beta_2}{\beta_1} \end{bmatrix}, \quad (20)$$

$$\boldsymbol{\psi}_1(\boldsymbol{\eta}, y) = \begin{bmatrix} y/\beta_1 \\ 0 \\ \frac{\Delta}{\beta_1} y + \Delta \dot{\eta}_1 + \rho \dot{\eta}_1 \end{bmatrix}, \quad \text{y} \quad \boldsymbol{\psi}_2(t) = \begin{bmatrix} 0 \\ 0 \\ \ddot{q}_{d3} \end{bmatrix},$$

con $\Delta = P_1/J_2$, que corresponde a la linealización de ρ en $\boldsymbol{\eta} = 0$.

Es posible demostrar que A tiene un valor propio nulo y los parámetros $\boldsymbol{\alpha} = [\alpha_1 \quad \alpha_2]^T$ y $\boldsymbol{\beta} = [\beta_1 \quad \beta_2]^T$ pueden ser seleccionados de tal manera que los otros dos valores propios sean reales y estrictamente negativos.

Ahora, el análisis se enfocará a estudiar la dinámica de ceros, la cual esta dada por (19) evaluada en $y = 0$, es decir,

$$\frac{d}{dt} \boldsymbol{\eta} = A \boldsymbol{\eta} + \boldsymbol{\psi}_1(\boldsymbol{\eta}, 0) + \boldsymbol{\psi}_2(t),$$

donde A está definida en (20) y $\boldsymbol{\psi}_1$ puede ser reescrita como

$$\boldsymbol{\psi}_1(0, \boldsymbol{\eta}) = B \boldsymbol{\eta},$$

donde

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ (\Delta - \rho) \frac{\alpha_1}{\beta_1} & (\Delta - \rho) \frac{\alpha_2}{\beta_1} & (\Delta - \rho) \frac{\beta_2}{\beta_1} \end{bmatrix}.$$

Ahora, se considera la transformación

$$\mathbf{w} = V^{-1} \boldsymbol{\eta}, \quad (21)$$

en donde V es dada por

$$V = \begin{bmatrix} \frac{1}{\Delta} & \frac{1}{\Delta} & -\frac{\alpha_2}{\alpha_1} \\ V_{21} & V_{22} & 1 \\ 1 & 1 & 0 \end{bmatrix} \quad (22)$$

con

$$V_{21} = -\frac{\beta_2}{2\alpha_2} - \frac{\alpha_1 - \gamma}{2\Delta\alpha_2}, \quad V_{22} = -\frac{\beta_2}{2\alpha_2} - \frac{\alpha_1 + \gamma}{2\Delta\alpha_2},$$

y

$$\gamma = [\Delta^2 \beta_2^2 + 2\Delta\alpha_1\beta_2 - 4\Delta\alpha_2\beta_1 + \alpha_1^2]^{1/2}.$$

Se considerará que $\det(V) = \frac{-\gamma}{\Delta\alpha_1} \neq 0$. La matriz A en (20) y la matriz V en (22) satisfacen

$$J = V^{-1} A V, \quad (23)$$

la cual es la forma normal de Jordan de la matriz A . Así, el sistema (19) se transforma en

$$\frac{d}{dt} \mathbf{w} = J \mathbf{w} + D \mathbf{w} + V^{-1} \boldsymbol{\psi}_2(t), \quad (24)$$

donde

$$J = \text{diag}\{\lambda_1, \lambda_2, 0\},$$

$$\lambda_1 = -\frac{(\alpha_1 + \Delta\beta_2)}{2\beta_1} - \frac{(\Delta^2\beta_2^2 + 2\Delta\alpha_1\alpha_2 - 4\alpha_2\beta_1\Delta + \alpha_1^2)^{1/2}}{2\beta_1},$$

$$\lambda_2 = -\frac{(\alpha_1 + \Delta\beta_2)}{2\beta_1} + \frac{(\Delta^2\beta_2^2 + 2\Delta\alpha_1\alpha_2 - 4\alpha_2\beta_1\Delta + \alpha_1^2)^{1/2}}{2\beta_1},$$

$$D = V^{-1} B V = \frac{\Delta - \rho}{2\beta_1} \begin{bmatrix} \frac{d_1}{\Delta} + \beta_2 & -\frac{d_2}{\Delta} + \alpha_1 & 0 \\ \frac{d_2}{\Delta} + \alpha_1 & -\frac{d_1}{\Delta} + \beta_2 & 0 \\ \frac{d_3 + \gamma}{\Delta^2\alpha_2} & \frac{d_3 - \gamma}{\Delta^2\alpha_2} & 0 \end{bmatrix},$$

con

$$d_1 = \alpha_1\beta_2 - 2\alpha_2\beta_1 + \Delta\beta_2^2,$$

$$d_2 = \alpha_1 + \Delta\alpha_1\beta_2 - 2\Delta\alpha_2\beta_1,$$

$$d_3 = \alpha_1^2 + \Delta\alpha_1\beta_2,$$

y

$$V^{-1} \boldsymbol{\psi}_2(t) = \begin{bmatrix} \frac{\Delta\beta_2 - \alpha_1 + \gamma}{2\gamma} \\ \frac{\alpha_1 - \Delta\beta_2 + \gamma}{2\gamma} \\ \frac{\alpha_1}{\Delta\alpha_2} \end{bmatrix} \ddot{q}_{d3}.$$

En base a los cálculos anteriores, la dinámica de ceros transformada en (24) se puede reescribir explícitamente como

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} &= \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} + \frac{\Delta - \rho}{2\beta_1} \\ &\times \underbrace{\begin{bmatrix} \frac{d_1}{\gamma} + \beta_2 & -\frac{d_2}{\Delta\gamma} + \alpha_1 \\ \frac{d_2}{\Delta\gamma} + \alpha_1 & -\frac{d_1}{\gamma} + \beta_2 \end{bmatrix}}_{\boldsymbol{\vartheta}} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \\ &+ \underbrace{\begin{bmatrix} \frac{\Delta\beta_2 - \alpha_1 + \gamma}{2\gamma} \\ \frac{\alpha_1 - \Delta\beta_2 + \gamma}{2\gamma} \end{bmatrix}}_{\boldsymbol{\vartheta}} \ddot{q}_{d3} \quad (25) \\ \frac{d}{dt} w_3 &= \frac{\Delta - \rho}{2\beta_1} \left[\frac{d_3 + \gamma}{\Delta^2 \alpha_2} w_1 + \frac{d_3 - \gamma}{\Delta^2 \alpha_2} w_2 \right] \\ &+ \frac{\alpha_1}{\Delta \alpha_2} \ddot{q}_{d3}. \quad (26) \end{aligned}$$

Dado que ρ en (15) es una señal continua para todo $\boldsymbol{\eta} \in B_r$, donde $B_r = \{\boldsymbol{\eta} \in \mathbb{R}^3 : \|\boldsymbol{\eta}\| \leq r\}$, se tiene que existe una constante c_0 suficientemente pequeña tal que

$$|\Delta - \rho| \leq c_0, \quad (27)$$

para todo $\boldsymbol{\eta} \in B_r$. De forma análoga, usando la transformación (21), la desigualdad (27) se satisface para $\boldsymbol{w} \in B_s$, donde $B_s = \{\boldsymbol{w} \in \mathbb{R}^3 : \|\boldsymbol{w}\| \leq s\}$.

Ahora se establecerá un resultado en las condiciones suficientes para el acotamiento de las soluciones $\boldsymbol{w}(t)$.

Proposición 1. *La transformación de la dinámica de ceros (24) tiene soluciones $\boldsymbol{w}(t)$ uniformemente últimamente acotadas bajo las siguientes condiciones:*

- C1 Los valores propios λ_1 y λ_2 son negativos y reales.
- C2 La señal $\ddot{q}_{d3}(t) = a \sin(\omega t)$, donde $a, \omega > 0$.

Prueba: Se considera la siguiente función definida positiva

$$L = \frac{1}{2} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}^T \begin{bmatrix} w_1 \\ w_2 \end{bmatrix},$$

cuya derivada temporal es

$$\begin{aligned} \dot{L} &= \lambda_1 w_1^2 + \lambda_2 w_2^2 + \frac{\Delta - \rho}{2\beta_1} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}^T E \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \\ &+ \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}^T \boldsymbol{\vartheta} \ddot{q}_{d3}, \quad (28) \end{aligned}$$

donde $E \in \mathbb{R}^{2 \times 2}$ y $\boldsymbol{\vartheta} \in \mathbb{R}^2$ son constantes y están definidas en la ecuación (25).

Usando la transformación $\left\| \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \right\| = \sqrt{2L} = x$, que satisface $\dot{x} = \frac{\dot{L}}{x}$, y dada la condición C1 y que $|\lambda_1| > |\lambda_2|$, se puede escribir

$$\dot{x} \leq -[|\lambda_2| - \frac{c_0}{2\beta_1} \|E\|]x + \|\boldsymbol{\vartheta}\| \ddot{q}_{d3},$$

Debe de notarse que la cota superior de \dot{x} tiene la forma de

$$\dot{x} \leq -c_1 x + c_2 \ddot{q}_{d3}, \quad (29)$$

donde $c_1 = |\lambda_2| - \frac{c_0}{2\beta_1}$, c_0 satisfaciendo la desigualdad (27), y $c_2 = \|\boldsymbol{\vartheta}\|$. En particular, note que $c_1 > 0$ para c_0 suficientemente pequeña y que $\ddot{q}_{d3}(t) = -\omega^2 a \sin(\omega t)$.

Usando el principio de comparación [14], se puede obtener una solución $x(t)$ que satisfaga (29). Usando la condición C2, esta solución es dada por

$$\begin{aligned} x(t) &\leq x(0)e^{-c_1 t} + c_2 e^{-c_1 t} \int_0^t |\ddot{q}_{d3}(t)| e^{c_1 t} dt \\ &= x(0)e^{-c_1 t} + \frac{ac_2 \operatorname{sgn}(a \sin(\omega t))(c_1 \sin(\omega t) - \omega \cos(\omega t))}{c_1^2 + \omega^2}. \quad (30) \end{aligned}$$

Por tanto, $[w_1(t) \ w_2(t)]^T \in \mathbb{R}^2$ es una señal uniformemente últimamente acotada.

La señal $w_3(t)$ puede calcularse usando (26) como sigue:

$$w_3(t) = w_3(0) + [\Delta - \rho] \int_0^t [c_3 \ c_4] \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} dt + c_5 \int_0^t \ddot{q}_{d3} dt \quad (31)$$

donde $c_3 = \frac{d_3 + \gamma}{2\beta_1 \Delta^2 \alpha_2}$, $c_4 = \frac{d_3 - \gamma}{2\beta_1 \Delta^2 \alpha_2}$ y $c_5 = \frac{d_1}{\Delta \alpha_2}$.

La solución $w_3(t)$ en (31) puede acotarse superiormente usando (30) como sigue

$$\begin{aligned} |w_3(t)| &\leq |w_3(0)| + c_0 \max\{|c_3|, |c_4|\} \int_0^t x(t) dt + c_5 \dot{q}_{d3} \\ &\leq |w_3(0)| + c_0 \max\{|c_3|, |c_4|\} \left[-\frac{x(0)}{c_1} (e^{-c_1 t} - 1) \right. \\ &\quad \left. - \frac{ac_2 \operatorname{sgn}(a \sin(\omega t))(c_1 \cos(\omega t) + \omega \sin(\omega t))}{\omega(c_1^2 + \omega^2)} \right] \\ &\quad + c_5 \dot{q}_{d3}, \end{aligned}$$

lo que demuestra que también $w_3(t)$ es una señal uniformemente últimamente acotada. \square

Proposición 2. *Las soluciones $\boldsymbol{\eta}(t) \in \mathbb{R}^3$ de la dinámica interna son uniformemente últimamente acotadas.*

Prueba: Esto surge como consecuencia de la Proposición 1, es decir, del hecho de que $y(t)$ converge exponencialmente, y de que la matriz V es invertible. Así, dado que el lado derecho de las ecuaciones (16)–(18) es un campo vectorial continuo, las soluciones $\boldsymbol{\eta}(t)$ de la dinámica interna (19) están uniformemente últimamente acotadas. \square

Proposición 3. *Las soluciones $\boldsymbol{x}(t) \in \mathbb{R}^4$ obtenidas de la dinámica del error (9) bajo la acción de control (12) son uniformemente últimamente acotadas.*

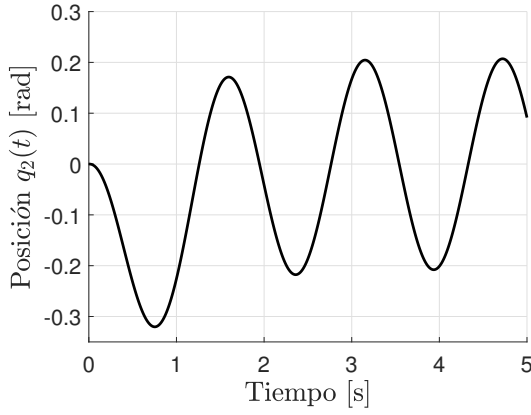
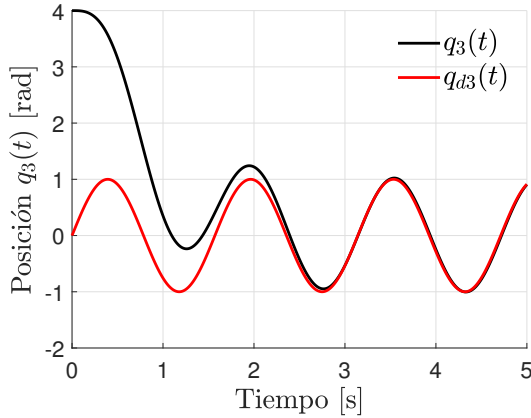
Prueba: Esto parte del hecho de que la transformación H es de rango completo, $y(t)$ converge exponencialmente y el hecho de que

$$\boldsymbol{x} = H^{-1} \begin{bmatrix} \boldsymbol{\eta}(t) \\ y(t) \end{bmatrix},$$

lo cual implica que

$$\|\boldsymbol{x}(t)\| \leq \|H^{-1}\| \left\| \begin{bmatrix} \boldsymbol{\eta}(t) \\ y(t) \end{bmatrix} \right\|.$$

Entonces, dado que $[\boldsymbol{\eta}(t)^T \ y(t)]$ es uniformemente últimamente acotada, entonces $\boldsymbol{x}(t)$ también lo es. \square


 Fig. 2. Evolución temporal de la posición $q_2(t)$.

 Fig. 3. Evolución temporal de la posición $q_3(t)$.

IV. RESULTADOS EN SIMULACIÓN

Se han utilizado los siguientes parámetros para el giroscopio de dos grados de libertad [16]–[17]:

$$\begin{aligned} I_{B_{zz}} &= 0.0179, & I_{C_{xx}} &= 0.0012, \\ I_{C_{yy}} &= 0.0008, & I_{C_{zz}} &= 0.0019, \\ I_{D_{xx}} &= 0.0056, & I_{D_{yy}} &= 0.0028, \\ I_{D_{zz}} &= 0.0028. \end{aligned}$$

Ésto resulta en

$$J_1 = 0.0021, \quad J_2 = 0.0226, \quad J_3 = 0.0036, \quad J_a = -0.0035.$$

Ademas, $P_1 = I_{D_{xx}} \dot{q}_1 = 0.0056 \text{ [Kg}\cdot\text{m}^2/\text{rad}] \times 78.53 \text{ [rad/s]} = 0.4397 \text{ [Kg}\cdot\text{m}^2/\text{s}]$.

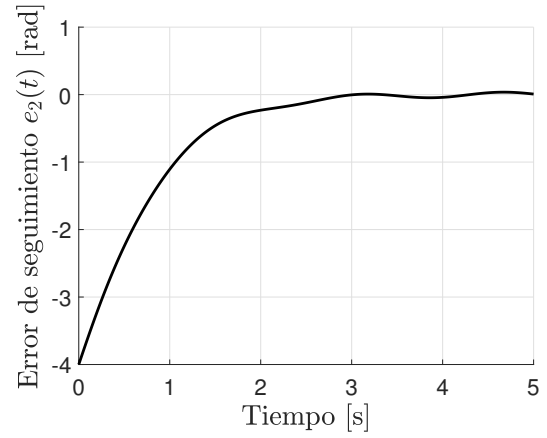
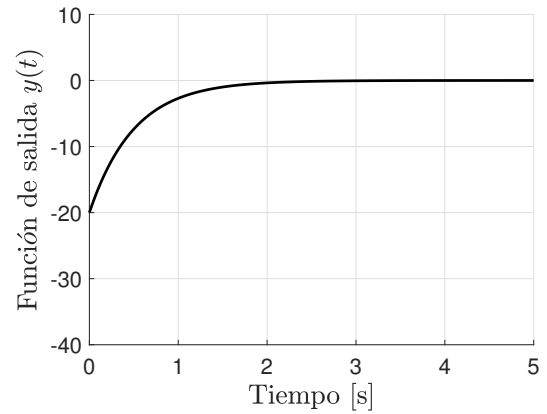
Las ganancias se escogieron como

$$\begin{aligned} \alpha_1 &= 1, & \alpha_2 &= 10, \\ \beta_1 &= 1, & \beta_2 &= 5, \end{aligned}$$

y $k_p = 2$. con estos valores, $J = \text{diag}\{-96.26, -2.02, 0\}$, entonces la condición C1 de la proposición 1 se satisface.

Como lo indica la condición C2 de la proposición 1, la simulación se ha llevado a cabo utilizando

$$q_{d3}(t) = a \sin(\omega t) \text{ [rad]},$$


 Fig. 4. Evolución temporal del error de seguimiento $e_2(t)$.

 Fig. 5. Evolución temporal de la función de salida $y(t)$.

donde $a = 1$ y $\omega = 4 \text{ [rad/s]}$.

Las condiciones iniciales para el giroscopio fueron

$$\begin{aligned} q_2(0) &= 0 \text{ [rad]}, & q_3(0) &= 4 \text{ [rad]}, \\ \dot{q}_2(0) &= 0 \text{ [rad/s]}, & \dot{q}_3(0) &= 0 \text{ [rad/s]}. \end{aligned}$$

Los resultados se presentan a continuación. La figura 2 muestra la evolución temporal de la posición $q_2(t)$ y la figura 3 describe la evolución temporal de la posición $q_3(t)$. Como se aprecia, las señales de posición y velocidad permanecen acotadas y la posición $q_3(t)$ converge a la posición deseada $q_{d3}(t)$.

El error del seguimiento de trayectorias $e_2(t)$ se muestra en la figura 4. Se observa que el error de seguimiento permanece acotado por debajo de 0.05 [rad] para todo tiempo $t \geq 3$. La figura 5 muestra la evolución temporal de la función de salida $y(t)$ propuesta en (10) la cual claramente converge exponencialmente a cero como lo predice la teoría. La figura 7 muestra la evolución temporal de la función $\rho(t)$, como se ve, la función $\rho(t)$ permanece acotada para todo t .

Otras pruebas de simulación mostraron que el error $e_2(t)$ puede ser disminuido arbitrariamente en estado estacionario, incrementando el valor de α_2 .

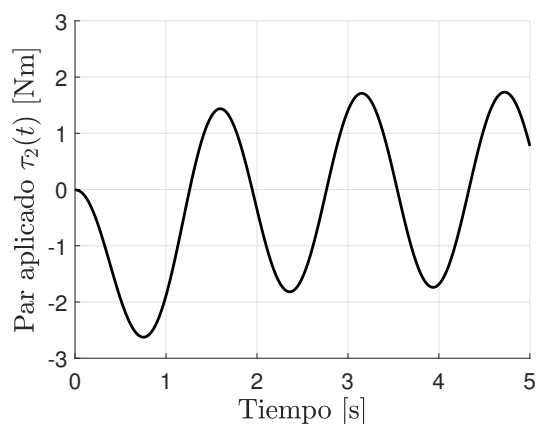


Fig. 6. Evolución temporal del par aplicado $\tau_2(t)$.

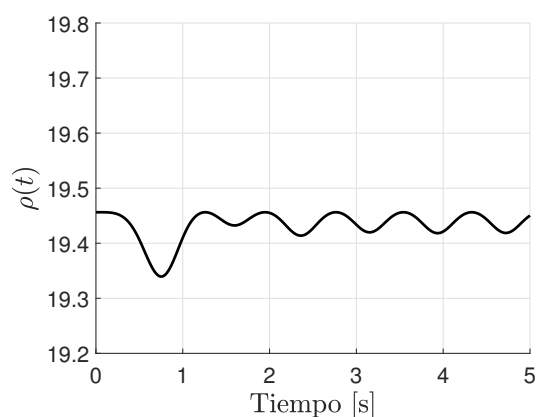


Fig. 7. Evolución temporal de la función $\rho(t)$.

Finalmente, el par aplicado $\tau_2(t)$ se describe en la figura 6, el cual aparenta ser una señal suave y acotada.

V. CONCLUSIONES

En este documento, se presentó un nuevo controlador para el seguimiento de trayectorias aplicado a un giroscopio subactuado. Similarmente a varios sistemas subactuados, la complejidad de este sistema reside en el acoplamiento no lineal de su modelo dinámico y en que se tiene una entrada de control para satisfacer el acotamiento del error de trayectorias.

La teoría es validada por medio de simulaciones. Investigaciones futuras incluirán la consideración de otros efectos no lineales en el sistema, tal como la dinámica de los actuadores [15].

REFERENCES

- [1] J. Moreno-Valenzuela y C. Aguilar-Avelar, *Motion Control of Underactuated Mechanical Systems*, Springer-Verlag, 2018
- [2] K. Majumder y B. M. Patre. "Sliding mode control for underactuated mechanical systems via nonlinear disturbance observer: stabilization of the rotational pendulum", *International Journal of Dynamics and Control*, pp. 1-10, 2018.
- [3] M. Reyhanoglu y J. van de Loo, "State feedback tracking of a non-holonomic control moment gyroscope", en memorias del 45th IEEE Conference on Decision and Control, pp. 6156-6161, 2006.

- [4] Y. Fujii, K. Yamada y T. Shima, "Analysis and suppression of limit cycle oscillation induced by CMG gimbal friction", *Transactions of the Japan Society for Aeronautical and Space Sciences, Aerospace Technology Japan*, vol. 8, no. ists27, pp. Pd_61-Pd_67, 2010.
- [5] L. Wu, Y. Guo, L. Wang, Z. Yu, y C. Zhong. "Experimental study on fuzzy PID control for DVSCMG based on decoupling feedback linearization", en memorias del 6th International Conference on Modelling, Identification & Control-ICMIC, pp. 155-159, 2017.
- [6] F. Farivar, M. A. Shoorehdeli, M. A. Nekoui, y M. Teshnehlab, "Synchronization of underactuated unknown heavy symmetric chaotic gyroscopes via optimal Gaussian radial basis adaptive variable structure control", *IEEE Transactions on Control Systems Technology*, vol. 21, no. 6, pp. 2374-2379, 2013.
- [7] G. He y Z. Geng, "Dynamics and robust control of an underactuated torsional vibratory gyroscope actuated by electrostatic actuator", *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 4, pp. 1725-1733, 2015.
- [8] C. Acar y A. Shkel, "Structural design and experimental characterization of torsional micromachined gyroscopes with non-resonant drive-mode," *J. Micromech. Microeng.*, vol. 14, pp. 1525, 2003.
- [9] C. Acar and A. Shkel, "Non-resonant micromachined gyroscopes with structural mode decoupling," *IEEE Sensors J.*, vol. 3, no. 4, pp. 497-506, 2003.
- [10] J. Montoya, V. Santibañez y J. Moreno. "Control adaptable de un giroscopio de 2 g.d.l. subactuado," en memorias del congreso mexicano de robótica 2017, Comrob XIX, Mazatlán, México, pp. 22-29, 2017.
- [11] N. Rana, P. Joag. *Classical Mechanics*, India, McGraw-Hill Education, 1991.
- [12] C. Aguilar-Avelar and J. Moreno-Valenzuela, "New feedback linearization-based control for arm trajectory tracking of the Furuta pendulum", *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 2, 638-648, 2016.
- [13] T. Otani, T. Urakubo, S. Maekawa, H. Tamaki and Y. Tada, "Position and attitude control of a spherical rolling robot equipped with a gyro", en memorias del 9th IEEE International Workshop on Advanced Motion Control, pp. 416-421, 2006.
- [14] H. K. Khalil, *Nonlinear Systems*, Upper Saddle River, Prentice-Hall, 2002.
- [15] J. Moreno-Valenzuela, Y. Quevedo-Pillado, R. Pérez-Aboytes and L. González-Hernández, "Lyapunov-Based Adaptive Control for the Permanent Magnet Synchronous Motor Driving a Robotic Load", *Journal of Circuits, Systems and Computers*, vol. 26, no. 11, pp. 1-31, paper no. 1750168, 2017.
- [16] http://www.quanser.com/products/3dof_gyroscope
- [17] J. Montoya, *Evaluación de esquemas adaptables de control aplicados a un giroscopio de 2 grados de libertad*, tesis de maestría en ciencias, Tecnológico Nacional de México-Instituto Tecnológico de la Laguna, Torreón, Coahuila, México, 2017

Dynamic tracking control for mechanical systems using only position measurements

Misael Hali Medina Barrera
 Sergio Gomez-Silva Flores
 Raúl Rascón Carmona
 Facultad de Ingeniería UABC
 Mexicali, Baja California

Email: misael.medina{sgomez56,raul.rascon}@uabc.edu.mx

Andrés Calvillo Tellez
 Instituto Politécnico Nacional - CITEDI
 Tijuana, Baja California
 Email: calvillo@citedi.mx

Abstract—In this paper, a dynamic feedback control law based in the “Super-twisting” algorithm is presented. Dynamic compensation is added to the proposed algorithm, in order to use position feedback only. In some mechanical applications, an observer or a differentiator design is required for control purposes when the whole state space is not available for measurement, but not in this proposal where the observer is obviated, this constitutes the main contribution of the present approach. Indeed, these new proposals do not require to measure or estimate another signal but the position of the mechanical system to solve the tracking problem. In the stability analysis, homogeneity property is studied as a step to show finite time stability in a near future.

I. INTRODUCTION

It is commonly recognized that control of mechanical systems has been subject of interest as a result of its industrial applications, some previous works are [1], [2], [3], [4]. Typical control issues are swing up, stabilization, tracking and regulation, of which the last one is a specific instance of the tracking problem, some recent works related to tracking in mechanical systems can be found on [5], [6], [7]. In most works associated with tracking control design for mechanical systems, the velocity state needs to be available for feedback, this could be a drawback if there is not an available velocity sensor, this situation can be solved implementing observers or differentiators, see [8], [9], [10], [11].

the previous works presented in [12], [13] are examples of tracking control in mechanical systems without velocity measurements, where in these works are addressed systems with Euler-Lagrange dynamics.

Persuaded for the aforementioned background, This paper builds up a design of a tracking controller for a class of second order mechanical systems, using only positions measurements in order to accomplish the control objective. the proposed solution consist of a possible finite time stable controller, the stability proof is made using the homogeneity principle [14], [15], although in order to prove finite time stability only remains to prove firstly global asymptotic stability of the closed-loop system, which it is intended to be publish elsewhere.

The following sections structure is: in Section II the statement of the problem in second order dynamical systems

is described, which can be applied in mechanical systems of one degree of freedom, the structure of the controller is also showed on this section. The continuous finite time stable algorithm is presented in Section III and its homogenous stability study. In Section IV presents simulation results comparing the proposed algorithm with other well-known controllers. Finally Section V presents some conclusions.

II. PROBLEM STATEMENT

The dynamics of a mechanical system are governed by the following state space equations

$$\begin{aligned}\dot{x} &= y \\ \dot{y} &= f(x, t) + g(x, t)u + w(x, y, t).\end{aligned}\quad (1)$$

The variable x stands for the position and y is the velocity of the mechanical system. The actual value of $w(x, t) \in \mathcal{L}_\infty$ is unknown, but it is bounded by a known constant M , i.e., $\sup_{t \geq 0} |w(x, t)| \leq M$. The solutions of all systems of differential equations are understood in the Filippov’s sense [16]. For system (1) the following control design is as follows

$$u = -g(x, t)^{-1} (f(x, t) - \tau - \ddot{x}_d) \quad (2)$$

where the first term is a compensation and τ is the proposed algorithm, note that (2) does not used the velocity variable y as feedback. Substituting (2) in (1) the remaining closed-loop system stands as follows

$$\begin{aligned}\dot{x} &= y \\ \dot{y} &= \tau + \ddot{x}_d + w(x, y, t).\end{aligned}\quad (3)$$

We rewrite system (3) in function of the errors, where $e_1 = x - x_d$ and $e_2 = \dot{x} - \dot{x}_d$, where x_d is the desired trajectory which is C^k , for a sufficiently large k

$$\begin{aligned}\dot{e}_1 &= e_2 \\ \dot{e}_2 &= \tau + w(x, y, t).\end{aligned}\quad (4)$$

III. DYNAMIC ALGORITHM

The proposed dynamic algorithm to stabilize the system (4) is

$$\begin{aligned}\tau &= -k_1[e_1]^{\frac{\alpha}{2-\alpha}} - k_2[e_2]^\alpha \\ \dot{e}_3 &= -k_3[e_1]^{\frac{\alpha}{2-\alpha}} - k_4[e_3]^\alpha\end{aligned}\quad (5)$$

considering the parameters $k_1, k_2, k_3, k_4 > 0$, and $\alpha \in (0, 1)$, note that (5) does not involved the velocity error e_2 in its design, so that the control structure (2) and (5) does not used velocity measurements y . Substituting (5) in (4) the closed-loop system is

$$\begin{aligned} \dot{e}_1 &= e_2 \\ \dot{e}_2 &= -k_1[e_1]^{\frac{\alpha}{2-\alpha}} - k_2[e_3]^\alpha + w(x, y, t) \\ \dot{e}_3 &= -k_3[e_1]^{\frac{\alpha}{2-\alpha}} - k_4[e_3]^\alpha. \end{aligned} \quad (6)$$

A. Stability of the closed-loop system (Nominal case)

Considering $w(x, y, t) = 0$, in this case the system (6) has an homogeneous vector field of degree $q = -1$ with respect to dilation $r = (\frac{2-\alpha}{1-\alpha}, \frac{1}{1-\alpha}, \frac{1}{1-\alpha})$ so that by the homogeneity principle presented in [17], Theorem 4.2, the global finite time stability of this system can be guaranteed if global asymptotic stability is first proven, which is our research goal in a near future.

IV. NUMERICAL SIMULATIONS

In order to compare the performance of the proposed algorithm with other controllers: PID and first order sliding mode, a test is made analyzing the response in a double integrator system. The initial conditions and parameters used are shown in Table I.

TABLE I
SIMULATION PARAMETERS

| Plant parameters | |
|--------------------|--------------------|
| Notation | Value |
| $x_1(0)$ | 0 |
| $x_2(0)$ | 0 |
| $x_d(t)$ | $0.1 \sin(t)$ |
| $w(x, y, t)$ | $2 \sin(2t) + x_1$ |
| Dynamic controller | |
| k_1 | 8 |
| k_2 | 3 |
| k_3 | 8 |
| k_4 | 3 |
| α | 1/12 |
| PID | |
| k_p | 800 |
| k_d | 80 |
| k_i | 8 |
| FOSM | |
| β | 5 |

The PID controller is operated as

$$\tau = -k_p e_1 - k_i \int_{t_0}^{t_1} e_1 dt - k_d \frac{de_1}{dt} + \ddot{x}_d$$

using the parameters showed on the table aforementioned. The first order sliding mode algorithm used is

$$\tau = -e_2 - \beta \operatorname{sgn}(s) + \ddot{x}_d$$

where $e_2 = de_1/dt$, with $\beta = 5$, the first order sliding mode can absorb perturbations and uncertainties $w(t, x, y)$ which are

bounded by $|w(t, x, y)| \leq D$, and the condition $\beta > D$ must be hold for stability purposes.

In Figure 1 can be seen the performance comparison of the three controllers tracking a reference, it can be observed that the proposed controller has a bigger transient amplitude than the other two controllers, in Figure 2 it can be appreciated that the proposed controller arrives to the reference in a shorter time than the other two controller, moreover notice that the PID controller can not achieve a zero steady state error in the nominal stage. In Figure 3 are shown the velocities x_2 , where the proposed controller has an aggressive transient compared with the other controllers. In Figure 4 are shown the control signals, where the proposed controller renders an acceptable amplitude for a future implementation in a mechanical system. In Figure 5 is shown the phase portrait, what is interesting in this Figure is that the proposed controller has a behavior alike to the twisting controller. Finally Figure 6 offers the energy of the control signal u , although the PID controller has a lower energy than the proposed controller, it can not achieve a zero steady state error during the nominal stage.

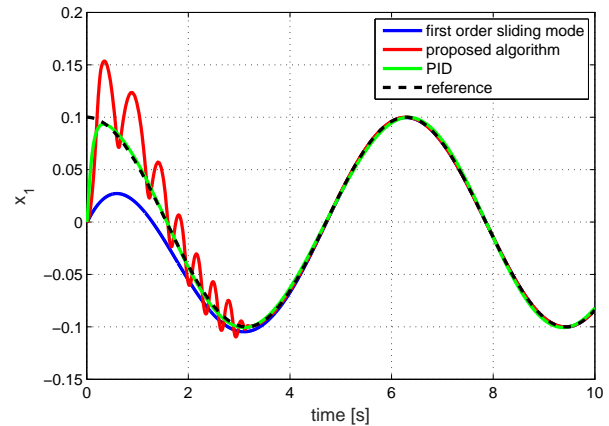


Fig. 1. Trajectory Tracking (simulation)

V. CONCLUSION

The proposed controller is still under development with the purpose to show finite time stability to the tracking reference, also it is necessary to give some criteria about the gain parameters in order to compensate external disturbances and uncertainties which may affect the closed-loop system. It is important to remark that the proposed controller only needs position measurements to achieve zero steady state error in the tracking control problem. The present controller design can be extended to Lagrangian systems. Throughout a comparison simulation, it is shown that the proposed controller renders a good performance compared with a well known controllers: first order sliding mode control and PID control.

REFERENCES

- [1] R. E. Bellman, *Adaptive control processes: a guided tour*. Princeton university press, 2015.

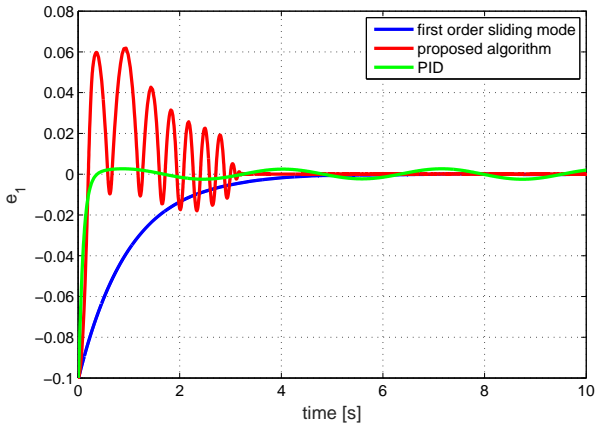


Fig. 2. position error $e_1 = x_1 - x_d$ (simulation)

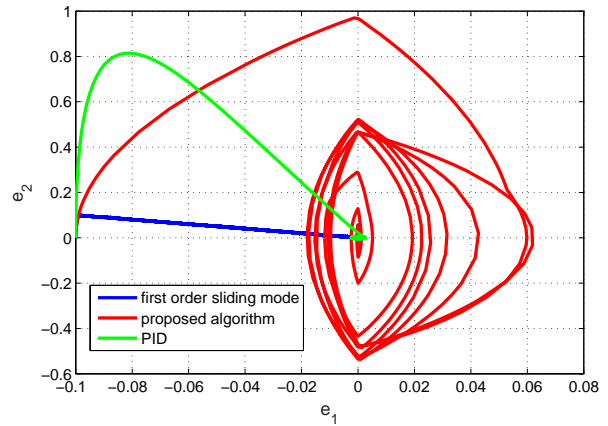


Fig. 5. Phase Portrait (simulation)

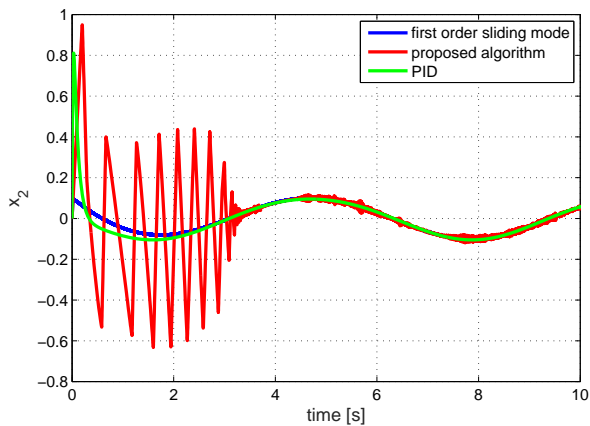


Fig. 3. Velocity comparison (simulation)

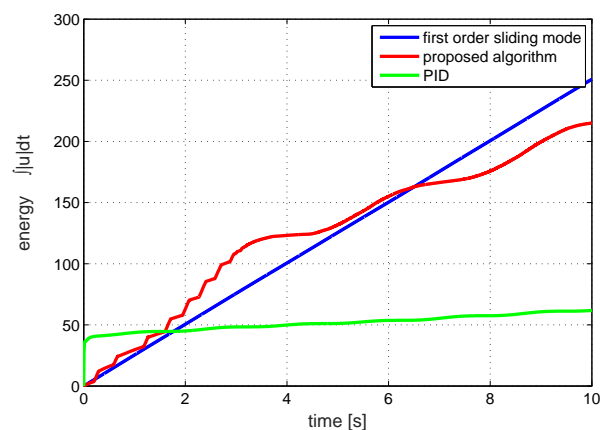


Fig. 6. energy (simulation)

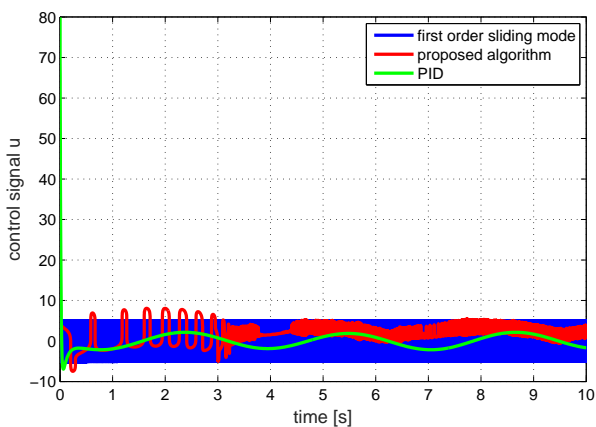


Fig. 4. Control Signal (simulation)

[2] W. He, S. S. Ge, B. V. E. How, and Y. S. Choo, *Dynamics and control of mechanical systems in offshore engineering*. Springer, 2014.

[3] J. Huang, C. Wen, W. Wang, and Y.-D. Song, "Adaptive finite-time consensus control of a group of uncertain nonlinear mechanical systems," *Automatica*, vol. 51, pp. 292–301, 2015.

[4] R. Sakthivel, A. Arunkumar, and K. Mathiyalagan, "Robust sampled-data h control for mechanical systems," *Complexity*, vol. 20, no. 4, pp. 19–29, 2015.

[5] C. Huang, F. Li, and Z. Jin, "Maximum power point tracking strategy for large-scale wind generation systems considering wind turbine dynamics," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 4, pp. 2530–2539, 2015.

[6] G. C. Lazaroiu, M. Longo, M. Roscia, and M. Pagano, "Comparative analysis of fixed and sun tracking low power pv systems considering energy consumption," *Energy Conversion and Management*, vol. 92, pp. 143–148, 2015.

[7] Q. Zhou, H. Li, and P. Shi, "Decentralized adaptive fuzzy tracking control for robot finger dynamics," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 3, pp. 501–510, 2015.

[8] G. Bartolini, A. Estrada, and E. Punta, "Observation and output adaptive tracking for a class of nonlinear nonminimum phase systems," *International Journal of Control*, no. just-accepted, pp. 1–20, 2016.

[9] A. Gutierrez-Giles and M. A. Arteaga-Prez, "{GPI} based velocity/force observer design for robot manipulators," *{ISA} Transactions*, vol. 53, no. 4, pp. 929 – 938, 2014, disturbance Estimation and Mitigation.

[10] D. Rosas, J. Alvarez, P. Rosas, and R. Rascon, "Robust observer for a class of nonlinear siso dynamical systems," *Mathematical Problems in*

- Engineering*, vol. 2016, 2016.
- [11] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant, *Sliding mode control and observation*. Springer, 2014.
 - [12] A. Loria, "Global tracking control of one degree of freedom euler-lagrange systems without velocity measurements," *IFAC Proceedings Volumes*, vol. 29, no. 1, pp. 2295–2300, 1996.
 - [13] A. Loria, "Observers are unnecessary for output-feedback control of lagrangian systems," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, pp. 905–920, 2016.
 - [14] Y. Orlov, "Finite time stability and robust control synthesis of uncertain switched systems," *SIAM Journal on Control and Optimization*, vol. 43, no. 4, pp. 1253–1271, 2004.
 - [15] Y. Orlov, Y. Aoustin, and C. Chevallereau, "Finite time stabilization of a perturbed double integrator part I: Continuous sliding mode-based output feedback synthesis," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 614–618, 2011.
 - [16] A. F. Filippov, *Differential Equations with Discontinuous Right-Hand sides*. Dordrecht, The Netherlands: Kluwer, 1988.
 - [17] Y. Orlov, *Discontinuous systems – Lyapunov analysis and robust synthesis under uncertainty conditions*. London: Springer, 2009.

Modeling of a Speed Sensor and its Application in a Sensorless Control System

Abraham Guerrero G.

Instituto Tecnológico de Tijuana
Tijuana, Baja California

Email: http://aerospace.abraham@gmail.com

Paul J. Campos

Instituto Tecnológico de Tijuana
Tijuana, Baja California

Email: paul.campos@tectijuana.edu.mx

Luis Muñoz

Instituto Tecnológico de Tijuana
Tijuana, Baja California

Email: lmunoz@tectijuana.edu.mx

Leonardo Trujillo

Instituto Tecnológico de Tijuana
Tijuana, Baja California

Email: leonardo.trujillo@tectijuana.edu.mx

Luis N. Coria

Instituto Tecnológico de Tijuana
Tijuana, Baja California

Email: luis.coria@tectijuana.edu.mx

Abstract—Permanent magnet synchronous motors (PMSM) have gained popularity in recent years, one important feature is that they use 30% less energy than a conventional system using DC traction motors. Also, they can be considered small in comparison to the pair-motor that they develop. Several control strategies have been applied to this type of machines, specifically the so-called Sensorless Control. This consists in the elimination of sensors that measure speed or position of the rotor, which are necessary for conventional speed or position control of the machine in closed loop.

This paper presents an alternative solution that can be used to solve the Sensorless vector control in PMSMs. The proposal is based on the ability of Genetic Programming (GP) to generate a model from the system dynamics and its desired behavior. The generated models are based on the electrical variables such as stator currents and voltages that are considered to be the only available for feedback. In this case, the training data is obtained from the closed-loop control system analysis when a positioning-sensor is used. A new speed estimator is then evolved using the Eureqa software by Nutonian[®]. Furthermore, the proposed solution is validated comparing its closed-loop behavior with that obtained by the mechanical sensor, as well as, a comparison with standard techniques is done.

I. INTRODUCTION

Motors are mechanical devices that allow to transduce energy and convert it into mechanical energy. There is a great variety of motor, among which are those known as Permanent Magnet Synchronous Motors (PMSMs) which is constituted by permanent neodymium magnets. PMSMs have attracted attention in the last few years, due to their high efficiency, fast dynamic response, high power factor, high torque with respect to weight, simple construction, easy maintenance and longer life compared to direct current (DC) and induction motors [2],[1]. Due to these advantages, these type of motors are commonly used in systems that require high speeds, such as home appliances, tools, turbines [19]. Regarding the control of this type of machines, it is necessary to know precise information about the variable to control, either speed or position, to ensure the desired operation. To achieve the control objective, mechanical or optical sensors are usually used. Achieving this objective without a sensor is a more ambitious

task. There are strategies that can be applied to solve this problem, among which are state observers [7], [18], [2], neural networks [9], fuzzy systems [8] and others.

On the other hand, GP is a research area particularly new and emerging, which has been applied to solve various problems in different areas of knowledge. In addition, it has been applied with success in real-world applications such as Optic Character Recognition (OCR) [4], signal processing, generation of dynamic models [5], [3]. Also, it is able to solve complex problems and modeling [6]. Considering the aforementioned, in this work is analyzed and apply the viability of GP as a different proposal to solve the sensorless control problem in PMSMs.

The remainder of this paper is organized as follows. Section 2 presents the dynamic model of a PMSM in a two-phase rotating frame of reference. Details of the method of solution, starting with an overview of GP followed by the design of generated model are presented in Section 3. Numerical simulation and results are shown in Section 4. Finally, a short discussion, conclusions and future work are given in section 5.

II. DYNAMIC MODEL OF A PMSM

PMSMs are categorized as three-phase AC machines, where their three-phase voltage equations are considered nonlinear and time varying. Therefore, Field Oriented Control (FOC) is applied to simplify their dynamical analysis and to solve control problems, making it possible to control an AC motor in a similar way that a separately excited DC motor is controlled. Due that FOC is based on a couple of transformations, there is required a precise rotor position knowledge to guarantee a correct transformation of a 3-phase time-variant AC machine into a 2-phase time-invariant machine. Moreover, this transformation does not modify the speed operational range of the machine [1], [2].

Hence and according with [1], [2], [9], [18], the state-space representation of a surface mounted PMSM with no-saliency

in rotating reference frame d - q is given as follows

$$\begin{aligned}
 \frac{di_d}{dt} &= \frac{1}{L_d}(L_q i_q \omega - R i_d + V_d), \\
 \frac{di_q}{dt} &= \frac{1}{L_q}(-(L_d i_d + \varphi)\omega - R i_q + V_q), \\
 m &= \frac{3}{2}\rho(L_d - L_q)i_d i_q + \frac{3}{2}\rho\varphi i_q, \\
 \frac{d\omega}{dt} &= \frac{1}{J}(m - m_r - F\omega);
 \end{aligned} \tag{1}$$

where i_d , i_q and V_d , V_q are stator currents [A] and voltages [V] respectively, in a rotating reference frame d - q ; ω and θ are the rotor speed [rad/s] and position [rad]; while L_d and L_q , R , φ , m , m_r , J and F are the physical parameters of the motor: winding inductances [H], stator winding resistance [Ω], permanent magnet flux [Wb], constant torque [Nm/A], load torque, rotor moment of inertia [Kgm²], viscous friction coefficient [Nm/rad], respectively, while ρ is the number of pole pairs, in the rotating reference frame d - q .

Assuming that the machine has no saliency and perfect electric symmetry, i.e., $L_d = L_q = L$, then m is reduced to $m = \frac{3}{2}\rho\varphi i_q$, and considering $m_r = 0$, the system represented by (1) can be expressed as

$$\begin{aligned}
 \frac{di_d}{dt} &= i_q \omega - \frac{R}{L} i_d + \frac{V_d}{L}, \\
 \frac{di_q}{dt} &= -(i_d + \frac{\varphi}{L})\omega - \frac{R}{L} i_q + \frac{V_q}{L}, \\
 \frac{d\omega}{dt} &= \frac{3\rho\varphi}{2J} i_q - \frac{F}{J} \omega.
 \end{aligned} \tag{2}$$

III. METHOD

A. Genetic Programming

GP was proposed by Koza [10], which is inspired in natural selection and genetics. It belongs to the class of Evolutionary Algorithms (EA). GP has been used to solve many difficult tasks, in particular symbolic regression problems. Traditionally, GP individuals are represented as tree diagrams, see Figure 1, where each node of the tree has an operator function and each terminal node has a variable or constant value, turning the tree into a mathematical model that can be summarized into a comprehensive mathematical expression. Figure (1) represents the model $((3+g)*p)-(8+p)$. Moreover, it formulates a solution where the only dependencies are the variables g and p . Therefore, GP favors the use of mathematical expressions that can be analyzed and interpreted by the user. It is worth to mention that it is one of the most powerful algorithms of computing [13], [14].

Additionally, it has been proven with numerical methods that there are many difficulties with respect to the accuracy and convergence of a solution. In such circumstances, GP is expected to be a better alternative, because it not only finds a solution, also provides many solutions whose structure can be evaluated by the designer. Therefore, depending on the precision and simplicity of the implementation, the choice of a solution is up to the user specifications or the problem

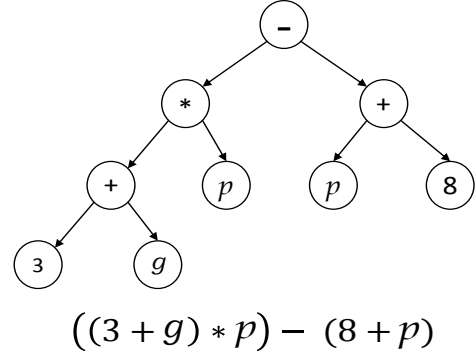


Fig. 1. Function represented as a tree structure

requirements[15]. Furthermore, it can be of interest in controllers design since it offers advantages over other known methods, due to its ease use and its little need for details about the inputs and dynamical behavior of a system variables. Also, since GP has the ability to solve problems based on incomplete, imprecise or ambiguous information, it is thought to be a good method to solve a control problem.

Additionally, GP search for the objective through heuristic methods, any other option will have a toll on time, resources and interpretability. In this work, GP was applied, using the Eureka software [17], to solve a speed sensorless control problem.

B. Numerical Experiment Design

FOC is applied to PMSMs when a better performance, reduction of the machine complexity and an easier control task are desired. However, to apply FOC is require an either a speed or a rotor position sensor. Figure 2, it is presented the classical scheme of a FOC system, which consists on two cascaded control loops, one for speed control and the other for current controlling. These control loops are constituted by three integral proportional (PI) controllers, where the PI controller located in the external loop is used to determine the reference current corresponding to the desired speed. The other two PIs, located in the inner loop, are used to ensure the transformation of the machine, to control the applied current in the $d - q$ axes and force the synchronization of the rotor flow at the desired speed. However, the overall closed-loop Sensorless FOC system is a more complex structure composed by coordinate transformations, a three-phase power inverter, a three-phase Permanent Magnet Synchronous Motor model MBE.300.E500 where its corresponding physical parameters are shown in Table I.

In the other hand, to implement a GP algorithm a training data is required. This data is collected by emulating the PMSM and its sensor using an algorithm that emulates the overall closed-loop system behavior. This emulator is developed in MATLAB/Simulink 2009a®, using the libraries

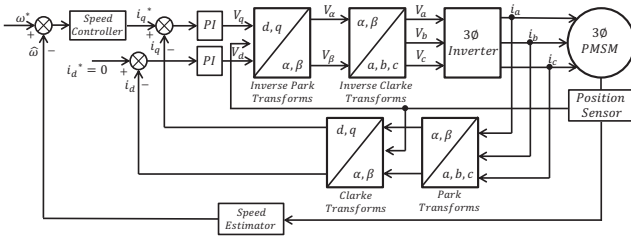


Fig. 2. Classic FOC system for a PMSM.

 TABLE I
 MOTOR PARAMETERS AND ELECTRICAL SPECIFICATIONS

| Symbols | Parameters | Values | Units |
|----------|------------------------------|------------------------|------------------|
| R | Stator Resistance | 4.3 | ohm |
| L | Stator Inductance | 359×10^{-3} | H |
| K_M | Rotor torque constant | 24.5×10^{-3} | Nm/A |
| p | Pole-pairs | 1 | |
| F | Viscous friction coefficient | 0.157×10^{-3} | Nms |
| J | Rotor inertia | 1.1×10^{-6} | kgm ² |
| P_r | Rated power | 15.7 | W |
| V_r | Rated voltage | 36 | V |
| Ω | Max. permissible speed | 15×10^3 | rpm |
| T_L | Max. cont. torque | 30×10^{-3} | Nm |

for motion control developed by Technosoft® for the DSP TMS320F28335. This platform allows to design each control block as well as to emulate the complete control system and at the same time analyze its performance by numerical simulations. Hence, the extracted data consists of a matrix of dimensions $m \times n$, with $m = 10000$ and $n = 5$, where each dimension corresponds to the available electrical variables behaviors, such as i_d , i_q , V_d , V_q and ω^* . Once the training data is acquired, it is loaded to Eureka as an individual column vector that it is used to generate models correlating the electrical information with the machine's speed. It is important to mention that no additional information or system's prior models are use. Moreover, the algorithm not generates only one solution, it generates a group of solutions that can be considered as good options, in terms of error and fitness obtained. Then, the best models (those with lower error) found are used to evaluate their performance using the closed-loop system of Figure (3). After that, the best model that behaves with the desired performance is chosen as the optimal one and it is used to replace the speed estimator. Therefore, the selected algorithm equation that emulates the desired speed sensor behavior is:

$$\omega = k_1 V_q - k_2 i_q, \quad (3)$$

where k_1 and k_2 are constants with 5.2×10^{-3} and 3×10^{-6} , respectively.

IV. RESULTS

In order to check the correct operation of the obtained model, the structure shown in Figure (3) is considered. In this case, the speed sensor, depicted as speed estimator in

Figure (2), is substituted by the generated model (3) and used to feedback the outer control-loop, corresponding to speed controlling. As can be seen in Figure (3), the proposed estimator named as GP-Speed-Estimator is fed by the electrical variables, in this specific case for i_q - current and V_q - voltage, corresponding to the quadrature axis in the stationary reference frame.

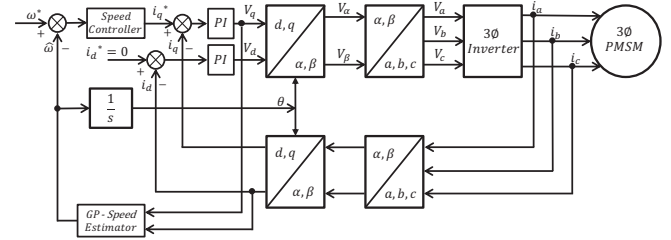
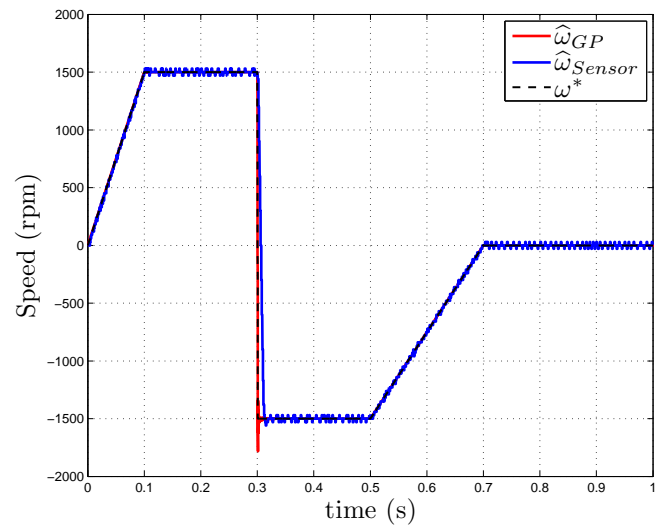


Fig. 3. FOC system with GP speed estimator.

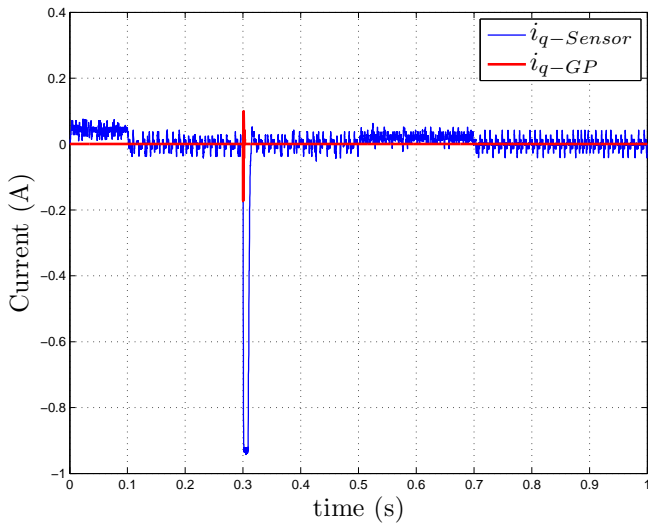

 Fig. 4. Experimental results using the GP approximate sensor ($\hat{\omega}_{GP}$), mechanical speed sensor ($\hat{\omega}_{Sensor}$) and the desired speed (ω^*).

The performance is evaluated by using a time-varying speed signal that considers interesting Sensorless control cases such as zero crossing, reversal speed and zero speed, see Figure (4). The simulation results, shown in Figure (4), present a performance comparison between both, the mechanical Speed-Estimator ($\hat{\omega}_{Sensor}$) and GP-Speed-Estimator ($\hat{\omega}_{GP}$). According with the results, it is clearly seen that with the GP-Speed-Estimator, not only a better performance can be appreciated, also a fast dynamic response even in cases of sudden changes of speed and reversal speed. Therefore, it can be considered an appropriate solution for this type of problems.

Furthermore, Figure (5) and Figure (6) depicts the corresponding waveform of the measured stator currents (i_d , i_q) obtained with the mechanical Speed-Estimator (i_d - sensor, i_q - sensor) and the GP-Speed-Estimator (i_d - GP, i_q - GP). As can be seen, the speed and current waveforms demonstrate

TABLE II
 AVERAGE ABSOLUTE ERROR

| Average Absolute Error (AAE-rpm) | |
|----------------------------------|---------|
| GP | 0.4289 |
| Sensor | 27.2815 |


 Fig. 5. Waveforms of the stator currents on the d and q axes, Sensor current ($i_{q-Sensor}$) against estimated current using GP (i_{q-GP}).

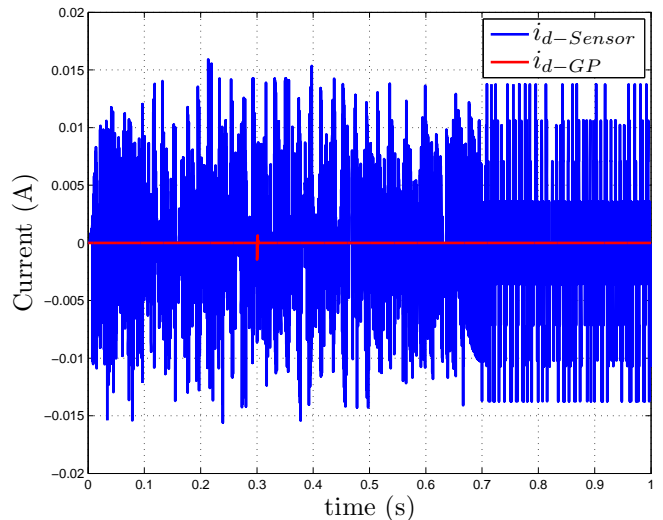
the good performance of the proposal, where not only a faster speed tracking is obtained when the speed estimator is replaced by the GP-Speed-Estimator, also a less stator currents consumption is needed to achieve the control objective. Moreover, a better i_d - current controlling is obtained, as it is required by FOC to guarantee a right transformation.

Finally, the aforementioned statements are supported by applying the Average Absolute Error (AAE) analysis, given in Table (II). As can be seen in Table (II), the AAE of both estimator are compared. However, the AAE given by GP is lower than that obtained by the mechanical Speed-Estimator, i.e., the model given in equation (3), used to replace this sensor, is a good option in the solution of this problem.

V. CONCLUSION

In this work a model to obtain the velocity of a PMSM was proposed and constructed with the help of Eureqa®[5], based on the data obtained of the behavior of a mechanical speed sensor. The proposed model is evaluated and compared with the performance of the mechanical sensor in closed-loop mode. Moreover, it is demonstrated that a real sensor can be replaced by a mathematical model, even a better performance and less complex control system can be obtained.

Finally, future works of this research will focus in implementing the proposal in a real experimental platform, validate by mathematical analysis the generated models and try to find


 Fig. 6. Waveforms of the stator currents on the d and q axes, Sensor current ($i_{d-Sensor}$) against estimated current using GP (i_{d-GP}).

not only models with good performances also generate optimal models that permits better energy consumption.

ACKNOWLEDGMENT

This work was supported by Proyecto Fronteras de la Ciencia CONACYT FC-2015-2/944, "Aprendizaje evolutivo a gran escala" and TecNM project 6579.18-P, "Control Sensorless para el péndulo de Furuta".

REFERENCES

- [1] Krause, P.C., Wasynczuk, O., and Sudhoff, S.D., *Analysis of electric machinery and drive systems*, IEEE Press, 2002.
- [2] Ahmad, and M., *High Performance AC Drives: Modelling, Analysis and Control*, Springer-Verlag, 2010.
- [3] William B. Langdon and Adil Qureshi, *Genetic Programming - Computers using Natural Selection to generate programs*, Dept of Computer Science, 2011.
- [4] D. Andre. Learning and upgrading rules for an OCR system using genetic programming. *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*, 462-467, 1994.
- [5] Veena Patil-Shinde and Sanjeev S. Tambe. Genetic programming based models for prediction of vapor-liquid equilibrium. *Calphad*, 68 - 80, vol. 60, 2018. ISSN 0364-5916.
- [6] Oluwaseun K. Oyebo, Josiah A. Adeyemo, *Genetic Programming: Principles, Applications and Opportunities for Hydrological Modelling*, World Academy of Science, Engineering and Technology International Journal of Environmental and Ecological Engineering, 2014.
- [7] Paul J. Campos and Luis N. Coria and Leonardo Trujillo, "Nonlinear speed sensorless control of a surface-mounted PMSM based on a Thau observer," *Springer-Verlag, Electrical Engineering*, 2016.
- [8] Bilal Gümüş and Mehmet Özdemir, "Sensorless vector control of a Permanent magnet synchronous motor with fuzzy logic observer," *Springer-Verlag, Electrical Engineering*, vol. 88, no. 5, pp. 395-402, 2006.
- [9] Accetta, A., Cirrincione, M., Pucci, M.: TLS EXIN based neuronal sensorless control of a high dynamic PMSM. *Control Engineering Practice*, vol. 20, no. 7, pp. 725-73, 2012.
- [10] J. R. Koza, *Genetic programming: on the programming of computers by means of natural selection*, Cambridge, Massachusetts, London, England: The MIT Press, 1992.
- [11] M. J. Mohamad, *A Proposed Genetic Programming Applied to Controller Design and System Identification*, Ph.D. Thesis, University of Technology, Baghdad, 2008.

- [12] M. Morawiec, *The Adaptive Backstepping Control of Permanent Magnet Synchronous Motor Supplied by Current Source Inverter*, IEEE Transactions on Industrial Informatics, 2013.
- [13] I. G. Tsoulos and I. E. Lagaris, *Solving Differential Equations with Genetic Programming*, Genetic Programming and Evolvable Machines, 2006.
- [14] J. Imae and J. Takahashi, *A Design Method for Nonlinear Control Systems via Hamilton Jacobi Isaacs Equations: A Genetic Programming Approach*, Proceedings of 38th conference on Decision and Control, 1999.
- [15] Rami A. Maher, Mohamed J. Mohamed, *An Enhanced Genetic Programming Algorithm for Optimal Controller Design*, Scientific Research Intelligent Control and Automation, 2012.
- [16] Perla Sarahi Juarez Smith, *Programacion Genetica Distribuida y el modelo evospace*, Tijuana, Mexico, 2007.
- [17] Schmidt M, Lipson H., <http://www.nutonian.com>, Ciudad, Pais: Editorial, 2015.
- [18] Tomei, P. and Verrelli, C. *Observer-based speed tracking control for sensorless permanent magnet synchronous motors with unknown load torque.*, IEEE Transactions on Automatic Control, 2011.
- [19] M. Kiuchi and T. Ohnishi and H. Hagiwara and Y. Yasuda *V/f control of permanent magnet synchronous motors suitable for home appliances by DC-link peak current control method*, The 2010 International Power Electronics Conference - ECCE ASIA -, 2010.

Sistema de control de camas ortopédicas para pacientes con cuadriplejía por medio de voz

V.H.García O., B.R. Jorge L.

Instituto Politécnico Nacional, ESCOM
Dpto. de Ing. en Sistemas Computacionales
Av. Juan de Dios Batiz s/n, Lindavista, 07738
Ciudad de México
vgarciaortega@yahoo.com.mx

R. Hernández T.

Instituto Politécnico Nacional, UPIITA
Av IPN 2580, Barrio la Laguna Ticoman, 07340
Ciudad de México.
rhtovar@ipn.mx

Resumen—El presente trabajo presenta el desarrollo de un sistema de reconocimiento de palabras aisladas para el control de camas ortopédicas articuladas de hospital, donde se encuentren aquellos pacientes que presenten cuadros de cuadriplejía que es un trastorno no progresivo que afecta la psicomotricidad y por tanto impide el movimiento de las extremidades. El sistema reconoce tres palabras aisladas para identificar un segmento de la cama ortopédica: cabeza, pelvis y pie; y tres palabras aisladas de acción sobre el segmento: subir, bajar y alto. El sistema reconoce la secuencia de estados de las palabras aisladas con la que se puede manipular las partes móviles de los modelos estándar de las camas ortopédicas articuladas. Los parámetros de voz que se analizan en cada palabra aislada son los Coeficientes Cepstrales en Frecuencia en escala de Mel (MFCC). Estos parámetros son reconocidos mediante un clasificador basado en el k-ésimo Vecino más Cercano (K-NN). El sistema está implementado en un sistema embebido basado en un procesador ARM Cortex A53 de cuatro núcleos. Este sistema embebido tiene implementado un servidor que utiliza una arquitectura cliente-servidor usando sockets TCP con lo que se tiene comunicación en red. En el sistema embebido se usa una versión personalizada de kernel mínima para linux creada con el proyecto Yocto con el sistema de arranque para sistema D. Los algoritmos se paralelizan usando técnicas de Cómputo de Alto Desempeño (HPC) en el sistema embebido mediante procesos e hilos para su ejecución.

Palabras clave— Sistema embebido, MFCC, procesamiento de voz, cama ortopédica.

I. INTRODUCCION

En México, el 6.6% de la población (7.2 millones de personas) reportan tener mucha dificultad o no poder hacer alguna de las actividades básicas por las que se indaga, es decir, sufren algún tipo de discapacidad [1].

Las personas con una capacidad diferente enfrentan problemas y retos singulares, de entre las cuales aquellas con problemas de movilidad, se enfrentan a imposibilidad para desplazarse, caminar y manipular su entorno, teniendo complicaciones al momento de realizar tareas cotidianas [2]. Dentro de este grupo de personas, existen aquellas con capacidad diferente sobrevenida, es decir, que es adquirida por un accidente, un ejemplo de este caso es la cuadriplejía, la cual traerá como consecuencia una adaptación con ayuda de allegados, tendiendo a permanecer en una cama ortopédica en

su estancia dentro del hospital y posteriormente en casa [3]. La cuadriplejía es calculada de acuerdo a la lesión medular de acuerdo a las capacidades funcionales correspondientes a su nivel de lesión, una de estas es el habla en pacientes que no requieren el uso de respirador [4].

Actualmente los sistemas de control han comenzado a ser utilizados en el ámbito médico, tanto para el apoyo del personal de los centros médicos, así como herramientas de ayuda para las personas con discapacidad. Algunos sistemas de control patentados utilizan el habla del usuario para la manipulación de camas para hospitales [5-7] mediante la extracción de parámetros característicos de la señal de voz.

Los parámetros característicos de voz que han dominado el reconocimiento del habla han sido: Coeficientes Cepstrales Reales (RCC) [8], Coeficientes de Predicción Lineal (LPC) [9], Coeficientes Cepstrales de Predicción Lineal (LPCC) [10], Coeficientes Cepstrales en Frecuencia en escala de Mel (MFCC) [11] y Coeficientes de Predicción Lineal Perceptivos (PLPC) [12]. Los parámetros MFCC son características basadas en el modelo auditivo humano, por esta razón múltiples proyectos enfocados al reconocimiento de voz, hacen uso de estas características.

Se han propuesto algunos sistemas para el control de camas en hospitales. Lin Tan en [13], propone el diseño de una cama robótica enfatizando el diseño mecánico de la misma, contiene una tarjeta comercial, VoiceEzTM, que realiza el reconocimiento de voz usando LPCC, un microcontrolador AT89C4051 que realiza el control de un teclado y actuadores de la cama, una tarjeta central y un Bus CAN para comunicar los módulos comerciales.

Abdallah K. en [14], propone un sistema para controlar camas en hospitales. El sistema recibe un comando de voz en un Smartphone y lo envía al API de procesamiento de voz de Google para convertir el audio a texto. Dicho texto se envía a un servidor. El servidor manda las cadenas de texto a una tarjeta Raspberry la cual controla la cama.

En este trabajo se presenta la implementación de un sistema que permite el control de camas ortopédicas de 3 segmentos por medio de procesamiento de voz usando parámetros MFCC para el reconocimiento de comandos específicos. Todos los algoritmos de procesamiento y reconocimiento son implementados en un sistema embebido.

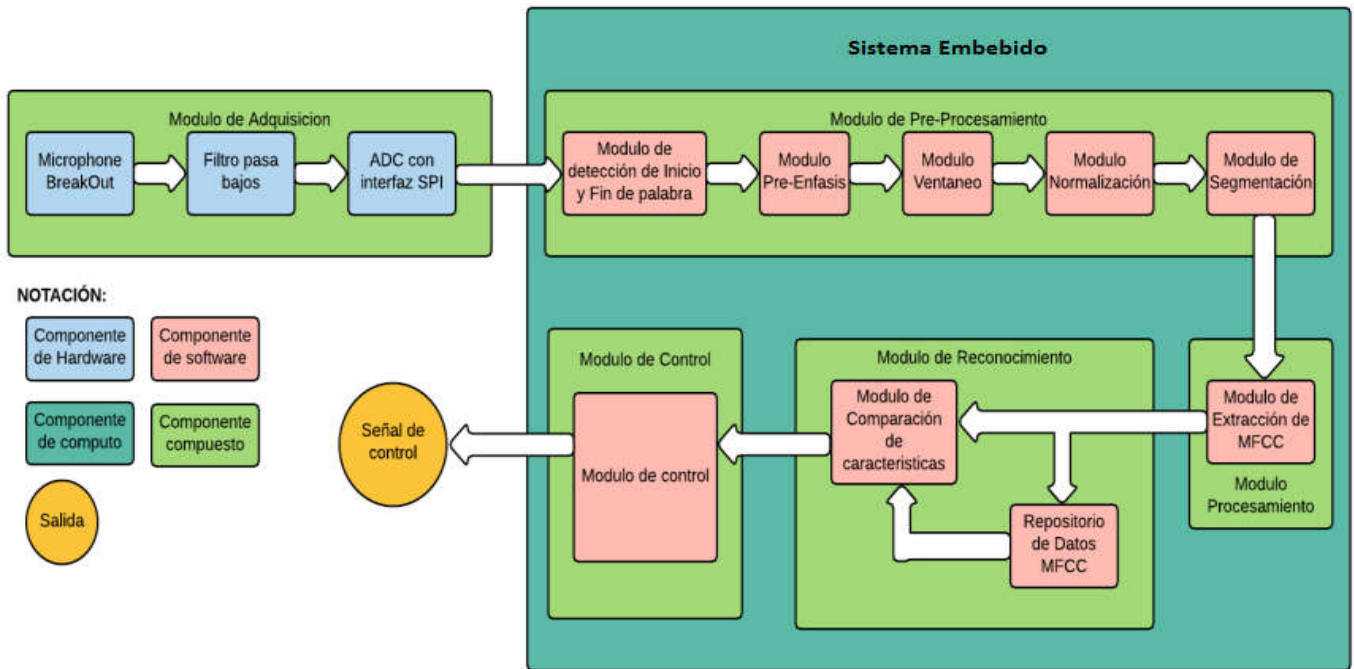


Fig. 1. Arquitectura general del sistema.

II. ARQUITECTURA DEL SISTEMA

La arquitectura general del sistema se muestra en la Fig. 1.

La arquitectura del sistema propuesto está formado de los siguientes módulos: Módulo de Pre-procesamiento, Módulo de procesamiento, Módulo de reconocimiento, Módulo de control, Módulo de adquisición y el Sistema embebido. Los módulos que incorpora el sistema se explican a continuación:

A. Módulo de Pre-procesamiento

Módulo de Detección de inicio y fin de palabra.

Este módulo, verifica la cantidad de energía contenida en las muestras de la señal de voz, $x(n)$, usando la ecuación 1. Si el la cantidad de energía supera un umbral de energía dado por el 15% con respecto al máximo valor de la señal, se realiza la extracción de un segmento de muestras correspondientes a 1 segundo.

$$E = \sum_{n=0}^{N-1} x(n)^2 \quad (1)$$

Módulo de Pre-énfasis

El tracto vocal manifiesta un número muy grande de resonancias, sin embargo, se consideran sólo las tres o cuatro primeras que toman el nombre de formantes y cubren un rango de frecuencias entre 100 y 3500 Hz. Los formantes de alta frecuencia tienen una amplitud pequeña con respecto a los formantes de baja frecuencia. El pre-énfasis para altas frecuencias es requerido para obtener amplitudes similares para todos los formantes mediante un filtro de Respuesta Finita al

Impulso (FIR-Finite Impulse Response) de primer orden dado por la ecuación 2. Donde $x(n)$ es la señal de voz.

$$y(n) = x(n) - ax(n-1) \quad (2)$$

$$a = 0.95$$

Módulo de Ventaneo

Al realizar el procesamiento en bloques de N muestras, se introduce un efecto conocido como Gibbs, para evitarlo se utiliza una ventana de Hann, $v(n)$, dada por la ecuación 3, mediante el producto de la ventana y la señal de voz, $v(n)x(n)$.

$$v(n) = 0.5 - 0.5 \cos\left(\frac{2\pi n}{N-1}\right) \quad (3)$$

Módulo de Normalización

La normalización tiene como objetivo lograr una semejanza mayor entre las muestras que contienen un mismo comando. El algoritmo consiste en tomar el valor más grande, VM , de las muestras de la señal de voz, $x(n)$, y posteriormente dividir cada valor de las muestras entre el valor más grande, aplicando la ecuación 4.

$$x(n) = \frac{x(n)}{VM} \quad (4)$$

Módulo de Segmentación

Para que la voz pueda ser procesada de manera óptima, es necesario segmentarla. Se consideran segmentos de 20 ms ya que durante este periodo los parámetros de la voz permanecen

constantes, esto quiere decir que esa es la mínima duración en la que un fonema puede ser pronunciado.

B. Módulo de Procesamiento

Módulo de Extracción de parámetros MFCC.

El algoritmo para obtener los parámetros MFCC [11] se muestra en la Fig. 2.

El primer paso de este algoritmo consiste en obtener la Transformada de Fourier de la señal de voz, de acuerdo a la ecuación 5, sin embargo hacemos uso del algoritmo de la Transformada Rápida de Fourier (FFT-Fast Fourier Transform) [15] ya que reduce la cantidad de productos complejos a $\frac{N}{2} \log_2 N$.

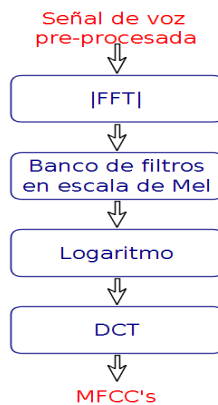


Fig. 2. Algoritmo de extracción de parámetros MFCC.

$$X[k] = \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi nk}{N}}; k = 0, 1, \dots, N-1 \quad (5)$$

Donde $x(n)$ es la señal de voz. Después se genera un banco de filtros en escala de Mels. Estos filtros normalmente son triangulares, repartidos en el rango de frecuencias completas, desde cero hasta la frecuencia de Nyquist. Para el análisis de filtros solamente, las frecuencias de corte más baja y la más alta se pueden establecer utilizando parámetros de configuración LOFREQ=80Hz (frecuencia más baja) y HIFREQ=4000Hz (frecuencia más alta). El diseño de los filtros se realiza espaciando las frecuencias centrales en una escala de Mel, $\beta(f)$, en función de las frecuencias de corte, haciendo uso de la ecuación 6.

$$\begin{aligned} \beta(f) &= 1125 \ln\left(1 + \frac{f}{700}\right) \\ \beta(80) &= 121.74 \text{ Mels} \\ \beta(4000) &= 2142.26 \text{ Mels} \end{aligned} \quad (6)$$

Posteriormente se genera un espaciado en esta escala, haciendo el uso de 30 filtros. Usando la ecuación 7.

$$\text{espacio} = \frac{2142.26 - 121.74}{30} = 67.35 \text{ Mels} \quad (7)$$

El siguiente paso es hacer el espaciado en escala de Mel, y hacer el cálculo de que frecuencia le corresponde a ese valor de Mel, despejamos f de la ecuación 6 y obtenemos la ecuación 8.

$$f = 700(e^{\frac{\beta(f)}{1125}} - 1) \quad (8)$$

Por último se buscarán los subíndices del arreglo de la FFT proporcionales a estas frecuencias, para crear un filtro por cada una de estas frecuencias, generando pendientes positivas cuando se generan valores hacia el próximo valor, y pendientes negativas cuando se viene del próximo valor, de esta forma se generan filtros triangulares, $H_m(k)$, mediante la ecuación 9.

$$H_m(k) = \begin{cases} 0, & k < f(m-1) \\ \frac{k - f(m-1)}{f(m) - f(m-1)}, & f(m-1) \leq k \leq f(m) \\ \frac{f(m+1) - k}{f(m+1) - f(m)}, & f(m) \leq k \leq f(m+1) \\ 0, & k > f(m+1) \end{cases} \quad (9)$$

Una vez que la envolvente del espectro de la señal de voz es multiplicada por el banco de filtros, se calcula la energía correspondiente en cada uno de los filtros, mediante la ecuación 10.

$$E_m = \sum_{k=0}^{N-1} |X[k]|^2 H_m[k], 1 \leq m \leq F \quad (10)$$

Tras obtener la energía, debemos de calcularle el logaritmo mediante la ecuación 11, pasando por lo tanto al dominio de la potencia espectral logarítmica.

$$Y_m = \log(E_m) \quad (11)$$

El inconveniente de trabajar en este dominio es que los espectros de los filtros en las bandas adyacentes presentan un alto grado de correlación, originando coeficientes espectrales estadísticamente muy dependientes entre ellos. Para eliminar dicha dependencia o correlación estadística, se hace uso de la Transformada Discreta del Coseno (DCT), que lleva los coeficientes espectrales en frecuencia, al dominio de la quefrecuencia convirtiéndolos en coeficientes cepstrales (MFCC) [11] usando la ecuación 12.

$$C_{MFCC}[m] = \sum_{k=0}^{N-1} Y_k \cos\left(\frac{m\pi}{N}\left(k + \frac{1}{2}\right)\right) \quad (12)$$

C. Módulo de reconocimiento

En la etapa de reconocimiento se utiliza un clasificador KNN, haciendo uso de la distancia Manhattan.

El sistema cuenta con un repositorio de datos, que contiene muestras de los parámetros MFCC de cada comando. Se hace lectura de todos los archivos de un comando, y se compara uno a uno, obteniendo la distancia Manhattan, a través del absoluto de la resta y una suma acumulativa usando la ecuación 13.

$$d = \sum_{k=0}^{29} |C_{MFCCin}(r) - C_{MFCCen}(r)| \quad (13)$$

Este proceso se repite por cada comando del corpus, obteniendo un vector de datos por cada comando. Una vez terminado, cada vector es promediado y el promedio con el valor menor es el comando reconocido.

D. Módulo de control

Las camas ortopédicas constan de un somier articulado de 2, 3 o 4 segmentos móviles. En las de 3 segmentos el superior soporta la cabeza y la espalda, el central es para la pelvis y el inferior es para las extremidades inferiores. Los comandos seleccionados para el control de la cama ortopédica son cabecera, pelvis y pies, los cuales identifican un segmento de la cama ortopédica; y arriba, abajo y alto, los cuales identifican una acción en el segmento. El sistema necesita de una secuencia de comandos que deben ir en un orden determinado, puesto que la movilidad de la cama tiene movimientos específicos, como son el movimiento de la cabecera hacia arriba o hacia abajo, lo mismo que en la pelvis y las piernas. Por lo tanto, para validar la sucesión de comandos, se usa una máquina de Mealy, que valida esta sucesión, y con respecto a esta, genera las salidas correspondientes.

Los estados de la máquina de Mealy y los códigos de cada comando se muestran en la Fig. 3 y en la Fig. 4 se muestra la máquina de Mealy completa.

| | |
|--------------------|---------------------------------|
| Q1: Estado inicial | 1: Código de comando cabecera |
| Q2: Cabeza | 2: Código de comando pelvis |
| Q3: Pelvis | 3: Código de comando pies |
| Q4: Pies | 4: Código de comando arriba |
| Q5: Arriba | 5: Código de comando abajo |
| Q6: Abajo | 6: Código de comando alto, otro |

Fig. 3. Comandos y estados de la máquina de Mealy.

E. Módulo de adquisición de datos

Este módulo está formada por un micrófono omnidireccional de tecnología MEM, ADMP401 de la compañía Analog Devices [16]. Después de realizar la adquisición de voz se usa un filtro pasabajas Butterworth de 8 orden con frecuencia de corte programable, MAX295 de la compañía Maxim, como filtro pre-alias [17]. Finalmente se usa

un Convertidor Analógico-Digital de 16 bits con interfaz Serial Periférica (SPI – Serial Peripheral Interface), MAX11161 de la compañía Maxim [18]. Este convertidor es configurado por el sistema embebido para la adquisición de la señal de voz.

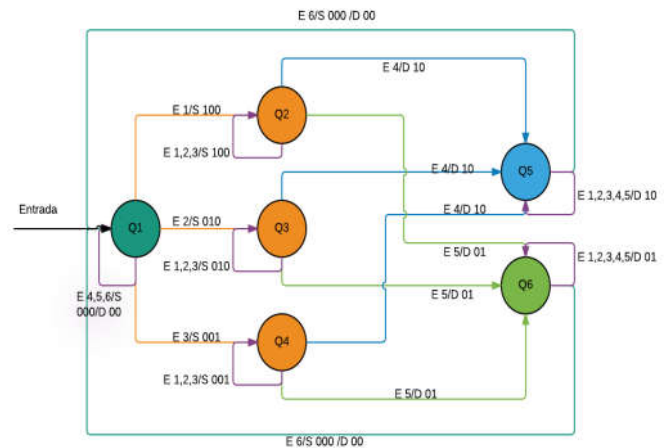


Fig. 4. Máquina de Mealy.

F. Sistema embebido

Para la implementación del sistema de procesamiento y reconocimiento de voz en camas ortopédicas es necesario un sistema computacional diseñado con hardware y software especialmente para desempeñar tareas específicas logrando así obtener beneficios en desempeño, costo y usabilidad del sistema, dicho sistema se le denomina sistema embebido [19].

El sistema embebido está basado en el sistema de desarrollo Raspberry Pi 3 [20], el cual tiene como principales características: un procesador Quad core Cortex A53 de 1.2GHz, con 512Kb en memoria cache L2 dentro del SoC BCM2837, 1GB en RAM, 1 procesador gráfico VideoCore IV y diversos periféricos. En la figura 5 se muestran los componentes implementados en el sistema embebido.

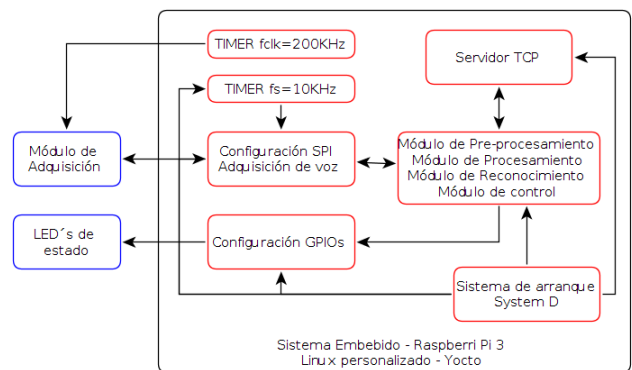


Fig. 5. Sistema Embebido.

El sistema operativo utilizado con el sistema de desarrollo es una versión personalizada, denominada rpi-basic-image, generada con el proyecto Yocto [21]. Esta imagen tiene la configuración del hardware para poder inicializar linux en la raspberry con algunas características adicionales y ocupa tan

solo 180Mb. La versión de kernel usada en la imagen es la 4.4.32. El sistema de arranque usado para el demonio que tiene la aplicación presentada aquí es *systemD* [22]. *SystemD* permite que la aplicación se ejecute por el proceso *init* como demonio desde el arranque del sistema mediante un script de configuración.

Además de los módulos de procesamiento y reconocimiento de voz, el sistema incorpora un módulo para la configuración SPI del ADC usado mediante la librería WiringPi [23]. Con este módulo se realiza la adquisición de la señal de voz.

Dos TIMERS son generados por software usando la librería WiringPi mediante hilos para su ejecución concurrente. Uno TIMER genera la frecuencia de muestreo $f_s=10420\text{Hz}$ y el otro TIMER genera la frecuencia de reloj, f_{clk} , mediante la ecuación 13, para configurar la frecuencia de corte, $f_c=4\text{KHz}$, del filtro pre-alias MAX295 [17]. En ambos TIMERS se usa una frecuencia en el procesador ARM del sistema de desarrollo de 600MHz para el cálculo de las frecuencias.

$$f_{clk} = 50f_c \quad (13)$$

La máquina de Mealy implementada en el módulo de control genera un código de salida de 3 símbolos, los cuales son visualizados en un conjunto de LEDs conectados a los GPIOs del sistema Raspberry Pi 3. Estos GPIOs se configuran y controlan usando un script para bash, el cual accede al Estándar de Jerarquía del Sistema de Archivos (FHS – File System Hierarchy Standard) de linux. En particular se accede a `/sys/class/gpio` donde se encuentran los archivos para acceder desde el espacio de usuario al device driver de los GPIOs en el espacio de kernel.

También se tiene programado un servidor mediante sockets TCP de Berkeley con una arquitectura cliente-servidor. Este servidor permite el envío de los comandos de voz por red, por lo que es posible controlar la cama ortopédica de manera remota mediante un Smartphone o algún otro equipo de cómputo. Esta característica permite que el sistema se convierta en un sistema autónomo para Internet de las Cosas (IoT- Internet of Things) [24].

III. PRUEBAS Y RESULTADOS

Para el diseño del sistema embebido se utilizó la metodología en V [25], en esta metodología se plantean primero pruebas unitarias de cada módulo del sistema y finalmente las pruebas de integración, por lo que estas pruebas se describen a continuación.

A. Prueba de los módulos de pre-procesamiento y procesamiento

Cada módulo de procesamiento y reconocimiento de la señal de voz se probó de manera individual.

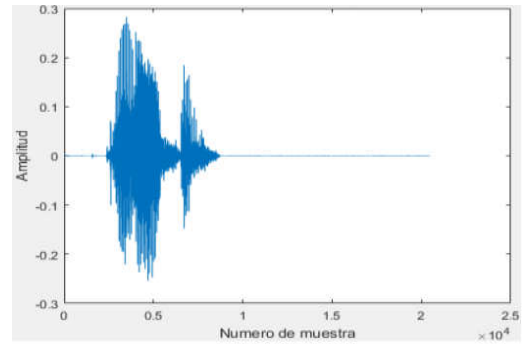


Fig. 6. Comando "cabeza" original.

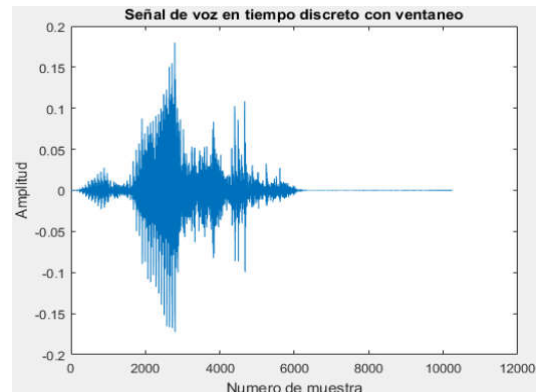


Fig. 7. Comando con pre-énfasis y ventaneo.

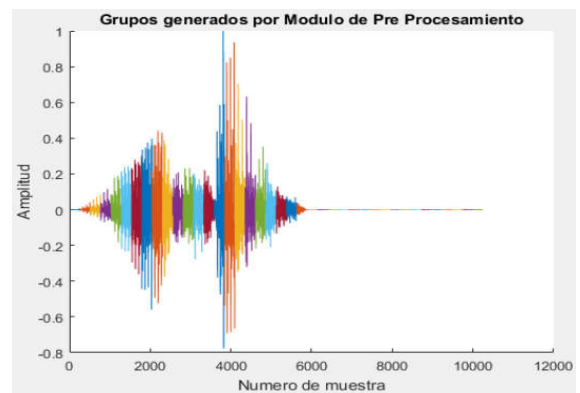


Fig. 8. Comando segmentado.

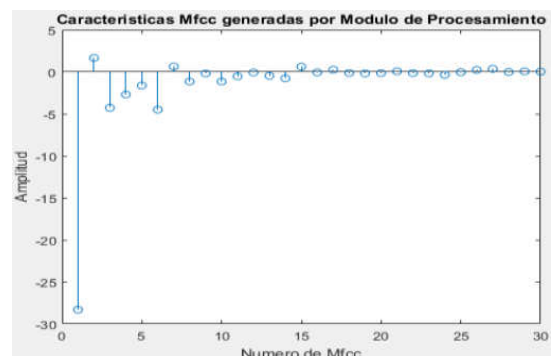


Fig. 9. Parámetros MFCC.

En la Fig. 6 se muestra el comando “cabeza” y los resultados de los módulos de pre-procesamiento, Fig. 7 y Fig. 8, y procesamiento del sistema, Fig. 9. Para cada comando se adquiere la señal de voz por un tiempo de 2 segundos después de detectar el inicio de dicho comando.

B. Prueba de eficiencia

Cada comando utilizado para el control de la cama cabecera, pelvis, pies, arriba, abajo y alto, tiene un repositorio de datos que consiste en un conjunto de 20 archivos, que contienen muestras de los parámetros MFCC de cada comando. Este repositorio se utiliza para la etapa de reconocimiento con el clasificador K-NN. En la etapa de pruebas se usaron 10 repeticiones de cada comando, en total 60 comandos, para probar la eficiencia del sistema. En las 60 pruebas realizadas todas se clasificaron correctamente, teniéndose 100% de aciertos. De cada comando se hizo una tabla de resultado como se muestra en la Fig. 10.

| # de prueba “Cabeza” | Correcta | Errónea | Mínimo | Máximo | Promedio |
|----------------------|----------|---------|--------|--------|----------|
| 1 | 1 | 0 | 0.6679 | 0.6679 | 0.6679 |
| 2 | 1 | 0 | 0.3429 | 0.3429 | 0.3429 |
| 3 | 1 | 0 | 0.4239 | 0.4239 | 0.4239 |
| 4 | 1 | 0 | 0.4284 | 0.4284 | 0.4284 |
| 5 | 1 | 0 | 0.5502 | 0.5502 | 0.5502 |
| 6 | 1 | 0 | 0.5332 | 0.5332 | 0.5332 |
| 7 | 1 | 0 | 0.5973 | 0.5973 | 0.5973 |
| 8 | 1 | 0 | 0.3905 | 0.3905 | 0.3905 |
| 9 | 1 | 0 | 0.6144 | 0.6144 | 0.6144 |
| 10 | 1 | 0 | 0.5552 | 0.5552 | 0.5552 |
| | 10 | 0 | 0.3429 | 0.6679 | 0.51039 |

Fig. 10. Pruebas del sistema para el comando “cabeza”.

En la Fig. 10 se observa el valor de la distancia mínima y máxima obtenida para cada comando usado en las pruebas. Los resultados de los demás comandos pueden consultarse en [26].

C. Módulo de adquisición

El sistema de desarrollo Raspberry Pi 3 no contiene un ADC como parte de sus periféricos, por lo que se desarrolló el diseño electrónico de un módulo adquisición para que funcione como un módulo de expansión. La Tarjeta de Circuito Impreso (PCB – Print Circuit Board) se muestra en la Fig. 11. Este módulo utiliza 1 conector basado en la especificación de interfaz PMOD [27].

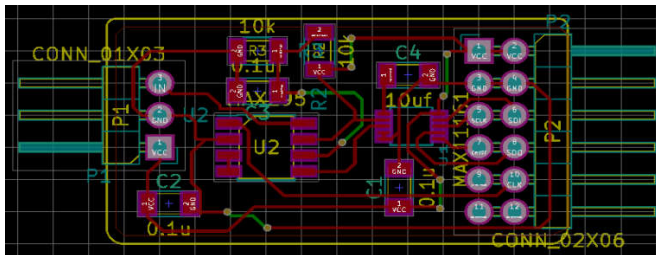


Fig. 11. PCB del módulo de adquisición.

D. Implementación paralela

Dado que el sistema de desarrollo Raspberry Pi 3 tiene un procesador ARM de 4 núcleos, podemos utilizar el paralelismo a nivel de datos para ejecutar los algoritmos más rápidamente en el sistema embebido. En la Fig. 12 se muestra la creación de hilos para ejecutar de forma paralela los algoritmos del módulo de procesamiento.

```

void moduloDePreProcesamiento(){
    pthread_t tids[NumeroDeHilos]; // Identificadores de hilos
    int rets[NumeroDeHilos], ids[NumeroDeHilos] = {0, 1};
    register int x;
    //Proceso de paralelizacion de la funcion de deteccionInicioFin
    for(x=0; x < NumeroDeHilos; x++)
        pthread_create(&tids[x], NULL, deteccionInicioFin, (void *)&tids[x]);
    for(x=0; x < NumeroDeHilos; x++)
        pthread_join(tids[x], (void *)&rets[x]);
    //La funcion de preenfasis no puede ser paralelizada puesto que necesita valores consecutivos
    preenfasis();
    //Proceso de paralelizacion de la funcion de ventaneo
    for(x=0; x < NumeroDeHilos; x++)
        pthread_create(&tids[x], NULL, ventaneo, (void *)&tids[x]);
    for(x=0; x < NumeroDeHilos; x++)
        pthread_join(tids[x], (void *)&rets[x]);
    //Proceso de paralelizacion de la funcion de normalizar
    mayor = buscaMayor(serialRecortada, NumeroDeDatos/2);
    for(x=0; x < NumeroDeHilos; x++)
        pthread_create(&tids[x], NULL, normalizar, (void *)&tids[x]);
    for(x=0; x < NumeroDeHilos; x++)
        pthread_join(tids[x], (void *)&rets[x]);
}
    
```

Fig. 12. Programación paralela del módulo de pre-procesamiento.

IV. CONCLUSIONES

En México existen una gran cantidad de personas con capacidades diferentes que necesitan el uso de camas ortopédicas articuladas ya sea en un hospital o en casa. Estas camas pueden ser controladas por medio de comandos de voz del paciente. En este trabajo se presenta un sistema de reconocimiento de palabras aisladas para el control de las camas ortopédicas, el cual presenta un porcentaje de reconocimiento del 100%. El sistema es implementado en un sistema embebido basado en una tarjeta de desarrollo Raspberry Pi 3 y un módulo de adquisición de datos para que el sistema funcione en el lugar donde se encuentre la cama ortopédica. A diferencia de los trabajos presentados en [13-14] nuestro sistema no utiliza tarjetas comerciales ni software de terceros (API's) para el procesamiento de voz. Este sistema evita el tener a una persona dedicada a auxiliar al paciente de forma permanente, resolviendo el problema eficientemente desde una perspectiva económica, de tiempo y espacio. El sistema embebido procesa cada comando en un tiempo aproximado de 1.49 segundos, dicho tiempo fue tomado con el comando time de Linux. Dado que se adquiere cada comando por un intervalo de tiempo de 2 segundos, entonces el sistema procesa todos los algoritmos en tiempo real.

AGRADECIMIENTOS

Se agradece al IPN por el apoyo brindado a la realización de este trabajo a través del financiamiento del proyecto SIP 20170127. También a los estudiantes Francisco Javier Hernández Cervantes y Narda Aixel Ballesteros Reyes.

REFERENCIAS

- [1] INEGI. "Estadísticas a propósito del día internacional de las personas con discapacidad", 2015. [Online]. Available: <http://www.inegi.org.mx/saladeprensa/aproposito/2015/discapacidad0.pdf>.
- [2] K. Thompson, Ph.D. Depresión y Discapacidad. 2002. [Online]. Available: http://fpg.unc.edu/sites/fpg.unc.edu/files/resources/otherresources/NCODH_DepresionSpanish.pdf.
- [3] R. Suriá Martínez; "Discapacidad adquirida y discapacidad sobrevenida: análisis comparativo de la sobrecarga que generan ambas formas de discapacidad en las madres de hijos afectados", Revista Española sobre Discapacidad Intelectual. Vol 42 (3), Num 239, pp. 67-84. 2011.
- [4] Aspam Madrid. "Lesión Medular: Guía para el manejo integral del paciente con LM crónica". [Online]. Available: http://www.isfie.org/documentos/guia_practica_lm.pdf. 2004.
http://onc.org.mx/wp-content/uploads/2018/02/PDF_dic17_final.pdf
- [5] Patents T. Douglas, "Voice-actuated, speakerdependent control system for hospital bed", US5335313A, 2 Ago 1994. [Online]. Available: <https://www.google.ch/patents/US5335313>.
- [6] Patents "Intelligent bed controlled by voice", CN 200620006730, Oct 31, 2007. [Online]. Available: <https://google.com/patents/CN200966481Y?c=nl>.
- [7] Patents "Voice-operated nursing bed", CN204411140 U, Jun 24, 2015. [Online]. Available: <https://www.google.com/patents/CN204411140U?c=zh>.
- [8] A. Oppenheim, "Homomorphic Analysis of Speech", IEEE Transactions on Audio and Electroacoustics, Volume 16, Issue 2, pp. 221-226, Jun 1968.
- [9] B.S.Atal, M.R.Schroeder. "Adaptive Predictive Coding of Speech Signals", The Bell System Technical Journal, Volume 49, Issue 8, pp. 1973-1986, Oct. 1970.
- [10] B.S.Atal, "Automatic Recognition of Speakers from their Voices", Proceedings of the IEEE, Volume 64, Issue 4, pp. 460-475, April 1976.
- [11] S.B.Davis, P. Mermelstein. "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences". IEEE Transactions on Acoustics, Speech, and Signal Processing. Volume 28, Issue 4, pp. 357-366, Aug 1980.
- [12] H. Hermansky, B. Hanson, H. Wakita, "Perceptually based linear predictive analysis of speech", Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP, 1985.
- [13] L. Tan, S. Lu, W. Zhanga, "Robotic Nursing Bed Design and Its Control System", Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics, pp. 2002-2006, 2009, Guilin, China.
- [14] A. Kassem, et al. "MedBed: Smart Medical Bed", Fourth International Conference on Advances in Biomedical Engineering (ICABME). 2017.
- [15] J. Cooley, et al, "Historical notes on the fast Fourier Transform", IEEE Transactions on Audio and Electroacoustics, Volume 15, Issue 2, June 1967.
- [16] Analog Devices, Inc. ADMP401 Data Sheet. 2018. <http://www.analog.com>.
- [17] Maxim Integrated, Inc. MAX295 Data Sheet. 2018. <http://www.maximintegrated.com>.
- [18] Maxim Integrated, Inc. MAX1161 Data Sheet. 2018. <http://www.maximintegrated.com>.
- [19] KAMAL, Raj. Embedded Systems: Architecture, programming and design. (Second Edition). India: McGraw-Hill Education, 2009, pp 681. ISBN-10: 0070151253.
- [20] Raspberry Pi Foundation. "Raspberry Pi 3 Model B", 2018.
- [21] Yocto Project. "Yocto Reference Manual", 2018.
- [22] L. Poettering, K. Sievers, T. Leemhuis, "Control Centre: The systemd Linux init system", The H Open, [Online]. Available: <http://www.h-online.com/open/features/Control-Centre-The-systemd-Linux-init-system-1565543.html>.
- [23] G. Henderson, WirinPi. [Online]. Available: <http://wiringpi.com>.
- [24] George Corser. "IEEE Talks IoT: George Corser". IEEE Internet of Things, 2018. [Online]. Available: <https://iot.ieee.org/articles-publications/ieee-talks-iot/206-ieee-talks-iot-george-corser.html>.
- [25] PEREZ, A; et al. "Una metodología para el desarrollo de hardware y software embebidos en sistemas críticos de seguridad". Systemics, Cybernetics and Informatics Journal, vol 3, Num. 2, pp. 70-75, 2006.
- [26] F.J.Hernández, N.A. Ballesteros, V.H.García, "SCCOM - Sistema de Control de Camas Ortopédicas Enfocado en Pacientes con Cuadruplejía", Tesis de Licenciatura, Escuela Superior de Cómputo del IPN. 2018.
- [27] Digilent Inc. "Digilent PMOD Interface Specification", 2011.

Sistema para el monitoreo y localización de vehículos usando IoT

V.H.García O., R.J. Romero T.

Instituto Politécnico Nacional, ESCOM
Dpto. de Ing. en Sistemas Computacionales
Av. Juan de Dios Batiz s/n, Lindavista, 07738
Ciudad de México
vgarciaortega@yahoo.com.mx

R. Hernández T.

Instituto Politécnico Nacional, UPIITA
Av IPN 2580, Barrio la Laguna Ticoman, 07340
Ciudad de México.
rhtovar@ipn.mx

Resumen—El presente trabajo presenta el desarrollo de un sistema de seguridad que permite realizar el monitoreo y localización de un vehículo terrestre. El sistema está formado por un sistema embebido, y una aplicación móvil. El sistema embebido utiliza un microcontrolador de bajo consumo de 16 bits, PIC24F32KA302 de la compañía Microchip, el cual contiene la aplicación. El monitoreo se realiza mediante sensores de efecto hall y toma de imágenes usando una cámara AD-7732 de la compañía AIDEVISION, la cual tiene un codificador JPEG. La localización del vehículo se realiza mediante el Sistema de Posicionamiento Global (GPS), usando el módulo SIM808 de la compañía SIMCom. Las imágenes y las coordenadas de localización son enviadas a una aplicación móvil usando el Sistema Global para Comunicaciones Móviles (GSM) incorporado en el mismo módulo SIM808 a través del Servicio de Mensajes Multimedia (MMS) y de Mensajes Cortos (SMS). El módulo SIM808 es configurado mediante comandos AT que son enviados desde el microcontrolador. La aplicación móvil permite visualizar las imágenes enviadas por el sistema embebido y ubica las coordenadas del módulo GPS en un mapa. La cámara y el módulo SIM808 tienen una interfaz UART de comunicación para su configuración con el microcontrolador. Toda la aplicación del sistema embebido es programada usando lenguaje C y ensamblador.

Palabras clave—Sistema embebido, GSM/GPS, monitoreo y localización de vehículos, IoT.

I. INTRODUCCION

En México, el robo de vehículos es considerado como delito de alto impacto social. Según la procuraduría general de justicia el Robo de Vehículo Automotor incluye vehículos particulares, del servicio público, de vehículos oficiales, motocicletas, de vehículos de transporte y vehículos de maquinaria.

Las cifras registradas por la Asociación Mexicana de Instituciones de Seguros, AMIS, en el periodo que comprende de enero a diciembre del 2017 muestran que el número de vehículos asegurados robados en todo el país fue de 90,187 teniendo un incremento del 27.2% con respecto al año 2016. El Estado de México es la entidad donde más carros asegurados se roban, pues en 2017 se robaron 26,061 [1]. Estas cifras solo muestran la cantidad de autos asegurados robados, por lo que al considerar el robo de autos robados no asegurados, aumenta.

De acuerdo al reporte sobre delitos de alto impacto publicado por el Observatorio Nacional Ciudadano en Diciembre del 2017 [2], el número de carpetas de investigación que se abrieron por robo de vehículos en 2017 fue de 186,896. Solo en el mes de Diciembre, el mayor número de carpetas de investigación por este ilícito se presentó en el Estado de México (4,366), Baja California (1,513) y Ciudad de México (943); concentrando 44.49% de los robos de vehículos del país.

Debido a que las cifras por robo de vehículos son muy elevadas y van en aumento de acuerdo a las cifras de AMIS y del Observatorio Nacional Ciudadano, es importante contar con un sistema que permita realizar el monitoreo y localización de vehículos automotores. El contar con información confiable y oportuna acerca del estado del vehículo, así como de su ubicación nos permite poder tomar acciones preventivas y correctivas para evitar el robo en conjunto con los servicios de emergencia.

Según George Corser [3] los sistemas autónomos de Internet de la cosas (IoT-Internet of Things) son un conjunto de dispositivos en red que se comunican entre sí, sin la intervención de humanos. Estos dispositivos normalmente usan un conjunto de sensores para realizar monitoreo y/o control del sistema e informar al usuario de su estado.

Un sistema que permite realizar el monitoreo y localización de vehículos automotores es un sistema autónomo de IoT. Ya que debe contar con sensores para el monitoreo y un módulo de comunicación para el envío de información en red.

Uno de los aspectos tecnológicos que tienen un mayor impacto en este tipo de sistemas es la red empleada en la comunicación, ya que debemos tomar en cuenta que si se emplea una red WiFi, no será viable la comunicación a grandes distancias, al menos que se cuente con conexión a internet, sin embargo, empleando una red como la GSM [4-7], es posible enviar información a distancias mayores, aunque esto reduce el ancho de banda de la red.

Este sistema autónomo de IoT es un sistema computacional diseñado con hardware y software especialmente para desempeñar tareas específicas logrando así obtener beneficios en desempeño, costo y usabilidad del sistema, por lo que se le denomina un sistema embebido [8].

Se han propuesto algunos sistemas para realizar el seguimiento y/o monitoreo de vehículos. Mohammad en [9],

propone un sistema de rastreo usando tecnología GSM/GPS enviando la información mediante SMS y visualizando la información en una computadora personal. Usa un módulo GSM SIM900D y un módulo GPS MT3329.

SeokJu Lee en [10] propone sistema de rastreo usando tecnología GPS/GSM/GPRS enviando la información mediante SMS y visualizando la información en una aplicación móvil en un smartphone. Usa un microcontrolador de Arduino, un módulo SM5100B GSM/GPRS y un módulo GPS EM-406.

Anil en [11-12] propone un sistema de rastreo usando tecnología GSM/GPS enviando la información mediante SMS y realizando monitoreo de varios sensores. Usa un microcontrolador ARM7 LPC 2148 y un módulo GSM SIM300.

En este trabajo se presenta la implementación de un sistema que permite el monitoreo de vehículos, usando sensores de efecto hall y una cámara, y el envío de esta información por medio de la red GSM, usando mensajes SMS y MMS, a un Smartphone, el cual cuenta con una aplicación móvil para poder desplegar la ubicación del vehículo en un mapa.

II. ARQUITECTURA DEL SISTEMA

La arquitectura general del sistema se muestra en la Fig. 1.

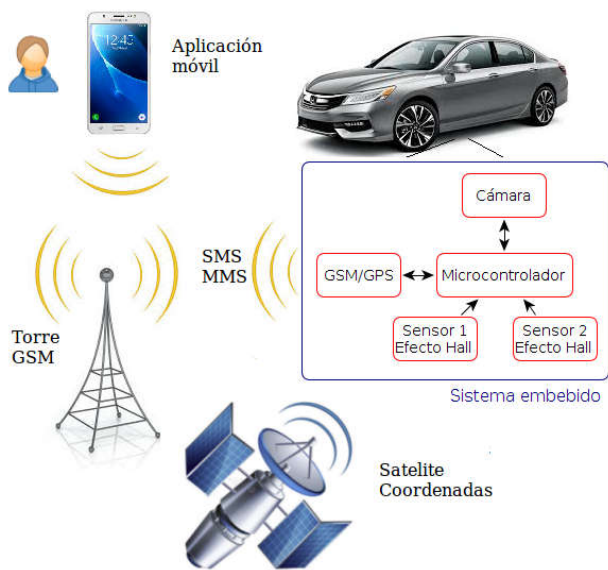


Fig. 1. Arquitectura general del sistema.

La arquitectura del sistema propuesto está formado del sistema embebido y de la aplicación móvil. Los módulos que incorpora el sistema se explican a continuación:

A. Módulo GSM/GPS

El Módulo usado es el modelo SIM808 [13] de la compañía SIMCom. Sus principales características son: Bandas de frecuencia 850/900/1800/1900 MHz, exactitud de

posicionamiento horizontal de 2.5m, tiempo para el primer ajuste 32s, una interfaz de Transmisión y Recepción Asíncrona Universal (UART – Universal Asynchronous Receiver Transmitter).

Después de la primera fijación, los datos recibidos por el módulo GPS están codificados de acuerdo al estándar definido por la NMEA (National Marine Electronics Association) [14], de la cual se extrae la cadena con encabezado GPGLL, la cual contiene información de posición, velocidad y tiempo (PVT - Position, Velocity, Time). Un ejemplo de la cadena recibida se muestra en la Fig. 2.

```
$GPGLL,4717.11634,N,00833.91297,E,124923.00,A,A*6E
```

Fig. 2. Mensaje GPGLL.

En la Tabla I, se puede apreciar el significado de cada parámetro contenido en el mensaje GPGLL.

TABLA I. DESCRIPCIÓN DEL MENSAJE GPGLL

| Campo | Formato | Ejemplo | Descripción |
|---------|-----------------|-------------|-----------------------------|
| xxGLL | String | \$GPGLL | Mensaje GLL |
| Lat | ddmm.mmm mm | 4717.11364 | Latitud (grados y minutos) |
| NS | character | N | North/South |
| long | dddmm.mm mmm | 00833.91297 | Longitud (grados y minutos) |
| EW | character | E | East/West |
| tiempo | hhmmss.ss | 124923.00 | UTC |
| estado | character | A | V=inválido, A=válido |
| posMode | character | A | Modo de posicionamiento |
| Cs | hexadecimal | *6E | Checksum |

GSM se ha usado en diversas aplicaciones de monitoreo y control [15-17] usando microcontroladores de 8 bits. El módulo SIM808 se configura, sin usar una computadora personal, mediante comandos AT [18], los cuales son enviados desde el microcontrolador para su inicialización, configuración, envío y recepción de mensajes. El módulo SIM808 utiliza la interfaz UART de comunicación para el envío y recepción de los comandos AT desde el microcontrolador.

El formato de dichos comandos y de las respuestas es el siguiente:

Comando: AT+CXXX= [<parámetros>]<CR>

Respuesta:

<Comando AT><CR><CR><LF><respuesta><CR><LF>

Donde: AT+: Es el prefijo que indica que un paquete se trata de un comando AT.

CXXX: El nombre del comando.

<Parámetros>: Si las instrucciones necesitarán parámetros, estos son puestos aquí.

<CR>: Retorno de Carro.

<LF>: Nueva línea.

Los comandos AT utilizados en esta aplicación son descritos en la Tabla II.

TABLA II. COMANDOS AT

| Comando | Información |
|----------|--|
| AT | Sintaxis: AT |
| | Descripción: Verifica que la comunicación ha sido establecida. El módem no realiza ninguna acción. Respuesta del Módulo: OK |
| AT+CMGF | Sintaxis: AT+CMGF=<o> |
| | Descripción: Activa o desactiva el modo texto. El parámetro o indica la acción a realizar: 0 para desactivar y 1 para activar. Respuesta del Módulo: OK |
| AT+CMGS | Sintaxis: AT+CMGS="<Prefijo internacional><Número de destino>" |
| | Descripción: Envía un mensaje de texto al número indicado en los parámetros prefijo internacional y número de destino. La respuesta del módem es '>' y entonces escribimos el cuerpo del mensaje, seguido de un 'z'. Respuesta del Módulo: La primera respuesta es el símbolo '>', tras el cual se debe escribir el cuerpo del mensaje. La respuesta al enviar el mensaje es CMGS: <número de referencia del mensaje> |
| AT+CMGR | Sintaxis: AT+CMGR=<indice> |
| | Descripción: Lee el mensaje almacenado en la posición de la memoria activa indicada por índice. Respuesta del Módulo: +CMGR: <estado>, <ca>, <dt>[, <tooa>, <pa>, <ldg>, <dlca>, <scsca>, <tosca>, <length>]<CR><LF><data> |
| AT+CMGD | Sintaxis: AT+CMGD=<indice> |
| | Descripción: Borra el mensaje almacenado en la posición de la memoria activa indicada por índice. Respuesta del Módulo: OK |
| AT+CMMSX | Comandos para el envío de mensajes MMS |

Con este módulo de comunicación el usuario puede monitorear los sensores usando SMS y el envío de imagen usando MMS.

Para la inicialización y configuración del módulo SIM808 mediante comandos AT, se programaron diferentes rutinas en el lenguaje ensamblador del microcontrolador. Se utiliza el lenguaje ensamblador para optimizar el código de la aplicación y para manejar las rutinas de atención a las interrupciones (ISR's - Interrupt Service Routines) de manera eficiente, puesto que la generación de código que realiza un compilador como el de lenguaje C introduce código ensamblador redundante, el cual afecta en la latencia de las ISR's. Las rutinas programadas en el lenguaje ensamblador son las siguientes:

a) Rutina de inicialización (INI_GSM). Esta rutina se encarga de inicializar el módulo SIM808. Establece la comunicación con el módem con el comando "AT", se deshabilita el eco en las respuestas del módem GSM y se establece el modo texto para usar el servicio de mensajes cortos (SMS - Short Message Service).

b) Rutina de envío de comandos AT (ENVIAR_CMD_GSM). Esta rutina se encarga de enviar los comandos AT al módem GSM a través de la interfaz de comunicación UART. Ver Fig. 3.

c) Rutina de envío de mensaje (ENVIAR_MSJ). Esta rutina se encarga de establecer el número telefónico al que será enviado el SMS además del contenido del mismo. Ver Fig. 3.

d) Rutina de recepción de mensaje (RECEPCION_MSJ). Establece la dirección del mensaje a leer de la memoria activa de la tarjeta SIM, posteriormente espera a que exista un

mensaje nuevo en dicha dirección, el cual será procesado por el microcontrolador y además será eliminado.

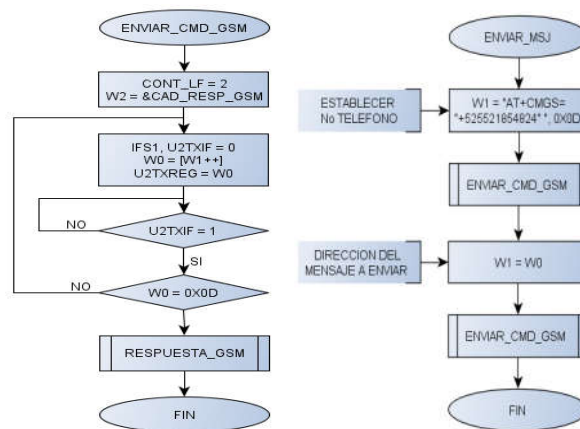


Fig. 3. Rutinas para el manejo del SIM808.

B. Cámara

La cámara usada es el modelo AD-7732 de la compañía AIDEVISION [19]. Sus principales características son: 0.3 Mega pixeles con compresión JPEG, resoluciones VGA (648x480), QVGA(320x240), QQVGA (160x120), usa el controlador OV528, interfaz UART con velocidades de 9600 a 115200 baudios.

La cámara cuenta con un conjunto de comandos que son usados para la inicialización, configuración, captura y lectura de imagen mediante un protocolo de comunicación entre el microcontrolador y la cámara [20]. Estos comandos tienen una estructura de 6 bytes, el byte inicial, el identificador del comando y cuatro bytes de parámetros. Estos comandos se muestran en la tabla III.

TABLA III. COMANDOS DE LA CÁMARA

| Command | ID Number | Parameter 1 | Parameter 2 | Parameter 3 | Parameter 4 |
|-----------------|-----------|---------------|----------------------|-----------------------|----------------------|
| Initialise | AA01h | 00h | Color Type | Preview Resolution | JPEG Resolution |
| Get Picture | AA04h | Picture Type | 00h | 00h | 00h |
| Snapshot | AA05h | Snapshot Type | Skip Frame Low Byte | Skip Frame High Byte | 00h |
| Set Packet Size | AA06h | 08h | Packet Size Low Byte | Packet Size High Byte | 00h |
| Set Baud rate | AA07h | 1st Divider | 2nd Divider | 00h | 00h |
| Reset | AA08h | Reset Type | 00h | 00h | xxh* |
| Power Off | AA09h | 00h | 00h | 00h | 00h |
| Data | AA0Ah | Data Type | Length Byte 0 | Length Byte 1 | Length Byte 2 |
| SYNC | AA0Dh | 00h | 00h | 00h | 00h |
| ACK | AA0Eh | Command ID | ACK counter | 00h/Packat ID Byte 0 | 00h/Packat ID Byte 1 |
| NAK | AA0Fh | 00h | NAK counter | Error Number | 00h |

Con estos comandos se sigue la secuencia mostrada en la Fig. 4 para realizar la captura de imágenes.

En la sección 1, se establece la comunicación entre el microcontrolador y la cámara con el uso del comando SYNC, después la inicialización de la cámara con el comando

INITIAL y finalmente se establece el tamaño de paquete con el comando Set Package Size.

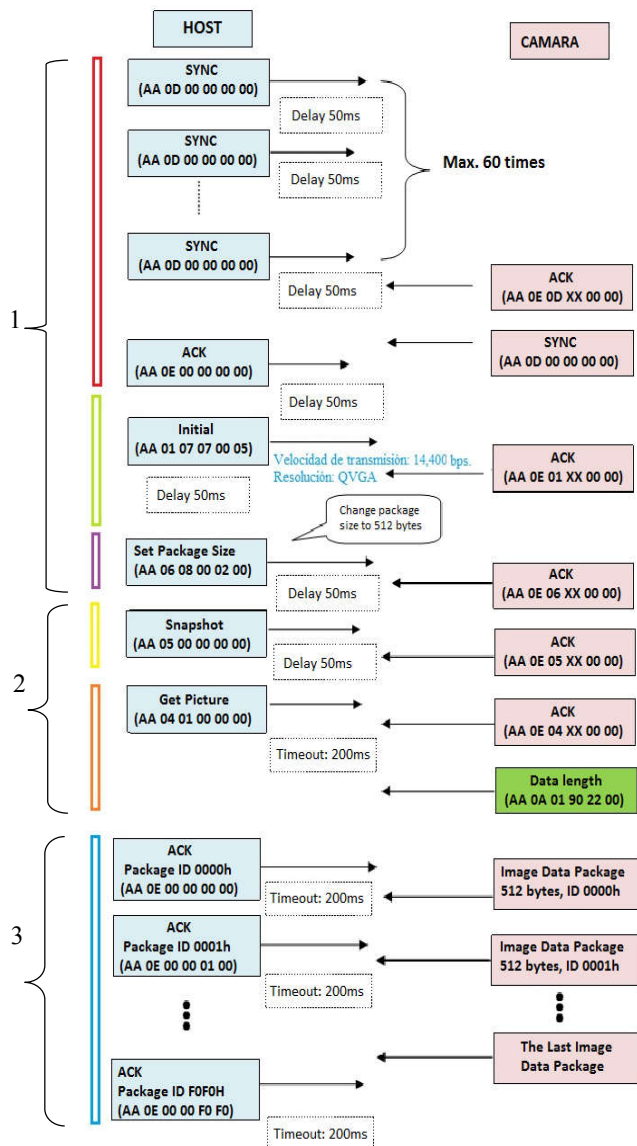


Fig. 4. Protocolo de comunicación para la lectura de imágenes.

En la sección 2, se realiza la captura de la imagen con el comando Snapshot, es aquí donde se realiza la compresión de la imagen en formato JPEG y se guarda en la memoria interna de la cámara. Es importante mencionar que este formato es necesario si se quiere mandar la imagen usando el servicio de mensajes multimedia del modem GSM, ya que los comandos AT para MMS solo soportan los formatos JPEG y GIF.

En la sección 3, se realiza la lectura de la imagen, con el comando GET PICTURE, en bloques de bytes de acuerdo al tamaño de paquete establecido.

C. Sensores

El sistema utiliza dos sensores de efecto hall modelo DN6851 [21] de la compañía Panasonic. Estos sensores se

utilizan para detectar la apertura de puertas y/o ventanas en el vehículo.

D. Microcontrolador de 16 bits

El sistema embebido está basado en microcontrolador PIC24F32KA302 [22] de la compañía Microchip, dicho fabricante nos provee de herramientas de desarrollo como el MPLAB IDE, MPLAB IPE, Programador Pickit 3, así como una amplia documentación y soporte en sus tecnologías. Algunas características relevantes del microcontrolador son: Arquitectura RISC de 16 bits, procesamiento de 16 MIPS memoria de programa de 32Kb y de datos de 2Kb, tecnología de bajo consumo XLP (Xtreme Low Power) con un consumo de 20 nA en modo Sleep, interfaces de comunicación UART, I2C, SPI.

El microcontrolador maneja cada uno de los módulos: Módulo SIM808, Cámara y Sensores usando diferentes interfaces de comunicación y recursos periféricos. Estas interfaces se describen en la Fig 5.

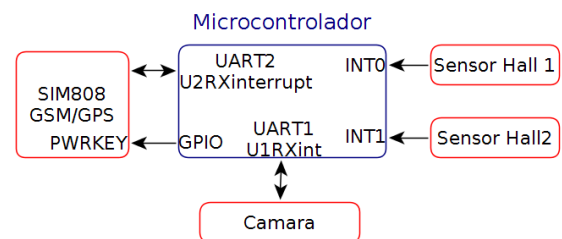


Fig. 5. Recursos periféricos usados en el microcontrolador.

El microcontrolador usa el UART1 para manejar la cámara y recibe sus respuestas mediante la interrupción de recepción. El UART2 maneja el SIM808 con los comandos AT y recibe las respuestas con la interrupción de recepción. Los sensores de efecto hall utilizan las interrupciones externas 0 y 1 para detectar la apertura de puertas del vehículo.

El microcontrolador tiene programada la aplicación para realizar la recepción de peticiones desde un Smartphone y realizar la captura de imagen, obtener coordenadas del vehículo y saldo del Módulo de Identificación del Suscriptor (SIM-Subscriber Identity Module). En caso de ocurrir algún evento en los sensores de efecto hall, se mandan un mensaje de alerta, las coordenadas del vehículo y una foto a un Smartphone. Las coordenadas del GPS y el saldo del SIM se envían por SMS y la imagen por MMS hacia la aplicación móvil. Esto se puede ver en la Fig. 6.

E. Aplicación móvil

Para el desarrollo de la aplicación móvil se optó por la plataforma Android, tomando en cuenta las características enlistadas a continuación:

- Integración con servicios de Google Inc.® como Google Maps
- Aplicaciones portables entre varios dispositivos
- Servicios de localización GPS

- Control de elementos de hardware
- Compatibilidad de entorno de desarrollo con Linux, Windows, Mac OS
- Lenguaje de programación de alto nivel (Java).

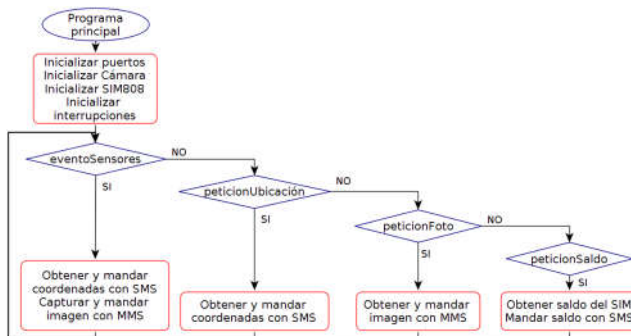


Fig. 6. Aplicación en el sistema embebido.

La aplicación móvil tiene un menú para solicitar tres peticiones:

- Ubicación: Se solicita al sistema embebido la ubicación del automóvil y éste le responde con las coordenadas en latitud-longitud en un SMS.
- Fotografía: Se solicita al sistema embebido una imagen del vehículo y éste envía en un mensaje MMS al usuario.
- Saldo: Se solicita al sistema embebido el saldo actual de la SIM del módulo GSM.

Una vez recibido el MMS se almacena la imagen y se muestra en un mapa la ubicación del sistema embebido.

III. PRUEBAS Y RESULTADOS

Para el diseño del sistema embebido se utilizó la metodología en V [23], en esta metodología se plantean primero pruebas unitarias de cada módulo del sistema y finalmente las pruebas de integración, por lo que estas pruebas se describen a continuación. Las pruebas iniciales consisten en comprobar cada una de las peticiones mediante SMS desde un Smartphone.

A. Prueba de petición de fotografía

En la Fig. 7 se muestra una prueba realizada con la cámara, la cual toma una foto desde el sistema embebido cuando se recibe la petición de fotografía desde un Smartphone con el SMS de “foto”. La foto es enviada por el sistema embebido mediante un MMS usando el módulo GSM del SIM808 hacia el Smartphone.

B. Prueba de petición de ubicación

En la Fig. 8 se muestra una prueba realizada con el módulo GPS, la cual toma las coordenadas desde el sistema embebido cuando se recibe la petición de ubicación desde un Smartphone con el SMS de “Ubicación”. Las coordenadas son enviadas por el sistema embebido mediante un SMS usando el módulo GSM del SIM808 hacia el Smartphone.

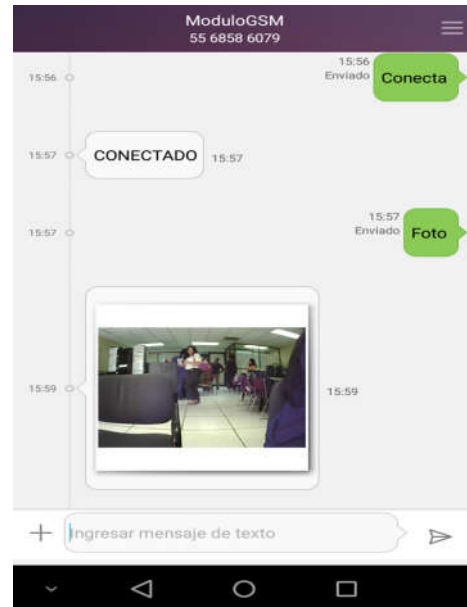


Fig. 7. Prueba de petición de fotografía.



Fig. 8. Prueba de petición de ubicación y saldo.

C. Prueba de petición de saldo

En la Fig. 8 se muestra una prueba realizada cuando se recibe la petición de saldo desde un Smartphone con el SMS de “Saldo”. El saldo del SIM es enviado por el sistema embebido mediante un SMS usando el módulo GSM del SIM808 hacia el Smartphone.

D. Prueba de activación de sensores

En la Fig. 9 se muestra una prueba realizada cuando ocurre un evento en los sensores de efecto hall, lo cual indica, que las puertas han sido abiertas. En ese momento es enviado un

mensaje de alerta, una fotografía y las coordenadas del vehículo desde el sistema embebido mediante un SMS y MMS usando el módulo GSM del SIM808 hacia el Smartphone.

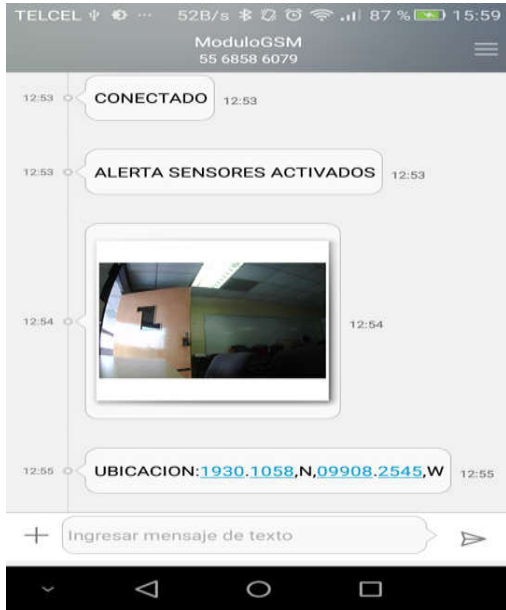


Fig. 9. Pruebas de activación de sensores.

E. Prueba de la aplicación móvil

La aplicación móvil es desarrollada con el IDE Android Studio, el cual provee las herramientas necesarias para la implementación de aplicaciones móviles. Se tiene el menú de peticiones y el menú de configuración. El menú de peticiones realiza las solicitudes de ubicación del vehículo, foto y saldo. El menú de configuración nos sirve para sincronizar nuestro número telefónico con el sistema y que sea el único al cual le responda el sistema además de tener la opción de activar/desactivar el sistema igualmente, agregando seguridad al mismo. La aplicación se muestra en la Fig. 10.

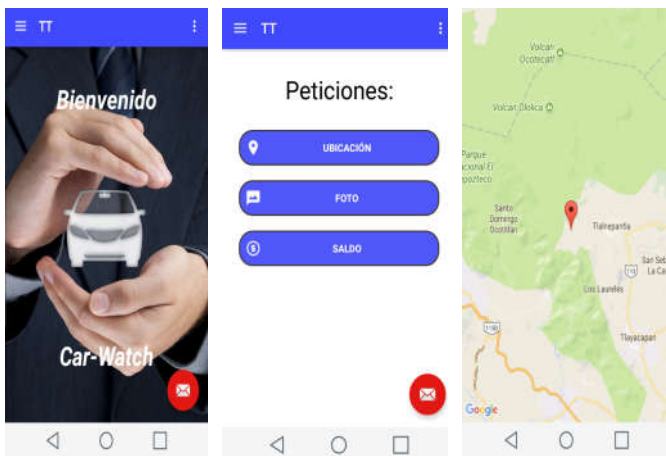


Fig. 10. Aplicación móvil.

F. Sistema embebido

El sistema embebido diseñado utiliza dos conectores basados en el estándar de MikroBus [24] y 4 conectores basados en la especificación de interfaz PMOD [25] para adaptar módulos periféricos fácilmente. Además, se tiene el conector para programación en circuito de forma serial (ICSP – In Circuit Serial Programming) [26]. El sistema embebido se muestra en la Fig. 9. En esta figura se muestra el módulo SIM808 y la cámara conectados a los conectores MikroBus.

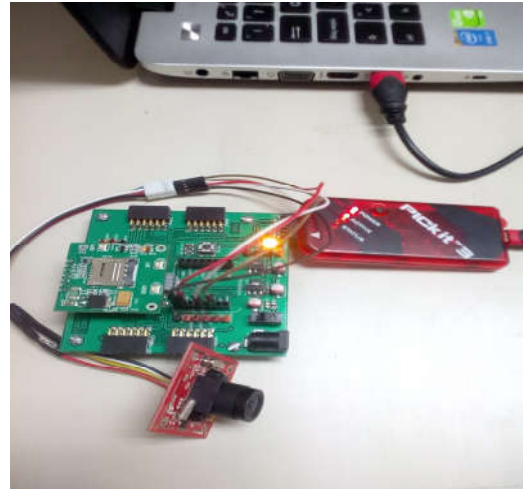


Fig. 11. Sistema embebido.

IV. CONCLUSIONES

El robo de vehículos es un delito de alto impacto muy grave que va en aumento en todo el país. Este delito provoca grandes pérdidas materiales a los ciudadanos. Los sistemas de monitoreo y localización de vehículos son una opción para recopilar información de la ubicación y estado del vehículo en el momento que el usuario lo solicita o cuando ocurre un evento. El desarrollo de sistemas embebidos es esencial en el control de los sistemas de monitoreo y localización de vehículos, ya que permiten diseñar e implementar sistemas para la obtención de datos. Estos datos pueden ser variados, en este trabajo se implementó un sistema embebido con un microcontrolador de 16 bits, para controlar una cámara para la toma de fotos, un módulo GSM/GPS para la obtención de coordenadas y para el envío de la información usando SMS y MMS. El uso de la cámara permite tener información del escenario donde se encuentra el vehículo, a diferencia de los sistemas presentados en [9-12]. Toda esta información es mostrada en una aplicación móvil, la cual controla el sistema embebido. Este sistema funciona de manera autónoma e independiente de la computadora del vehículo, por lo que puede ser implementado en cualquier tipo de vehículo no importando la marca, año y/o modelo.

AGRADECIMIENTOS

Se agradece al IPN por el apoyo brindado a la realización de este trabajo a través del financiamiento del proyecto SIP 20170127. También a los estudiantes Ramírez Aguilar Abigail Adoraim y Rodríguez Torres Gerardo.

REFERENCIAS

- [1] AMIS. Robo de Vehículos Asegurados. 2017. [Online]. Available: <http://www.amisprensa.org/wp-content/uploads/2018/01/Conf-Prensa-Robo-cifras-Diciembre-2017-v3.pdf>
- Observatorio Nacional Ciudadano. Reporte Sobre Delitos de Alto Impacto. 2017. [Online]. Available: http://onc.org.mx/wp-content/uploads/2018/02/PDF_dic17_final.pdf
- [2] George Corser. "IEEE Talks IoT: George Corser". IEEE Internet of Things, 2018. [Online]. Available: <https://iot.ieee.org/articles-publications/ieee-talks-iot/206-ieee-talks-iot-george-corser.html>.
- [3] AmolPoman, Mahesh G., Prashant P., "GSM Based LAN Monitoring System", International Journal of Computer Science and Information Technologies, Vol. 3 (3). 2012.
- [4] Hapsari, A.T., E.Y. Syamsudin, and I. Pramana, "Design of Automobile Position Tracking System Using Short Message Services And Its Implementation on FPGA", Proceedings of the Conference on Asia South Pacific Design Automation, Shanghai, China, 2005.
- [5] Fan, X., W. Xu, H. Chen, and L. Liu, "CCSMOMS:A Composite Communication Scheme for Mobile Object Management System", 20th International Conference on Advanced Information Networking and Applications, Volume 2, Issue 18-20, April 2006, pp. 235–239 .
- [6] B. Ramamurthy. "Development of a Low-Cost GSM SMS-Based Humidity Remote Monitoring and Control system for Industrial Applications", International Journal of Advanced Computer Science and Applications. Vol. 1, No. 4, October 2010.
- [7] KAMAL, Raj. Embedded Systems: Architecture, programming and design. (Second Edition). India: McGraw-Hill Education, 2009, pp 681. ISBN-10: 0070151253.
- [8] Mohammad, A. "Hybrid GPS-GSM Localization of Automobile Tracking System", International Journal of Computer Science & Information Technology (IJCSIT) Vol 3, No 6, Dec 2011.
- [9] SeokJu, L., Girma T. and Jaerock K. "Design and Implementation of Vehicle Tracking System Using GPS/GSM/GPRS Technology and Smartphone Application", IEEE World Forum on Internet of Things (WF-IoT). 2014.
- [10] Anil and Sree R., "Vehicle Tracking System", International Journal of Research in Electrical and Electronic Engineering, Vol. 4, No. 6, pp. 001-003, June 2015.
- [11] A. Anusha, Syed M. A., "Vehicle Tracking and Monitoring System to Enhance the Safety and Security Driving Using IoT", International Conference on Recent Trends in Electrical, Electronics and Computing Technologies, 2017.
- [12] SIMCom, Inc. GSM/GPRS+GPS Module SIM808 Data Sheet. 2018. [Online]. Available: <http://simcom.ee/modules/gsm-gprs-gnss/sim808/>.
- [13] SIMCom, Inc. SIM808 GPS Application Data Sheet. 2018. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/SIM808_GPS_Application_Note_V1.00.pdf.
- [14] B. Ramamurthy. "Development of a Low-Cost GSM SMS-Based Humidity Remote Monitoring and Control system for Industrial Applications", International Journal of Advanced Computer Science and Applications. Vol. 1, No. 4, October 2010.
- [15] S.R. Khan, "Design and Implementation of Low Cost Home Security System using GSM Network". International Journal of Scientific & Engineering Research. Volume 3, Issue 3, March -2012.
- [16] V. Madan, "GSM-Bluetooth based Remote Monitoring and control system with automatic light controller". International Journal of Computer applications. Volume 46-No 1, May -2012.
- [17] SIMCom, Inc. SIM808 Series AT Command Data Sheet. 2018. [Online]. Available: http://simcom.ee/documents/SIM808/SIM800%20Series_AT%20Command%20Manual_V1.09.pdf.
- [18] AIDEVISION, Inc. AIDEVISION Serial Port Camera Module AD-7732 Data Sheet. 2018.
- [19] AIDEVISION, Inc. AIDEVISION Serial Port Camera Series Communication Protocol Data Sheet. 2018.
- [20] Panasonic, Inc. *DN6851 Data Sheet*. 2018. <http://www.semicon.panasonic.co.jp/>.
- [21] Microchip Technology Inc. *PIC24FKA302 Data Sheet*. Chandler, Arizona. 2018. <http://www.microchip.com>.
- [22] PEREZ, A; et al. "Una metodología para el desarrollo de hardware y software embebidos en sistemas críticos de seguridad". Systemics, Cybernetics and Informatics Journal, vol 3, Num. 2, pp. 70-75, 2006.
- [23] Mikroelektronika Inc., "mikroBUS, Standard specifications", 2015.
- [24] Digilent Inc. "Digilent PMOD Interface Specification", 2011.
- [25] Microchip Inc. "ICSP Guide", 2003.

Estrategias basadas en cómputo ubicuo para el control de la temperatura en un invernadero

Esquivel Salas, A.

División de Informática y Sistemas
Instituto Tecnológico Superior Zacatecas Norte
Km, 3 Carr. A Gonzáles Ortega
Río Grande, Zacatecas, 98400, México
abraham.esquivel@itszn.edu.mx

Castañeda Delgado, J.

División de Informática y Sistemas
Instituto Tecnológico Superior Zacatecas Norte
Km, 3 Carr. A Gonzáles Ortega
Río Grande, Zacatecas, 98400, México
jaime.castaneda@itszn.edu.mx

Salas Guzmán, M. I.

División de Informática y Sistemas
Instituto Tecnológico Superior Zacatecas Norte
Km, 3 Carr. A Gonzáles Ortega
Río Grande, Zacatecas, 98400, México
ignacio.salas@itszn.edu.mx

Arredondo Salcedo, D.

División de Informática y Sistemas
Instituto Tecnológico Superior Zacatecas Norte
Km, 3 Carr. A Gonzáles Ortega
Río Grande, Zacatecas, 98400, México
daniel.arredondo@itszn.edu.mx

Molina Wong, M. R.

División de Informática y Sistemas
Instituto Tecnológico Superior Zacatecas Norte
Km, 3 Carr. A Gonzáles Ortega
Río Grande, Zacatecas, 98400, México
molina.wong@itszn.edu.mx

Resumen— Gran parte del éxito o fracaso de un cultivo en un entorno controlado, depende de tener información sobre la cual tomar decisiones. Existen desarrollos tecnológicos en el mercado para la monitorización de las variables climáticas, pero hasta cierto punto inaccesible por sus altos costos. En este artículo encontrará la propuesta de un sistema para la recolección de temperatura de un invernadero, capaz de actuar en la apertura o cierre de las ventanas como medio regulador, que puede resultar de interés para una gran cantidad de invernaderos que no pueden automatizarse por los altos costos.

Keywords— *computación ubicua, invernadero, temperatura*

I. INTRODUCCIÓN

Una fortaleza para las regiones centrales y norte de México, respecto a la agricultura protegida, en especial de invernadero, es que dadas las condiciones climáticas, por ejemplo en el altiplano Zacatecano, permiten producir en cualquier época del año, esto por estar en una parte alta, en un clima templado, árido y con buenas condiciones de luz, lo que posibilita alargar el ciclo agrícola. Por otra parte, entre los principales obstáculos para esta industria se encuentran una gran inversión de capital, alto costo de la energía, falta de infraestructura y la inconsistente calidad, que finalmente se traduce en bajos precios de venta para los productos.

Aun y cuando la tendencia hacia la agricultura protegida en México, Estados Unidos y Canadá va en continua expansión, en términos de tecnología y rendimiento nuestro país se encuentra en niveles inferiores, siendo lo más común encontrar sistemas de producción totalmente manuales, con limitado control del medio ambiente y sus variables[1].

Es necesario hacer énfasis, en que el crecimiento de los cultivos en los sistemas de agricultura protegida está en función del control de dos aspectos, las variables ambientales que le rodean, así como por la cantidad de agua y fertilizantes suministrados por el riego; el manejo adecuado de estas variables permitirá el control del crecimiento del cultivo. Un invernadero es ideal para el control de las variables mencionadas, ya que constituye un entorno cerrado en el que el clima y la fertirrigación pueden ser controlados, resultando en un crecimiento óptimo del cultivo. Sin embargo, el control de estos dos aspectos también está asociado a costos relacionados a la energía, agua y fertilizantes[2].

Desde la perspectiva económica, el objetivo de un invernadero no radica solo en la maximización de la producción, sino en la maximización del beneficio dado por la diferencia entre los ingresos de la venta final de la producción y sus costos asociados[2,3].

Finalmente, el control del clima en un invernadero se puede realizar mediante ventilación y calefacción, modificando las

condiciones de temperatura y humedad, sombreado y luz artificial para cambiar la radiación interna, inyección de CO₂ para influir en la fotosíntesis y nebulización para incrementar la humedad. Generalmente los invernaderos tecnificados enfocan sus esfuerzos en el control de la ventilación natural, calefacción y pantallas de sombreado [4].

Ya que la automatización de un invernadero capaz de controlar variables ambientales consiste en la adquisición de tecnología madura, que no es desarrollada en México, provoca que sus costos se eleven tornandola inaccesible. La propuesta de investigación que aquí se plantea, se dirige a buscar una solución que disminuya costos de automatización a partir de desarrollos basados en computación ubicua.

Partiendo de la necesidad expresada por productores locales, y que es extensiva a los invernaderos de la zona centro y norte, donde la administración y cuidado del cultivo es completamente manual, se lanzó el primer reto que consiste en el desarrollo de un mecanismo de supervisión que mida la temperatura, que posteriormente sera corregida, en medida de lo posible, a través de la apertura o cierre de las persianas.

En la sección II, encontrará una justificación que dió origen al proyecto, después se exponen los objetivos generales y específicos, continuando con otros trabajos relacionados, después podrá encontrar los Materiales y Métodos empleados, así como una descripción de los dispositivos usados en la sensorización y monitoreo. Finalmente encontrará las conclusiones y trabajo futuro.

II. MOTIVACIÓN

A partir de Convenio de Colaboración firmado en febrero de 2017, entre el Centro de Bachillerato Tecnológico Agropecuario No. 20 y el Instituto Tecnológico Superior Zacatecas Norte, con motivo de encontrar, en lo particular, una solución tecnológica que permita controlar la temperatura de un invernadero a partir del control automático de la temperatura a partir de la apertura o cierre de sus persianas. Necesidad que puede encontrarse en los invernaderos, al menos, del Estado de Zacatecas.

En lo general, el convenio pretende unir esfuerzos académicos y de vinculación para colaborar en futuros proyectos de investigación, así como de capacitación a personal del Centro de Bachillerato Tecnológico agropecuario No. 20.

Por otra parte, aún y cuando se habla de una agricultura protegida, algunas causas que inciden en la producción son:

- La variabilidad climática [5], que incluso es difícil de estimar cuánta producción se pierde por esta causa. Puede afectar en varios niveles (producción, cosecha, almacenamiento, transporte, etc.), incluso de manera indirecta a través de enfermedades, plagas, daños a la infraestructura, etc.
- Temperaturas extremosas (una helada por ejemplo), pueden significar la pérdida total de la producción.

- Los altos costos de los combustibles necesarios para controlar la temperatura, suponen una inversión que algunas veces es irrecuperable.
- La inversión es alta si se toma en cuenta que al ser invernaderos de baja o nula tecnología, requieren la atención constante de una persona con suficiente experiencia, durante el tiempo que dure el cultivo, controlando en medida de lo posible las variables ambientales.

III. OBJETIVOS

Aplicar la tecnología desarrollada en el Laboratorio de Inteligencia Ambiental de Instituto Tecnológico Superior Zacatecas Norte, en el control de variables de ambiente a través de la automatización de entornos de producción controlada como lo son los invernaderos.

- Diseñar e implementar el control automático para apertura y cierre de ventanas en un invernadero.
- Acondicionar plataformas de control automático para apertura y cierre de Ventanas.
- Diseño, instalación y normalización de parámetros eléctricos de sensores de proximidad para la detección de apertura y cierre total de ventanas en un invernadero.
- Instalar sensor de humedad y Temperatura para Monitorización de Ambiente.
- Diseñar e Instalar Sistema Fotovoltaico para energizar prototipos.
- Evaluar si la tecnología utilizada y generada disminuye los costos de automatización de un invernadero.

IV. ESTADO DEL ARTE

En [6], describen un sistema un sistema desarrollado en Holanda para el control de cultivos, realizado en tres etapas:

- 1) control del clima que ejerce influencia sobre la planta (temperatura del aire, humedad, dióxido de carbono)
- 2) factores que influyen en el crecimiento de la planta, a corto plazo (transpiración, respiración, fotosíntesis, CO₂),
- 3) el desarrollo de cultivos a largo plazo (con especial interés sobre la calidad, rendimiento y tiempo de cosecha, y el valor económico).

Otra alternativa para el control de las variables de un Invernadero, que se esta implementando en invernaderos del norte de Portugal se presenta en [7]. Su propuesta consiste en el desarrollo e implementación de un sistema de control y adquisición de datos distribuido, sus nodos se encargan de sensorizar y comunicarse por medios inalámbricos los datos recolectados, como temperatura y humedad de los diversos invernaderos.

Desde otra perspectiva, en [8] se puede encontrar una muy buena reflexión acerca de como aumenta el interés en el desarrollo de métodos para detectar automáticamente el estrés

de los cultivos, uso del agua, crecimiento y la nutrición en los cultivos de invernadero, con la promesa de una mejor gestión del clima para un mejor rendimiento y calidad, así como reducción del impacto ambiental, sin embargo, dependerá del desarrollo de software que pueda incorporar datos de los cultivos en tiempo real, así como también de modelos más dinámicos.

Otra tendencia, como la que se propone en [9], incorpora la Inteligencia Artificial al proceso de control de variables, intentando obtener una mejor distribución de temperatura en el cultivo de un invernadero.

En [10], se propone un sistema de monitoreo capaz de obtener información de temperatura, humedad, intensidad de luz y niveles de dióxido de carbono, que tiene como finalidad ayudar a los agricultores a aumentar el rendimiento, mejorar la calidad y puntualidad de los cultivos a partir de la recopilación de información de variables climáticas del entorno de un invernadero.

Como se puede apreciar, existen variadas técnicas para contruir un sistema de control de variables climáticas como las que requiere un invernadero, con el objetivo de bajar el consumo de energéticos, e incrementar la producción [11]. Sin embargo, generalmente estos sistemas resultan caros y no pueden ser aplicados a invernaderos con una producción pequeña.

V. MATERIALES Y MÉTODOS

Se pretende realizar el diseño y acondicionamiento de prototipos para la automatización de un Invernadero con el microcontrolador de Intel galileo y arduino.

El primer prototipo es el control automático de apertura y cierre de ventanas del Invernadero que se realizara mediante un motor de 24V500W, 1800RPM que se implementó con el PWM (SHT-146), que es un control reversible y de velocidad para motores 6V- 30V 6A (véase la figura 1).

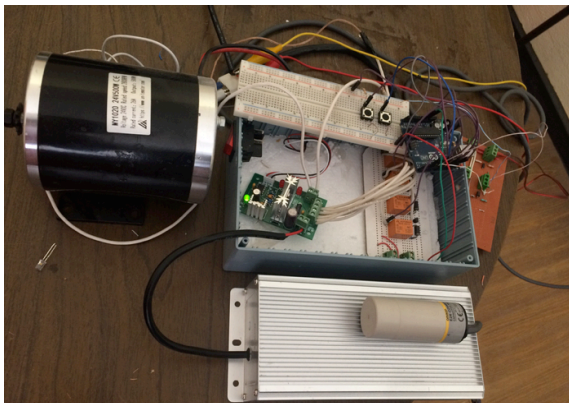


Fig. 1. Motor, prototipo de control automático, y sensores de proximidad capacitivos.

El segundo prototipo consiste en el acondicionamiento e instalación de dos sensores de proximidad capacitivos (E2K-C25MF1) ubicados estratégicamente para detectar el cierre y apertura total de las ventanas del invernadero.

Estos prototipos mantendrán una comunicación inalámbrica vía Xbee con una sonda de humedad relativa y temperatura (HMP60 - Vaisala) ubicada en el centro del Invernadero, para monitorear y optimizar la humedad relativa del Entorno. La Sensorización para el control de la apertura y cierre de persianas así como para el control de humedad y temperatura relativa del Invernadero, se energizara con un control de carga solar (EPRC- 12/24V 10 Amp) para un eficiente consumo de energía.

VI. SENSORIZACIÓN

Se desarrolló el módulo de sensorización, que consiste de dos partes:

A. Emisor

- Un primer módulo emisor, que consiste de un microcontrolador Arduino Uno, inicialmente dotado de un sensor de temperatura Keyes DS18B20, encargado de hacer la adquisición de la temperatura, registrando cualquier variación en el Invernadero (véase la figura 3). Además, se instaló un microcontrolador Xbee, que doto de comunicación inalámbrica al módulo, de tal manera, que al registrar cambios de temperatura, estos son enviados a un segundo módulo receptor.
- La comunicación inalámbrica proporcionada por los Xbee, fue una restricción dada la distancia que hay del centro del invernadero hasta las esquinas, que es donde se sitúan los mecanismos manuales de apertura-cierre de las ventanas. La alimentación de energía de este módulo Emisor está a cargo de una celda solar.
- Calibración de la sonda Vaisala HMP60. La sonda envía voltaje como producto de la lectura de temperatura y humedad. Se realizó el algoritmo para convertir los valores expresados en voltaje por la sonda a temperatura. Se verificaron las lecturas obtenidas del sensor Keyes DS18B20, la sonda Vaisala y un termómetro analógico, con una diferencia menor a medio grado centígrado entre las tres lecturas. Una vez verificada la entrada, se sustituyó el sensor Keyes DS18B20 empleado para prototipado por la sonda Vaisala, de uso industrial (véase la figura 2).



Fig. 2. Calibración de sonda Vaisala con respecto al termómetro que tienen en el Invernadero



Fig. 3. Emisor. Note un microcontrolador Arduino Uno, alimentado por energía proveniente de una celda solar. Véase el cable que sube, es la sonda Vaisala, que hace la medición de la temperatura.

B. Receptor

- El receptor está integrado por un microcontrolador Galileo Generación 2 con sistema operativo Yocto, las lecturas de temperatura llegan desde remoto a través de un microcontrolador Xbee (véase la figura 4).
- Configuración de límites. El microcontrolador Galileo cuenta con un servidor web y algunos scripts desarrollados en Python, que permiten configurar desde una aplicación web, los límites de temperatura.



Fig. 4. Receptor Galileo. Véase la interfaz ethernet que reenvía las mediciones a un servidor remoto.

VII. MONITOREO

Una parte importante del proyecto, desde el punto de vista de la computación ubicua, consiste en el seguimiento de los registros de temperatura. Dado que el invernadero se localiza en una escuela, se habilitó el servicio de internet.

Dada la limitada capacidad de procesamiento y almacenamiento de microcontrolador Galileo, se desarrolló un conjunto de web services WSDL que fueron instalados en un servidor remoto público.

Cada vez que al receptor llega una temperatura proveniente del emisor, y una vez que toma una decisión en cuanto a la corrección de la temperatura, ésta es enviada al servidor remoto vía web service.

- Límite inferior define la temperatura mínima permitida en la que la planta puede estar sin estrés.
 - Límite superior define la temperatura máxima.
 - Ideal, define la temperatura óptima para el tipo de cultivo
- El objetivo de los límites es controlar la apertura y cierre de las ventanas, de acuerdo a las lecturas de temperatura que le llegan. El algoritmo de control, pieza faltante en este rompecabezas, intentará hacer que la temperatura se acerque de la mejor manera posible, a la temperatura ideal. Básicamente, a medida que se vaya acercando a los límites máximos y mínimos, la corrección de la temperatura será más frecuente.
 - Las ventanas tienen unos sensores que permiten conocer si la ventana está completamente abierta o completamente cerrada, en cuyo caso, el motor ya no acciona la ventana solo en ese sentido.

Se tomo la decisión de emplear XML como lenguaje de representación de datos, y la de emplear web services, obedece a la idea de presentar la información a través de diferentes tipos de interfaces. Se desarrollo una página web que consume los servicios y que muestra los reportes de la temperatura (véase la figura 5 y 6).

Temperature Status

| Temperature on 2018-02-15 between 18:00 and 19:00 | | |
|---|-------------------|--------------------|
| Lower | Average | Higger |
| 26.989999771118164 | 27.29746399655867 | 27.420000076293945 |

Fig. 5. Temperatura Máxima, Promedio y Mínima de un rango de horas en un día en específico.

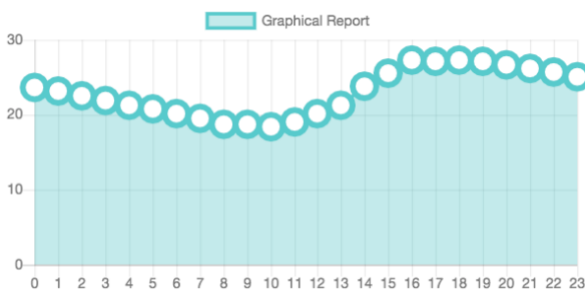


Fig. 6. Gráfica con las temperaturas promedio por cada hora de un día

VIII. PREDICCIÓN

A la página web se le ha añadido un plugin que proporciona el pronóstico del tiempo. Esta parte es importante, ya que puede permitir al productor preparar el combustible que requiere para evitar por ejemplo, la perdida total de la producción por una helada (véase la figura 7).



Fig. 7. Plugin para predicción del tiempo que fue agregado a la página desde la que se puede hacer el monitoreo

IX. CONCLUSIONES Y TRABAJO FUTURO

Se ha presentado una propuesta para el control de temperatura en un invernadero, basada en la experiencia obtenida a partir del trabajo con cómputo ubicuo. Se elaboró un prototipo que ya tiene un periodo de prueba de alrededor de

tres meses, restando para su implementación en el invernadero el algoritmo de control, mismo en el que ya se esta trabajando. Adicionalmente, se esta trabajando en el desarrollo de una aplicación para dispositivos móviles, que a partir de los web services, permita el monitoreo del invernadero.

Se puede concluir que la aportación de esta tecnología a los invernaderos de la región, es relativamente económica con respecto a los productos comerciales. Proporcionara un control automático ahorrando costo de combustibles, y principalmente, les ahorrara tiempo al no tener que estar una persona rigurosamente todo el día subiendo o bajando las ventanas.

Se espera, una vez que la solución este totalmente implementada, poder hacerla extensiva a los demás invernaderos de la región.

AGRADECIMIENTOS

Este proyecto de investigación fue posible gracias al patrocinio del Instituto Tecnológico Superior Zacatecas Norte y Tecnológico Nacional de México, a través de los proyectos 358.15-PD, 359.15-PD y al programa de Fortalecimiento de Cuerpos Académicos (IDCA-28030).

REFERENCIAS

- [1] L. E. Bernal, A. F. Rumayor, and O. Pérez, "La competitividad sistémica de la industria del tomate de agricultura protegida en Zacatecas," *Mercados y Negocios* (1665-7039), No. 18, pp. 38-59, 2016.
- [2] F. Rodríguez, M. Berenguel and M. R. Arahall, "A hierarchical control system for maximizing profit in greenhouse crop production," *European Control Conference (ECC)*, pp. 2753-2758, 2003.
- [3] R. F. Tap, L. G. Van Willigenburg and G. Van Straten, "Receding horizon optimal control of greenhouse climate based on the lazy man weather prediction," *IFAC Proceedings Volumes*, vol. 29 (1), pp. 889-894, 1996.
- [4] M. Berenguel, L. J. Yebra and F. Rodríguez, "Adaptive Control Strategies For Greenhouse Temperature Control," *European Control Conference (ECC)*, pp. 2747-2752, 2003.
- [5] R. Gomme, "Climate-related risk in agriculture," A note prepared for the IPCC Expert Meeting on Risk Management Methods, Toronto, Canada, 1998.
- [6] A. J. Udink, G. P. A. Bot and J. J. Van, "Computer control of greenhouse climates," *Symposium on Potential Productivity in Protected Cultivation* 87, pp. 265-272, 1978.
- [7] C. Serodio, J. B. Cunha, R. Morais, C. Couto and J. Monteiro, "A networked platform for agricultural management systems," *Computers and electronics in agriculture*, vol. 31(1), pp. 75-90, 2001.
- [8] D. Ehret, A. Lau, S. Bittman, W. Lin and T. Shelford, "Automated monitoring of greenhouse crops," *Agronomie*, vol. 21(4), pp. 403-414, 2001.
- [9] N. Sigrimis, K. G. Arvanitis, G. D. Pasgianos and K. P. Ferentinos, "Computer integrated management and intelligent control of greenhouses," *Environment Control in Biology*, vol. 40(1), pp. 39-53, 2002.
- [10] S. Yang and D. S. Simbeye, "Computerized Greenhouse Environmental Monitoring and Control System Based on LabWindows/CVI.," on *JCP*, vol. 8(2), pp. 399-408, 2013.
- [11] J. M. Aaslyng, J. B. Lund, N. Ehler and E. Rosenqvist, "IntelliGrow: a greenhouse component-based climate control system," *Environmental Modelling & Software*, vol. 18(7), pp. 657-666, 2003.

MRAC basado en la regla MIT y el método de Lyapunov para el control en altitud de un UAV de ala fija

T. Espinoza-Fraire, A. Saenz-Esqueda, J. Rentarías-Soto, F. Cortes-Martinez
 Facultad de Ingeniería, Ciencias y Arquitectura
 Universidad Juárez del Estado de Durango
 Gómez Palacio, Durango, México
 Email: atespinoza@ujed.mx

Resumen—Este trabajo presenta la comparación del método de Lyapunov y el método de la regla MIT para la obtención del ajuste de las ganancias de un controlador PD, la obtención del ajuste de las ganancias esta basado en la metodología de control adaptable por modelo de referencia. El controlador PD adaptable diseñado, se aplica al control en altitud de un vehículo aéreo no tripulado de ala fija sometido a perturbaciones por ráfagas de viento. Los resultados obtenidos con el controlador PD adaptable se presentan en simulaciones utilizando el programa MatLab.

I. INTRODUCCIÓN

El diseño y control de los vehículos aéreos no tripulados o por sus siglas en inglés UAV (Unmanned Aerial Vehicle), es un tema de estudio en la teoría de control automático, como también para los ingenieros en aeronáutica, entre otras ramas de ciencia e ingeniería que pueden aportar a este tipo de sistemas aéreos no tripulados. El tema de los UAVs es interesante debido a que estos robots aéreos están sometidos a las mismas inclemencias climáticas a las que se encuentran los transportes aéreos como son los aviones y helicópteros [1].

Un aspecto importante que esta presente durante el vuelo de cualquier aeronave son las ráfagas de viento, las cuales son de diferentes magnitudes y velocidades [2]. Considerando lo anterior, en este trabajo se diseña un controlador adaptable para un vehículo aéreo no tripulado de ala fija perturbado por ráfagas de viento, el objetivo de control es que el UAV de ala fija converja a diferentes alturas deseadas.

Entonces, en este trabajo se diseña un controlador adaptable PD (Proporcional-Derivativo), el cual esta diseñado basados en el modelo de referencia con la metodología de la regla MIT y por la metodología de Lyapunov, estos métodos son comparados a fin de conocer cual tiene un mejor funcionamiento para controlar la altura de un UAV de ala fija perturbado por ráfagas de viento. Cabe mencionar, que las perturbaciones por ráfagas de viento son desconocidas por el controlador adaptable. La parte adaptable del controlador propuesto en este trabajo, son las ganancias proporcional y derivativa que conforman al controlador PD.

En la literatura se pueden encontrar una variedad de trabajos donde se aplica la metodología por modelo de referencia con la regla MIT y Lyapunov, por ejemplo, en [3] se utiliza la

metodología de la regla MIT para una turbina *Steam*, el control de una turbina con un gobernador es esencial, ya que las turbinas necesitan evitar daños y algunas aplicaciones como en la generación de electricidad, ya que requieren un control de velocidad preciso y un controlador adaptable por modelo de referencia basado en la regla MIT es propuesto para la solución de dicho problema. En [4] se presenta el sistema de control adaptable por modelo de referencia basado en la regla del MIT para un humidificador de manera de tener un cierto control sobre el contenido de humedad en la fibra de algodón. Por otro lado, un controlador adaptable por modelo de referencia basado en la metodología de Lyapunov es utilizando en [5], el cual muestra ciertas consideraciones que se requieren para decir que un sistema sea adaptable y algunas condiciones necesarias para el sistema de control adaptable. En [6] es presentado un controlador adaptable por modelo de referencia basado en la metodología de Lyapunov, el cual es aplicado un reactor el cual muestra características no lineales y cuyos parámetros cambian durante un período de tiempo por una variedad de razones mostradas en [6].

La organización del artículo es la siguiente: la sección II muestra las ecuaciones que definen el modelo dinámico para la dinámica longitudinal del vehículo aéreo no tripulado de ala fija; la sección III trata sobre la ley de control desarrollada para el control en altitud del UAV de ala fija. En la sección IV se muestran los resultados obtenidos en simulación. Finalmente, en la sección V se tienen las conclusiones y el trabajo a futuro.

II. MODELO DINÁMICO

En esta sección se presenta el modelo longitudinal del vehículo aéreo no tripulado para llevar a cabo el control de la altitud del vehículo aéreo no tripulado de ala fija.

Cabe mencionar que, el modelo aerodinámico que se presenta para la dinámica longitudinal del vehículo aéreo no tripulado, se obtuvo considerando que no existe alguna parte flexible sobre éste, y además se considera la tierra como plana, lo anterior debido a que el vehículo aéreo no tripulado sólo recorrerá distancias cortas, es por ello que en el modelo matemático de dicha aeronave no tripulada, no se considera la curvatura terrestre. Entonces, la obtención del modelo es

a partir de la segunda ley de movimiento de Newton. A continuación, se presenta el modelo longitudinal que define al vehículo aéreo no tripulado de ala fija:

$$\dot{V} = \frac{1}{m}(-D + T \cos \alpha - mg \sin \gamma) \quad (1)$$

$$\dot{\gamma} = \frac{1}{mV}(L + T \sin \alpha - mg) \sin \gamma \quad (2)$$

$$\dot{\theta} = q \quad (3)$$

$$\dot{q} = M_q q + M_{\delta_e} \delta_e \quad (4)$$

$$\dot{h} = V \sin(\theta) \quad (5)$$

donde V es la velocidad de vuelo, α describe el ángulo de ataque, γ representa el ángulo de incidencia del viento y θ denota el ángulo de cabeceo. Además, q es la velocidad angular de cabeceo (con respecto al eje y en el cuerpo del vehículo aéreo no tripulado de ala fija), T denota la fuerza de empuje del motor, h es la altura de la aeronave no tripulada y δ_e representa la desviación de la superficie de control conocida en aerodinámica como elevador [7].

Los efectos aerodinámicos en el vehículo aéreo son obtenidos por la fuerza de elevación L y la fuerza de oposición al movimiento D . La masa total del avión esta dada por m , g es la constante gravitacional y I_{yy} describe la componente y de la diagonal de la matriz de inercia. El valor del ángulo de ataque es obtenido utilizando la siguiente relación $\alpha = \theta - \gamma$ [7]. En aerodinámica, M_q y M_{δ_e} son las derivadas de estabilidad implícitas en el movimiento de cabeceo. La fuerza de elevación L , la fuerza D son definidas como [7]:

$$L = \bar{q} S C_L \quad (6)$$

$$D = \bar{q} S C_D \quad (7)$$

donde \bar{q} denota la presión aerodinámica. S representa el área del ala y \bar{c} es la respuesta de la cuerda. C_D y C_L son los coeficientes aerodinámicos para las fuerzas de sustentación y de oposición, respectivamente. Las derivadas de estabilidad aerodinámica son definidas por:

$$M_q = \frac{\rho S V \bar{c}^2}{4 I_{yy}} C_{m_q}$$

$$M_{\delta_e} = \frac{\rho V^2 S \bar{c}}{2 I_{yy}} C_{m_{\delta_e}}$$

donde:

ρ : Densidad del aire (1.05 kg/m³).

S : Área del ala (0.09 m²).

\bar{c} : Respuesta estándar de la cuerda (0.14 m).

b : Envergadura, (0.914 m).

I_{yy} : Momento de inercia en cabeceo (0.17 kg · m²).

C_{m_q} : Coeficiente adimensional para el movimiento longitudinal, obtenido de forma experimental (-50).

$C_{m_{\delta_e}}$: Coeficiente adimensional para el movimiento del elevador, obtenido de forma experimental (0.25).

III. DISEÑO DEL CONTROLADOR

Con el objetivo de diseñar el controlador PD adaptable para la altura, se ha considerado las ecuaciones (3)-(5), esto es

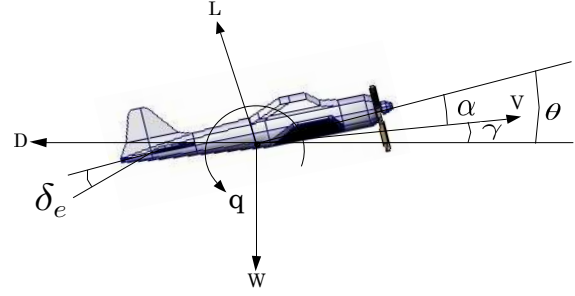


Figura 1. Movimiento de cabeceo

debido a que la ecuación(1) representa la velocidad de la aeronave no tripulada, pero para las simulaciones realizadas en este trabajo se consideró una velocidad de vuelo constante, y la ecuación (2) es el ángulo generado por la incidencia del viento y en este trabajo el controlador es diseñado sobre el sólido (el cuerpo del vehículo aéreo no tripulado). Entonces, el error de altura esta definido como $\tilde{e}_h = h_d - h$, donde h_d es la altura deseada y h es la altura actual a la que se encuentra el vehículo aéreo no tripulado.

La altura deseada is alcanzada controlando el ángulo de cabeceo de la aeronave no tripulada, por lo tanto, se define un error para este ángulo $\tilde{e}_\theta = \theta_d - \theta(t)$, donde $\theta_d = \arctan(\tilde{e}_h/\zeta)$ es el ángulo de cabeceo deseado, y ζ denota la longitud desde el centro de masa del vehículo aéreo hasta la nariz del mismo.

Para el diseño del controlador se consideran las ecuaciones (3)-(4), δ_e define la entrada de control. Así, el controlador adaptable está dado por:

$$\delta_e = \hat{k}_{pm} \tilde{e}_\theta + \hat{k}_{vm} \dot{\tilde{e}}_\theta \quad (8)$$

donde \hat{k}_{pm} y \hat{k}_{vm} son llamadas como las ganancias de posición y de velocidad, respectivamente, estas son las ganancias adaptables. Las ganancias del controlador PD adaptable tiene implícito un sub índice para indicar que metodología es la utilizada para la adaptación de las ganancias, el mismo sub índice será utilizado para las ganancias de adaptación γ , por lo tanto, $m = m_1, m_2$:

- m_1 Corresponde al método de la regla MIT.
- m_2 Corresponde al método de Lyapunov.

Por otro lado, para el diseño del controlador por los métodos de la regla MIT y por Lyapunov, se introduce un error dado por:

$$e_{\theta_m} = \theta_m - \theta \quad (9)$$

donde θ_m es la salida desde el modelo de referencia. Siguiendo los métodos presentados en [8], el modelo matemático ha sido transformado en la representación de una función de transferencia con el objetivo de obtener las derivadas de sensibilidad; calculando las derivadas parciales con respecto a los parámetros del controlador \hat{k}_{pa} and \hat{k}_{va} . Entonces, la

función de transferencia en lazo cerrado con el controlador PD adaptable está definida por:

$$\theta = \frac{M_{\delta_e}(\hat{k}_p + \hat{k}_v s)}{s^2 + (M_q + M_{\delta_e} \hat{k}_v)s + M_{\delta_e} \hat{k}_p} \theta_d \quad (10)$$

y el modelo de referencia para la dinámica en altitud ha sido definida como:

$$\theta_m = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \theta_d \quad (11)$$

donde $\zeta = 3,17$ y $\omega = 3,16$. Considerando (9)-(11) y calculando las derivadas parciales con respecto a \hat{k}_{pa} y \hat{k}_{va} , se obtiene:

$$\frac{\partial e_{\theta_m}}{\partial \hat{k}_p} = \frac{M_{\delta_e}}{s^2 + (M_q + M_{\delta_e} \hat{k}_v)s + M_{\delta_e} \hat{k}_p} (\theta - \theta_d) \quad (12)$$

$$\frac{\partial e_{\theta_m}}{\partial \hat{k}_v} = \frac{M_{\delta_e} s}{s^2 + (M_q + M_{\delta_e} \hat{k}_v)s + M_{\delta_e} \hat{k}_p} (\theta - \theta_d) \quad (13)$$

Generalmente, las expresiones (12) y (13) no pueden ser utilizadas debido a los parámetros desconocidos \hat{k}_{pa} y \hat{k}_{va} . Así que, se asume un caso óptimo y se define como:

$$s^2 + (M_q + M_{\delta_e} \hat{k}_{vl})s + M_{\delta_e} \hat{k}_{pl} = s^2 + 2\zeta\omega_n s + \omega_n^2 \quad (14)$$

Después de estas aproximaciones, se obtienen las ecuaciones diferenciales para la adaptación de las ganancias del controlador PD:

$$\dot{\hat{k}}_{pm1} = -\gamma_{1m1} \left(\frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} (\theta - \theta_d) \right) e_{\theta_m} \quad (15)$$

$$\dot{\hat{k}}_{vm1} = -\gamma_{2m1} \left(\frac{s}{s^2 + 2\zeta\omega_n s + \omega_n^2} (\theta - \theta_d) \right) e_{\theta_m} \quad (16)$$

Para obtener la adaptación de las ganancias para el controlador PD por el método de Lyapunov, las ecuaciones que definen a la dinámica de cabeceo pueden escribirse de forma matricial, entonces se tiene que:

$$\dot{x} = A_x x + B_x \theta_d \quad (17)$$

Así mismo el modelo de referencia esta dado por:

$$\dot{x}_m = A_m x_m + B_m \theta_d \quad (18)$$

La dinámica que describe la evolución del error 9 puede expresarse por:

$$\dot{e}_{\theta_m} = A_m e_{\theta_m} + (A_m - A_x) \underline{\theta} + (B_m - B_x) \theta_d \quad (19)$$

donde $e_{\theta_m} = [e_{\theta_m} \dot{e}_{\theta_m}]^T$ y $\underline{\theta} = [\theta \dot{\theta}]^T$. El punto de equilibrio $e_{\theta_m} = 0$ es asintóticamente estable si se elige la ley de adaptación como:

$$\dot{A}_x = \gamma P e_{\theta_m} \theta^T \quad (20)$$

$$\dot{B}_m = \gamma P e_{\theta_m} \theta_d \quad (21)$$

donde P es una matriz definida positiva, la cual es solución de la ecuación de Lyapunov $A_m^T P + P A_m = -Q < 0$. Así que, resolviendo para Q la matriz identidad se tiene que:

$$P = \begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix} = \begin{pmatrix} 25 & 200 \\ 200 & 2005 \end{pmatrix}$$

Finalmente, a partir de las ecuaciones 20-21 para obtener $\dot{\hat{k}}_{pm2}$ y $\dot{\hat{k}}_{vm2}$, la ley de adaptación para el controlador adaptable por el método de Lyapunov están dadas por:

$$\dot{\hat{k}}_{pm2} = -\gamma_{1m2} (p_{21} e_{\theta_m} + p_{22} \dot{e}_{\theta_m}) (\theta - \theta_d) \quad (22)$$

$$\dot{\hat{k}}_{vm2} = -\gamma_{2m2} (p_{21} e_{\theta_m} + p_{22} \dot{e}_{\theta_m}) \dot{\theta} \quad (23)$$

IV. RESULTADOS EN SIMULACIÓN

Para describir los resultados obtenidos en simulación con los métodos del MIT y Lyapunov para la adaptación de las ganancias del controlador PD, se utilizara la norma \mathcal{L}_2 [9] para conocer que controlador muestra un menor error de convergencia al valor deseado. Así, la norma \mathcal{L}_2 para el error está definida como:

$$\mathcal{L}_2[e_h] = \sqrt{\frac{1}{T - t_0} \int_{t_0}^T \|e_h\|^2 dt} \quad (24)$$

La norma \mathcal{L}_2 será utilizada nuevamente para el análisis del esfuerzo del controlador y obtener que metodología con el controlador PD adaptable genera mayor esfuerzo de control para alcanzar la altura deseada por el vehículo aéreo no tripulado, y esta definida por:

$$\mathcal{L}_2[\delta_e] = \sqrt{\frac{1}{T - t_0} \int_{t_0}^T \|\delta_e\|^2 dt} \quad (25)$$

Así, con el uso de las normas (24) y (25) se obtiene la Tabla I.

Cuadro I
NORMA \mathcal{L}_2 PARA LOS ERRORES Y ESFUERZOS DEL PD ADAPTABLE EN ALTURA

| Método | Altura | |
|----------|---------------------------|-----------------------------------|
| | $\mathcal{L}_2[e_h](mts)$ | $\mathcal{L}_2[\delta_e](grados)$ |
| MIT | 0.2294 | 0.1438 |
| Lyapunov | 0.2099 | 0.0028 |

En la Figura 2 se muestran los resultados obtenidos del controlador PD adaptable, en la misma figura se tienen diferentes alturas (línea punteada) a las que debe de converger el vehículo aéreo no tripulado. Se considera que el control automático en altitud se activa cuando la aeronave a alcanzado una altura considerable de vuelo. Basados en la Tabla I, el controlador PD adaptable basado en el método de Lyapunov muestra un error 8,52% menor que el diseño de las ganancias de adaptables por el método de la regla MIT.

Las señal de salida del controlador PD con las ganancias adaptables obtenidas por el método de la regla MIT se presentan en la Figura 3. En la Figura 4 se aprecia la señal de control generada por el controlador PD con las ganancias de adaptación obtenidas por el método de Lyapunov. Considerando los resultados obtenidos en la Tabla I, la obtención de la ley de adaptación para las ganancias del controlador PD por el método de Lyapunov es 98,06% menor el esfuerzo del controlador que el obtenido por el método de la regla MIT.

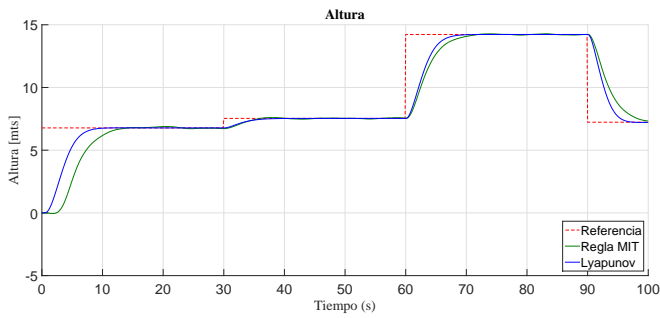


Figura 2. Respuesta MRAC basado en los métodos de la Regla MIT y Lyapunov

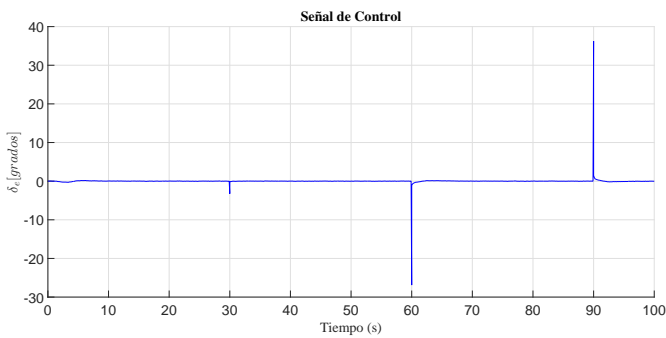


Figura 3. Respuesta de control del MRAC basado en el método de la Regla MIT

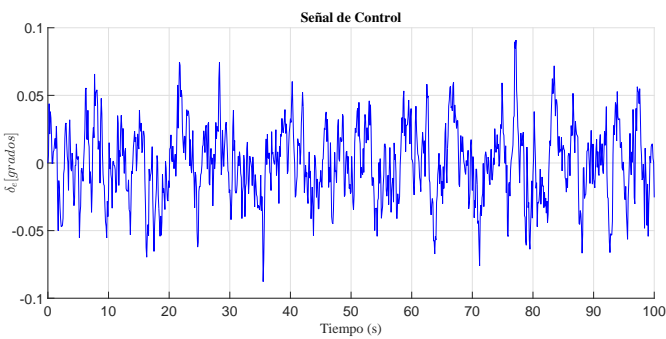


Figura 4. Respuesta de control del MRAC basado en el método de Lyapunov

En las Figuras 5 y 6 se muestra la convergencia de las ganancias adaptables para el controlador PD obtenidas por el método de la regla MIT y Lyapunov, respectivamente.

En la Figura 7, se presenta un acercamiento en las respuestas de los controladores MRAC basados en la metodología por la regla MIT y por el método de Lyapunov, sepuede apreciar como es afectado por las ráfagas de viento el controlador PD adaptable basado en la regla del MIT. En la misma figura se puede percibir que el controlador basado en en el método de lyapunov muestra una mejor respuesta al permanecer más cerca de la referencia deseada y con afectación muy mínima por las ráfagas de viento.

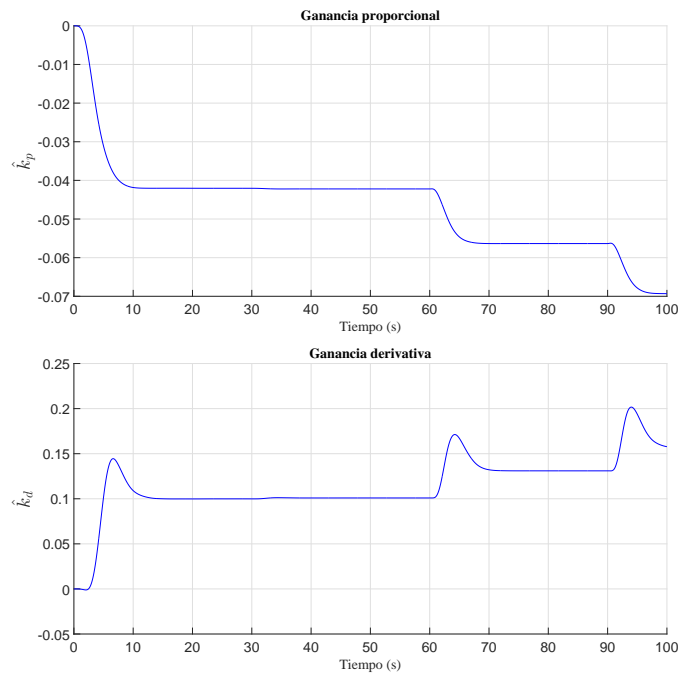


Figura 5. Convergencia de las ganancias adaptables con el método de la Regla MIT al lograr el objetivo de control

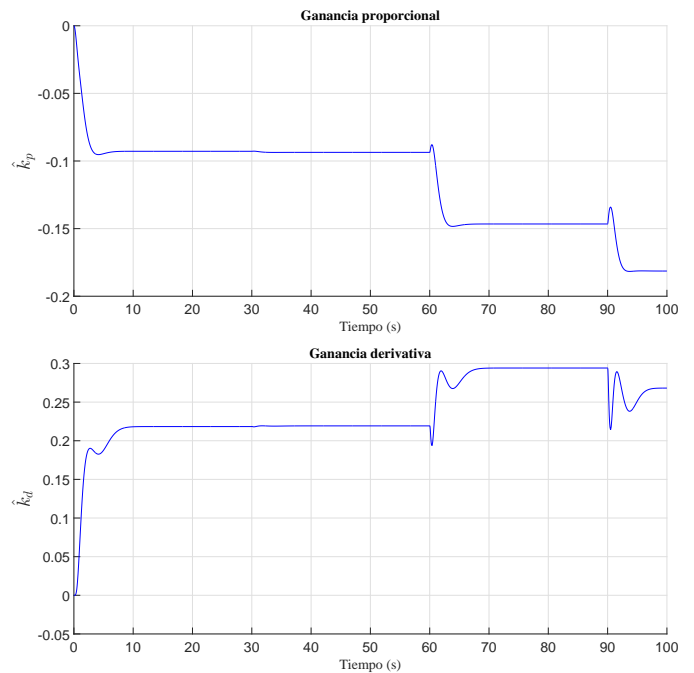


Figura 6. Convergencia de las ganancias adaptables con el método de Lyapunov al lograr el objetivo de control

V. CONCLUSIONES

Considerando los resultados obtenidos después de varias simulaciones con un controlador PD adaptable para el control en altitud de un vehículo aéreo no tripulado de ala fija sujeto a

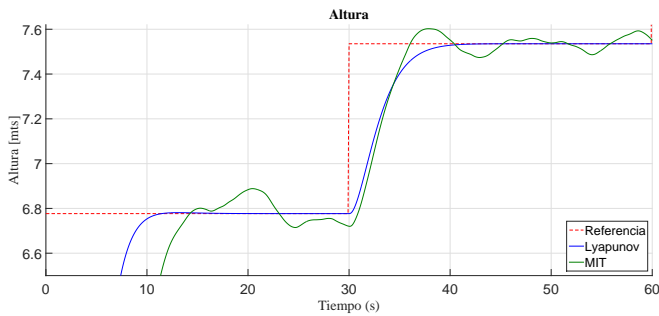


Figura 7. Respuesta del MRAC basado en el método de la Regla MIT y por el método de Lyapunov en presencia de ráfagas de viento

perturbaciones por ráfagas de viento. El controlador adaptable basado en el método de Lyapunov mostró una mejor respuesta en error y en esfuerzo del controlador.

El controlador PD diseñado las ganancias adaptables por el método de la regla MIT, converge a las alturas deseadas, pero presentó un error y un esfuerzo del controlador mayor que el diseño de las ganancias por el método de Lyapunov.

Para el trabajo futuro se realizará el diseño de estos métodos para el movimiento de guiñada y alabeo del vehículo aéreo no tripulado de ala fija.

AGRADECIMIENTOS

Los autores agradecen el apoyo para el desarrollo y presentación de este trabajo a la Facultad de Ciencias, Ingeniería y Arquitectura de la Universidad Juárez del Estado de Durango (FICA-UJED).

REFERENCIAS

- [1] M. V. Cook. Flight Dynamics Principles, Second edition, Ed. Elsevier, 2007
- [2] R. Bellasio. "Analysis of wind data for airport runway design", Journal of Airline and Airport Management JAIRM, Volume 4, Number 2, pp. 97-116, 2014
- [3] S. Kochummen, E. Steam, A. Nasar. "Model Reference Adaptive Controller Designs of Steam Turbine Speed Based on MIT Rule", International Conference on Control, Communication and Computing India (ICCC), 19-21 November, India, 2015
- [4] T.Garikayi, L. Nyanga, T. Mushiri, S. Mhlanga, A. Muzinda, T. Mutangi. "Model Reference Adaptive Control System for Moisture Regulation in Cotton Ginning", SAIIE25 Proceedings, 9-11 July, South Africa, 2013
- [5] A. Misbawu, K. Edmund, E. Shajrouz. "Adaptive Control Strategy using Lyapunov Stability Theory", International Journal of Engineering Research and Technology (IJERT), Vol. 3, Issue 9, 2014
- [6] S. Anbu, N. Jaya. "Design of Adaptive Controller Based on Lyapunov Stability for a CSTR", World Academy of Science and Technology International Journal of Electronics and Communication Engineering, Vol: 8, Issue 1, 2014
- [7] L. Stevens and L. Lewis, Aircraft Control and Simulation, Ed. Jhon Wiley and Sons, 1992
- [8] K. J. Åström and B. Wittenmark, Adaptive Control, 2nd Edition, Ed. Prentice Hall, 1994
- [9] H. Khalil, Nonlinear Systems, Ed. Prentice Hall, ISBN: 0-13-067389-7, 1996

Síntesis de controladores neuronales para locomoción y auto-ensamblado en robots modulares MARBot

Jorge Padilla*, Eddie Clemente[§], Marlen Meza-Sánchez**, Cristina Ramírez-Fernández[§], I. Hernández-Capuchin[§]

* Instituto Tecnológico de Ensenada

Bld. Tecnológico 150, Ex-ejido Chapultepec,
Ensenada, B.C., México 22780

Email: jorgepa85@gmail.com, {eclemente, cramirez, ihernandez}@ite.edu.mx

[§]División de Estudios de Posgrado e Investigación

**marlen.meza.sanchez@gmail.com

Resumen—En este trabajo se aborda el diseño de neurocontroladores, para resolver los problemas de auto-ensamblaje y locomoción de cuerpo completo, sobre robots móviles modulares y autoreconfigurables tipo MARBot. El método propuesto se encuentra basado en el diseño de una red neuronal perceptrón multicapa y un Generador Central de Patrones, aplicando el paradigma del cómputo evolutivo para la optimización de sus parámetros. Se presentan resultados de simulación utilizando los softwares V-REP y MATLAB, los cuales muestran el auto-ensamblaje de dos módulos de MARBot y la locomoción de la estructura formada.

I. INTRODUCCIÓN

La robótica colaborativa otorga un mayor potencial para la realización de tareas complejas donde es necesario el uso de dos o más robots (agentes). De acuerdo con [1], el comportamiento colaborativo se define como *dada una tarea especificada por un diseñador, un sistema multi-robot muestra un comportamiento colaborativo si, debido a algún mecanismo subyacente (es decir, el mecanismo de cooperación), permite un aumento en la utilidad total del sistema*. Por lo tanto, una condición para llevar a cabo la robótica colaborativa es que cada agente de manera independiente cuente con la capacidad de comunicación con los demás elementos del sistema; esto es, cada agente perteneciente al conjunto puede obtener información del estado de los demás robots para lograr una tarea en coordinación. De esta forma, es posible llevar a cabo tareas complejas como la creación de formaciones para la navegación en conjunto, transporte, supervisión o patrullaje, entre otros (véase por ejemplo [2], [3] y sus referencias).

Un claro ejemplo de robótica colaborativa se encuentra en el uso de los robots modulares. Un robot modular es un sistema formado por módulos independientes, homogéneos e interconectados, capaces de modificar sus conexiones con otros módulos para formar una estructura apta para un objetivo dado, dándoles la propiedad de auto-reconfigurabilidad. A su vez, cada módulo debe estar equipado computacionalmente con la capacidad de sensado, comunicación y actuación [4].

Existen dos problemas de locomoción para esta clase de robots [5]. El primer tipo se refiere al auto-ensamblaje de los módulos y su auto-reconfiguración para establecer una morfología; que puede ser de tipo cadena (estructura 1D), o bien de tipo malla en dos o tres dimensiones (estructuras 2D y 3D). El segundo tipo de locomoción, se refiere al desplazamiento al movimiento de la estructura formada por el auto-ensamblaje de los módulos (cuerpo completo), o bien, de la morfología establecida después de una auto-reconfiguración. En el primer caso, el problema de auto-ensamblaje requiere que los módulos se desplacen para coincidir tanto en la orientación del mecanismo de sujeción, como a un punto en el espacio en donde se mueven. La forma de desplazamiento depende de la estructura de cada módulo. Existen prototipos que se desplazan por medio de ruedas (locomoción tipo diferencial [6]), mientras que hay otros que dependen de una articulación para lograrlo mediante rastreo [4], [7]. En la locomoción de cuerpo completo de robots modulares se debe considerar la distribución y el número de gdl's de esta nueva estructura. Por lo tanto, el algoritmo de control a diseñar deberá adaptarse a la estructura ensamblada para definir una forma específica de desplazamiento y a la tarea por realizar.

I-A. Planteamiento del problema

En este trabajo se abordan problemas de locomoción en robots modulares MARBot (Modular Auto-Reconfigurable RoBot). Este es un nuevo robot diseñado y construido en la División de Estudios de Posgrado e Investigación del Instituto Tecnológico de Ensenada. Los módulos MARBot, Figura 1, se conforman de dos cuerpos unidos por una articulación universal tipo cardán cuyo movimiento depende de la actuación de dos servomotores. Poseen dos ruedas laterales de forma cuadrada con el objetivo de facilitar el acople con otros módulos, así como la capacidad de una locomoción diferencial.

La primer problemática que se aborda corresponde al auto-ensamblaje de dos módulos de MARBot implementan-

do una optimización de neurocontroladores. Se propone que el modelado de movimiento para cada módulo considere la posición y/o velocidad requerida por los actuadores, sin importar las condiciones iniciales del sistema completo. La implementación de redes neuronales para este problema de control es a través de una red neuronal perceptrón multicapa. El segundo problema de control abordado corresponde a la locomoción de la estructura formada al ensamblar los dos módulos MARBot. Este problema, basándose en la estructura conectiva de la morfología de cada módulo, se realiza a través de la implementación de Generadores Central de Patrones (CPG, por sus siglas en inglés). Los CPG son redes neuronales capaces de producir como salida señales rítmicas en forma de patrones (oscilaciones) a una señal simple de entrada [8]. Estas redes permiten relacionar la conectividad de sus neuronas con la morfología del robot y pueden ser sometidos a una optimización probabilística, [9], [10].

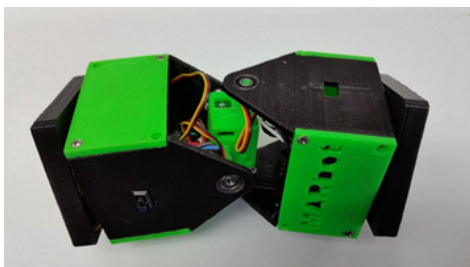


Figura 1: Prototipo de robot modular MARBot.

Tomando en cuenta que la estructura de esta clase de robots es dinámica y compleja, donde sus grados de libertad se encuentran distribuidos de forma no convencional, implementar un método de control clásico o determinístico es un problema difícil. Sin embargo, las redes neuronales perceptrón multicapa, pueden utilizarse como controladores viables para estos robots gracias a que, al igual que los CPG, pueden ser sometidos a una optimización de sus parámetros, [11], [12]. Así, el uso de neurocontroladores pueden utilizarse para resolver ambos problemas de locomoción sobre los robots modulares MARBot. Del estado del arte con respecto a procesos de optimización, se destaca la implementación de algoritmos evolutivos, véase [13],[14], entre otros, como una técnica de búsqueda de los parámetros adecuados para cada neurocontrolador.

En este sentido, se propone la implementación de algoritmos genéticos, como técnica perteneciente a los algoritmos evolutivos, aplicada sobre una red neuronal para la síntesis de dos tipos de neurocontroladores. El primer controlador desarrollado, una red neuronal perceptrón multicapa, ofrece una solución al problema de desplazamiento y auto-ensamblaje de dos módulos MARBot; mientras que el segundo neurocontrolador, una estructura basada en CPG, es dedicado a la locomoción de cuerpo completo de dos robots modulares MARBot.

I-B. Contribución

El robot modular MARBot es un desarrollo reciente, el cual aún carece de un procedimiento para su control. Así, en este trabajo se abordan los siguientes puntos:

- Se desarrolla una metodología para sintetizar controladores de locomoción en robots modulares MARBot.
- Se ofrece una alternativa de resolución del problema de auto-ensamblaje de dos módulos virtuales de MARBot para formar una estructura tipo cadena, aplicando neurocontroladores perceptrón multicapa.
- Generación de locomoción de la estructura formada en el proceso de auto-ensamblaje utilizando una red neuronal tipo CPG.
- Optimización de los parámetros de ambas redes neuronales mediante un Algoritmo Genético para coordinar el movimiento de los módulos MARBot.
- Simulación virtual del desplazamiento del robot mediante la plataforma V-REP y Matlab para demostrar la efectividad de la metodología propuesta.

El presente documento está organizado como se describe a continuación. La metodología para resolver los problemas de locomoción en robots modulares MARBot se describe en la sección II. Los resultados en simulación obtenidos de la solución a ambos problemas de control se presentan en la sección III. Finalmente, la discusión y conclusiones de este trabajo se realizan en la sección IV.

II. AUTOENSAMBLAJE Y LOCOMOCIÓN DEL ROBOT MARBOT

El diseño geométrico virtual de MARBot fue construido en la plataforma BLENDER® y posteriormente importado al simulador V-REP (por sus siglas en inglés *Virtual Robot Experimentation Platform*, desarrollo hecho por la empresa Coppelia Robotics). En esta etapa se le asignan los valores físicos aproximados de propiedades dinámicas del robot tales como momentos de inercia, matriz de inercia, fricción, posición de centros de masas. Asimismo, se asignan los movimientos articulares, sensores y el cuerpo del robot considerándolo uniforme.

El escenario en donde se desplazarán los robots es un plano XY donde la posición, la orientación y la velocidad de las ruedas de cada módulo, conforman un vector $\mathcal{I} = [x_1 \ x_2 \ y_1 \ y_2 \ v_1 \ v_2]^T$. El vector \mathcal{I} define las entradas del neurocontrolador dado por una red perceptrón multicapa para resolver el problema de auto-ensamblaje, y un CPG para el caso de locomoción de la estructura ensamblada. Se diseña un neurocontrolador independiente para cada robot con la misma definición en su vector de entrada \mathcal{I} . Esto permite que ambos robots reciban información de los estados del otro robot, y de esta forma logren la colaboración entre ambos módulos.

La optimización se implementa a través de un Algoritmo Genético (GA por sus siglas en inglés) utilizando el entorno de Matlab. El primer paso consiste en la inicialización de la población a través de la creación de una matriz con dimensiones $(numParams \times popsize)$, donde $popsize$ es el tamaño

de la población y $numparams$ el número de parámetros de cada posible solución. Posteriormente, se inicia el proceso de optimización por parte del neurocontrolador mediante la evaluación de su desempeño de acuerdo a una función objetivo. Esto es, cada individuo generado de la población es ingresado al neurocontrolador, donde los elementos de cada individuo corresponden a los pesos o umbrales de las neuronas de la red. La selección de la función objetivo y la evaluación de aptitud para cada individuo, depende del objetivo de control y de su capacidad para resolver el problema.

II-A. Auto-ensamblaje de dos módulos de MARBot

Considere un plano XY de $0,75m \times 0,75m$ donde los módulos son posicionados simétricamente opuestos con respecto al origen, como se muestra en la Figura 2. Las condiciones iniciales correspondientes a la posición, orientación y velocidad de cada módulo se definen como un vector de entradas inicial $\mathcal{I}(0) = [0.30 \text{ (m)}, -0.30 \text{ (m)}, 0.25 \text{ (m)}, -0.25 \text{ (m)}, 0 \text{ (rad/s)}, 0 \text{ (rad/s)}]^T$. Adicionalmente, se especifican las condiciones iniciales de orientación sobre el eje Z para cada módulo como $Z(0) = [z_1(0), z_2(0)]^T = [0(\text{rad}), 0(\text{rad})]^T$.

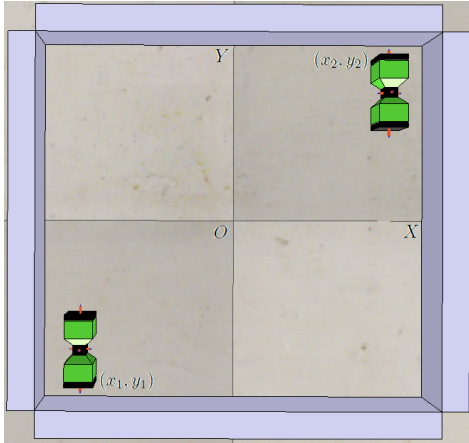


Figura 2: Construcción virtual en V-REP de los robots modulares MARBot en el escenario para auto-ensamblado, donde cada módulo se encuentra colocado en su posición inicial para la simulación.

Para resolver el problema de auto-ensamblaje de los dos módulos se implementan neurocontroladores definidos como redes neuronales perceptrón multicapa homogéneos con 6 neuronas de entrada para los estados de los módulos, 6 neuronas en la capa oculta, 2 neuronas de salida para las velocidades de las ruedas, y se establece que tanto los pesos como los umbrales pertenecen al rango $(-1, 1)$. Aunque cada módulo cuenta con un perceptrón individualmente, los estados de los dos módulos ingresaran a cada una de las redes neuronales. Lo anterior, permite emular la comunicación entre ambos módulos para desempeñarse colaborativamente. En la Figura 3 se observa la arquitectura propuesta para este caso, donde $P = \{p_1, p_2\} = \{(x_1, y_1), (x_2, y_2)\}$ son las posiciones normalizadas correspondientes a cada módulo, $Z = \{z_1, z_2\}$

las orientaciones del eje Z , y $V = Vel_1, Vel_2$ las velocidades de cada rueda.

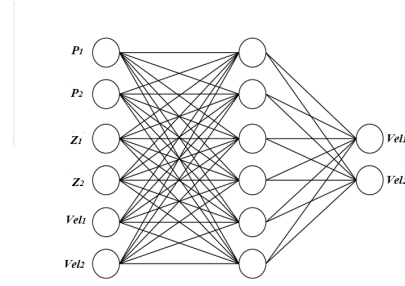


Figura 3: Estructura de la RNA perceptrón multicapa como controlador para el auto-ensamblaje de los robots modulares.

La arquitectura del perceptrón desarrollado se implementa en MATLAB mediante las funciones de activación comúnmente usadas, *ReLU* (unidad lineal rectificadora, $f(x) = \max(0, x)$) y *Sigmoidal* (definida como $f(x) = \text{sigmoid}(x) = 1/(1 + e^{-x})$). Por lo tanto, la red neuronal queda modelada como:

$$\begin{aligned} h &= w_1 \mathcal{I} + b_1 \\ z &= \max(0, h) \\ \text{output} &= \text{sig}(w_2 z + b_2) \\ V &= 2\pi \cdot \text{output} - \pi \end{aligned}$$

donde z es la salida de la capa oculta con la aplicación de la función de activación *ReLU*, \mathcal{I} la entrada definida anteriormente, $w = \{w_1, w_2\}$ son las matrices de pesos, $b = \{b_1, b_2\}$ los umbrales de cada capa de salida, y output es la salida del perceptrón una vez que ha sido aplicada la función de activación *sigmoidal*. Debido a que la aplicación de la función *sigmoidal* output obtiene valores normalizados entre 0, 1, las velocidades V de los robots se escalan a los valores de $-\pi, \pi$, que son los rangos otorgados para las velocidades de las ruedas. Dadas las dimensiones de las matrices $w_1 = (nHidden \times numInputs)$, $w_2 = (nOutputs \times nHidden)$, $b_1 = (nHidden \times 1)$, $b_2 = (nOutputs \times 1)$, es posible determinar el número de parámetros para la optimización del controlador; por lo tanto, se tiene que la dimensión del vector de parámetros es de 112 por ambas redes neuronales.

Se plantea un problema de minimización para la tarea de auto-ensamblaje, ya que se requiere la menor distancia y orientación entre las ruedas de cada módulo y un punto específico deseado P_d del escenario. Nótese que la estructura resultante del ensamblado de los dos módulos es un robot 1D en forma de serpiente. Esta observación será importante más adelante para resolver el problema de locomoción de la estructura formada. En este caso la función de aptitud u objetivo se define como

$$\begin{aligned} \min[f(\hat{P}, Z, P_d)] &= \|\hat{p}_1 - \hat{p}_2\| + \|z_1 - z_2\| \\ &+ \|P_d - \hat{p}_1\| + \|P_d - \hat{p}_2\|, \end{aligned}$$

donde $\hat{P} = \hat{p}_1, \hat{p}_2$ son las posiciones normalizadas de cada rueda, Z es el vector de las orientaciones y P_d la posición en el plano donde se desea que se realice el auto-ensamblaje. La Tabla I muestra las especificaciones del GA para este problema.

Tabla I: Parámetros utilizados por el GA durante el proceso de optimización de los neurocontroladores, aplicados al problema de auto-ensamblado de dos módulos MARBot.

| Parámetros de GA | |
|---------------------------|------------------|
| Tamaño de población | 74 |
| Tamaño de cromosoma | 112 |
| Tipo de selección | Rango |
| Tasa de élite | 5 % |
| Tipo de reproducción | Cruce aritmético |
| Tasa de reproducción | 80 % |
| Tasa mutación decremental | 40 % a 10 % |
| Generaciones | 74 |

II-B. Locomoción de dos módulos de MARBot

Una vez formada la estructura 1D mediante el auto-ensamblaje de dos módulos MARBot, se procede a optimizar una red neuronal CPG para controlar la posición de las articulaciones y generar un desplazamiento. Simultáneamente, se optimiza también una red neuronal artificial como función de retroalimentación para el CPG. Considere como escenario para este problema, el plano XY con dimensiones $2m \times 1m$, tal como lo muestra la Figura 4. Asimismo, se propone que el desplazamiento de la estructura formada por ambos módulos se desplace a lo largo de eje X , con la finalidad de analizar su locomoción.

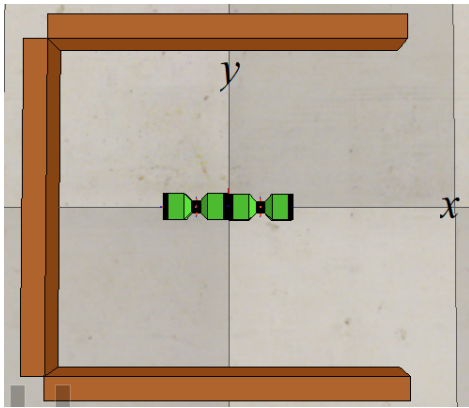


Figura 4: Construcción virtual de los robots modulares MARBot en escena V-REP para simulación de locomoción de la estructura ensamblada.

Debido a que ahora se debe controlar un solo robot formado por dos módulos, las medidas de posición y orientación en el plano dependen de un módulo maestro, para este caso ha sido elegido el módulo 1. De esta forma, las condiciones iniciales como posición, orientación, velocidad, de cada uno de los módulos son propuestas en cero. La implementación del CPG

se lleva a cabo en MATLAB, donde los estados del sistema son definidos como

$$\begin{aligned}\dot{x} &= T_r^{-1}[-\sum_{j=1}^n w_{ij}y_j + s_i - x_i - bf_i + feed], \\ \dot{f} &= T_a^{-1}[y_i - f_i].\end{aligned}$$

Observe que la morfología 1D creada cuenta con cuatro gdl's, y de acuerdo con el modelo de los hemicentros, por cada grado de libertad es necesario considerar dos neuronas. Por lo tanto, los elementos del modelo del CPG ($y_i, s_i, x_i, f_i, b, feed$) tienen una dimensión de (8×1) , y la matriz de conectividad w_{ij} es de (8×8) . Este modelo representa la dinámica del nuevo robot, por lo que la solución de este sistema requiere de la implementación de una integración numérica. Para este trabajo, se utilizó el método de Runge-Kutta de 4^{to} orden. Los parámetros correspondientes al modelo del CPG que han sido utilizados son presentados en la Tabla II y están basados en el trabajo de [5] (véase esta referencia para mayores detalles).

Tabla II: Parámetros del modelo de CPG.

| Parámetros de CPG | |
|-------------------|-------|
| Parámetro | Rango |
| x_i | -8,8 |
| f_i | 0,3 |
| T_r | 0.5 |
| T_a | 0.06 |
| s_i | 8.5 |
| b | 4 |

En este trabajo se propone optimizar la matriz de conectividad w debido a la distribución de los grados de libertad con la que cuentan los módulos. En varios de los trabajos citados, que fueron encontrados en la literatura, se propone que las conexiones de la neuronas sean definidas de acuerdo a la morfología de un robot determinado. Sin embargo, dada la características de autoreconfiguración de MARBot, resulta conveniente la optimización de sus conexiones siempre y cuando se respete el modelo de los hemicentros (esto es, dos neuronas por gdl del robot). La representación de la matriz de conectividad w está dada por

$$w = \begin{bmatrix} 0 & 1 & p & p & p & p & p & p \\ 1 & 0 & p & p & p & p & p & p \\ p & p & 0 & 1 & p & p & p & p \\ p & p & 1 & 0 & p & p & p & p \\ p & p & p & p & 0 & 1 & p & p \\ p & p & p & p & 1 & 0 & p & p \\ p & p & p & p & p & p & 0 & 1 \\ p & p & p & p & p & p & 1 & 0 \end{bmatrix},$$

donde p son los elementos de conectividad. Como función de retroalimentación para el CPG se propone una red neuronal tipo perceptrón multicapa, donde las entradas son los estados del robot y las salidas es la información de entrada al CPG. De esta forma, el CPG propone un movimiento a seguir, y

bajo esta referencia se implemente un controlador tradicional PID.

Para la optimización de los parámetros del robot, se propone minimizar una función objetivo, que dependa de la diferencia de la distancia entre el robot y un punto deseado del plano, y en la orientación del robot; además de maximizar la velocidad del robot. Es importante mencionar que se pondera el error en posición y la velocidad del robot en el eje y , dado que se busca un desplazamiento a lo largo de dicho eje. El movimiento generado deberá emular un desplazamiento similar a una serpiente. La función objetivo propuesta está dada por

$$\begin{aligned} \min[f(P, z, p_y)] = & |P_d - P_1| + |z| + \sum_{i=1}^{size} p_{y_i} \\ & + \max[f(v_x)] \sum_{i=1}^{size} v_{x_i} \end{aligned}$$

donde P_1 es la posición de inicio del robot, P_d es la posición deseada, z es la orientación, p_y es la posición en el eje y , v_x la velocidad en el eje x , y $size$ el número de iteraciones por simulación. La Tabla III muestra las especificaciones de GA para este problema.

Tabla III: Parámetros de GA utilizados para optimizar el problema de locomoción de la estructura formada por dos módulos MARBot ensamblados.

| Parámetros de GA | |
|---------------------------|------------------|
| Tamaño de población | 120 |
| Tamaño de cromosoma | 150 |
| Tipo de selección | Rango |
| Tasa de élite | 5 % |
| Tipo de reproducción | Cruce aritmético |
| Tasa de reproducción | 80 % |
| Tasa mutación decremental | 40 % a 10 % |
| Generaciones | 130 |

III. RESULTADOS

Para el problema de auto-ensamblaje y locomoción, se obtuvieron resultados en simulación. El ambiente de simulación fue implementado en la plataforma V-REP, en donde fueron evaluados las soluciones propuestas por el algoritmo genético. La frecuencia de cada iteración de la simulación de la estructura ensamblada con el controlador es de 50HZ.

III-A. Auto-ensamblaje de dos módulos MARBot

En esta simulación se consideraron dos módulos MARBot con la tarea de acoplarse en el centro del escenario, esto es, $P_d = (0,0)$. Después de 74 generaciones, el algoritmo genético ha descubierto los parámetros adecuados para lograr dicha tarea. La Figura 5 muestra la gráfica del desplazamiento del robot en el plano XY hacia el punto de auto-ensamblaje. La línea uniforme representa la trayectoria del primer módulo, mientras que la línea punteada corresponde a la del segundo módulo. Existe un ligero error en el centro que corresponde a la tolerancia dada por el sistema de acople de ambos robots.

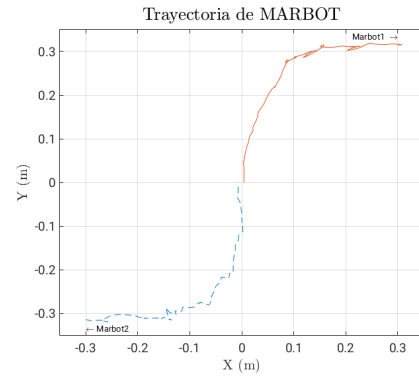


Figura 5: Desplazamiento de los módulos de MARBOT.

El desempeño ideal de la función aptitud propuesta corresponde a un índice de 300 unidades. En este caso, el desempeño máximo alcanzado por nuestro proceso fue de 290.29 unidades, por lo que se puede concluir que el resultado obtenido es muy cercano al óptimo. La Tabla IV muestra el máximo valor ideal de cada elemento de la función de aptitud, así como el valor obtenido con la solución óptima encontrada por la evolución genética.

Tabla IV: Desempeño alcanzado en la tarea de auto-ensamblaje. El valor máximo fue calculado al suponer una trayectoria y parámetros ideales en la función de ajuste.

| Elemento | Recompensas | |
|-----------------|-------------|----------|
| | Máxima | Obtenida |
| $\ x_1 - x_2\ $ | 100 | 98.0171 |
| $\ z_1 - z_2\ $ | 100 | 96.1621 |
| $\ P_d - x_1\ $ | 50 | 48.3302 |
| $\ P_d - x_2\ $ | 50 | 47.7900 |
| Total | 300 | 290.2994 |

III-B. Locomoción de dos módulos MARBot acoplados

Los resultados obtenidos de la optimización de los parámetros del neurocontrolador CPG se muestran en las Figuras 6 y 7 para el desplazamiento virtual y medido sobre eje y , de los módulos MARBot previamente acoplados.

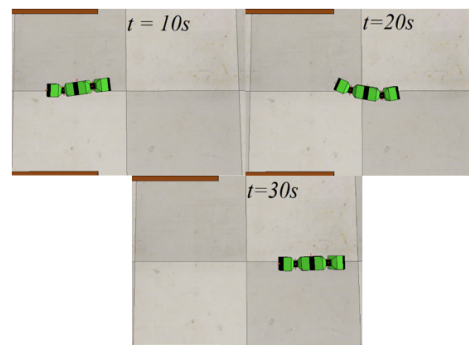


Figura 6: Secuencia de desplazamiento de MARBot

Nuevamente se obtuvo un resultado cercano al óptimo. La aptitud obtenida fue de 650.3240 unidades, mientras que el

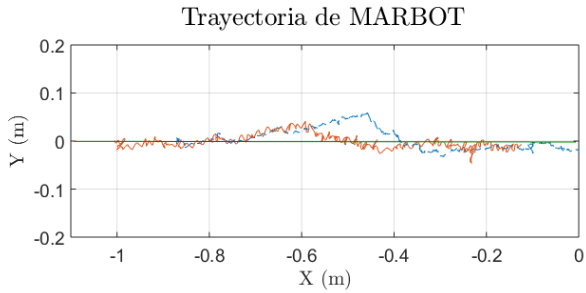


Figura 7: Gráfica de desplazamiento de dos módulos acoplados MARBot. La línea roja representa la trayectoria realizada por el centro de masa del primer módulo, mientras que la línea azul es correspondiente al segundo módulo. Nótese que ambos trazos se aproximan al movimiento ideal que es sobre el eje X.

desempeño ideal es de 700 unidades. La comparación entre los índices ideales y los obtenidos se muestran en la Tabla V. Es importante mencionar que la matriz de conectividad obtenida, esquema en Figura 8, es incompleta a diferencia de las configuraciones comúnmente utilizadas en la literatura. Lo anterior implica que las conexiones que no son útiles en la red neuronal se han desechado.

Tabla V: Desempeño en la tarea de locomoción de la estructura ensamblada. El valor máximo fue calculado al suponer una trayectoria y parámetros ideales en la función de ajuste.

| Elemento | Desempeño | |
|---------------|-----------|----------|
| | Máxima | Obtenida |
| $ P_d - P_1 $ | 300 | 300 |
| $ z $ | 200 | 176.8733 |
| $posx$ | 100 | 84.9031 |
| $velx$ | 100 | 88.5476 |
| Total | 700 | 650.3240 |

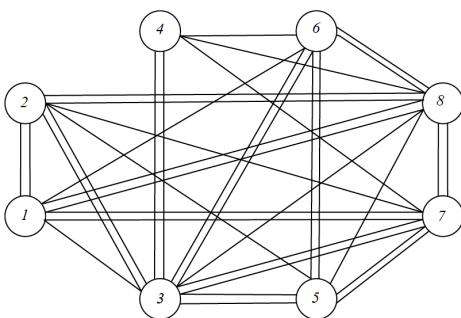


Figura 8: Grafo de conexiones del CPG.

IV. CONCLUSIÓN

En este trabajo se presentaron dos métodos para el control del robot modular MARBot. El primero de ellos enfocado al auto-ensamblaje de dos módulos por medio del empleo de

un neurocontrolador perceptrón multicapa homogéneo, optimizado para cada módulo. El segundo método consistió en utilizar un neurocontrolador CPG para resolver el problema de locomoción de la estructura generada por el acoplamiento de ambos módulos MARBot. Los parámetros de ambos neurocontroladores fueron optimizados fuera de línea mediante un GA. Los resultados obtenidos han sido implementados en el ambiente virtual de simulación V-REP y confirman la solución de los dos problemas planteados. El rendimiento de ambos neurocontroladores optimizados se encuentran muy cercanos al óptimo definido de acuerdo a cada objetivo deseado. Se propone la implementación de los neurocontroladores propuestos en los robots modulares MARBot reales como trabajo futuro.

AGRADECIMIENTOS

El desarrollo de este proyecto fue apoyado por el TecNM-ITE a través del proyecto "Navegación de robots móviles como un sistema adaptativo complejo", con clave 6474.18-P.

REFERENCIAS

- [1] Y. U. Cao, A. S. Fukunaga, A. B. Kahng, and F. Meng, "Cooperative mobile robotics: Antecedents and directions," in *Intelligent Robots and Systems 95: Human Robot Interaction and Cooperative Robots*, Proceedings. 1995 IEEE/RSJ International Conference on, vol. 1. IEEE, 1995, pp. 226–234.
- [2] L. Bayındır, "A review of swarm robotics tasks," *Neurocomputing*, vol. 172, pp. 292 – 321, 2016.
- [3] T. Arai, E. Pagello, and L. E. Parker, "Editorial: Advances in multi-robot systems," *IEEE Trans on Robotics and Automation*, vol. 18, no. 5, pp. 655 – 661, 2002.
- [4] S. Murata, E. Yoshida, A. Kamimura, H. Kurokawa, K. Tomita, and S. Kokaji, "M-tran: Self-reconfigurable modular robotic system," *IEEE/ASME transactions on mechatronics*, vol. 7, no. 4, pp. 431–441, 2002.
- [5] A. Kamimura, H. Kurokawa, E. Yoshida, S. Murata, K. Tomita, and S. Kokaji, "Automatic locomotion design and experiments for a modular robotic system," *IEEE/ASME Transactions on mechatronics*, vol. 10, no. 3, pp. 314–325, 2005.
- [6] G. G. Ryland and H. H. Cheng, "Design of imobot, an intelligent reconfigurable mobile robot with novel locomotion," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 60–65.
- [7] A. Kamimura, S. Murata, E. Yoshida, H. Kurokawa, K. Tomita, and S. Kokaji, "Self-reconfigurable modular robot-experiments on reconfiguration and locomotion," in *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, vol. 1. IEEE, 2001, pp. 606–612.
- [8] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: a review," *Neural networks*, vol. 21, no. 4, pp. 642–653, 2008.
- [9] S. Ok, K. Miyashita, and K. Hase, "Evolving bipedal locomotion with genetic programming—a preliminary report," in *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, vol. 2. IEEE, 2001, pp. 1025–1032.
- [10] J. Fan, Y. Zhang, H. Jin, X. Wang, D. Bie, J. Zhao, and Y. Zhu, "Chaotic cpg based locomotion control for modular self-reconfigurable robot," *Journal of Bionic Engineering*, vol. 13, no. 1, pp. 30–38, 2016.
- [11] Y. Li, Y. Liu, and X. Liu, "Active vibration control of a modular robot combining a back-propagation neural network with a genetic algorithm," *Journal of Vibration and Control*, vol. 11, no. 1, pp. 3–17, 2005.
- [12] X. B. Peng, G. Berseth, K. Yin, and M. Van De Panne, "Deeploco: Dynamic locomotion skills using hierarchical deep reinforcement learning," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 41, 2017.
- [13] D. J. Montana and L. Davis, "Training feedforward neural networks using genetic algorithms," in *IJCAI*, vol. 89, 1989, pp. 762–767.
- [14] J. Cristiano, D. Puig, and M. A. García, "Locomotion control of a biped robot through a feedback cpg network," in *ROBOT2013: First Iberian Robotics Conference*. Springer, 2014, pp. 527–540.

Modelado, Control y Sintonía de un Péndulo Invertido sobre Ruedas

Morales Viscaya,
Joel Artemio
División de Estudios de
Posgrado e Investigación
ITLP
La Paz, B.C.S, México
jmoralesv@itlp.edu.mx

Gómez Torres,
José Luis
División de Estudios de
Posgrado e Investigación
ITLP
La Paz, B.C.S, México
jgomezt@itlp.edu.mx

Maravilla Castro,
Merlín Octavio
Ingeniería
Electromecánica
ITLP
La Paz, B.C.S, México
merlinomc96@gmail.com

Benoit Mendivil,
Carlos de Jesús
Ingeniería en
Sistemas Computacionales
ITLP
La Paz, B.C.S, México
cdjbm03@gmail.com

Resumen—Los sistemas subactuados, particularmente los no lineales, son más difíciles de controlar, sin embargo, son en general más eficientes en cuestiones de energía y su construcción requiere menos actuadores que un sistema completamente actuado. En el presente artículo se presenta el modelado, control, simulación y sintonía del controlador propuesto para un sistema tipo péndulo invertido sobre ruedas (WIP, por *Wheeled Inverted Pendulum*). Las aplicaciones de este tipo de dispositivos van desde el transporte de cargas medianas, hasta la seguridad y el uso recreativo.

Index Terms—Optimización, WIP, Péndulo Invertido sobre Ruedas, Sintonía, Regulador Cuadrático Lineal, Segway, Modelado, Control.

I. INTRODUCCIÓN

El péndulo invertido sobre ruedas (WIP) es un sistema subactuado no lineal y sin embargo, relativamente simple, que nos permite estudiar el comportamiento y los métodos de control para dichos sistemas. Este tipo de sistemas se utilizan como dispositivos de transporte en entornos estructurados, como puede verse en [1], [2], [3], y poseen una serie de ventajas como permitir al usuario el desenvolvimiento en espacios reducidos y en una postura de pie, resultando más adecuado que otros dispositivos como las sillas de ruedas. Dada la gama de aplicaciones cabe considerar la variedad de criterios de desempeño para su funcionamiento, lo que hace deseable estudiar metodologías que permitan modificar de forma sencilla el sistema de control en función de la aplicación. Algunas propuestas de Control para el sistema se presentan en [1], [2], [3], [4], [5], [6]. El WIP consiste en un péndulo colocado sobre una plataforma con dos ruedas coaxiales actuadas; un diagrama general se muestra en la Fig. 1, siendo el Segway [7] una versión comercial bastante conocida.

El objetivo principal de control, es mantener al péndulo (que en una posible aplicación incluiría al usuario) en la posición vertical y las ruedas en reposo. Además, como sólo posee dos actuadores, uno en cada rueda y no es posible controlar directamente la aceleración angular del péndulo, el sistema se considera subactuado (con tres grados de libertad y sólo dos actuadores).

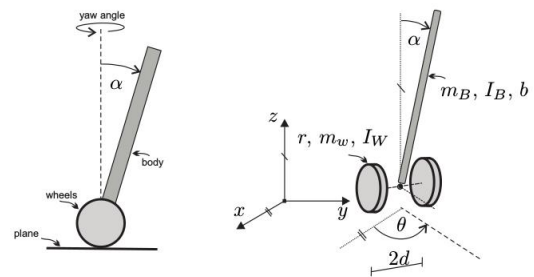


Fig. 1: Diagrama del péndulo invertido sobre ruedas. Fuente: [8].

La estructura del presente artículo es la siguiente: En esta sección se presentan algunos conceptos introductorios; en la segunda sección se presenta el modelo del WIP considerando restricciones no-holónicas; en la sección III se presenta el Control LQR aplicado al sistema linealizado, en la sección IV se presenta el procedimiento para sintonizar el controlador propuesto; en las secciones V y VI se muestran resultados en simulación y finalmente se presentan conclusiones acerca del desempeño y el trabajo futuro.

Sistemas subactuados no lineales

La condición de subactuación en mecánica, robótica, mecatrónica y sistemas dinámicos se refiere a un sistema con más grados de libertad (número de variables independientes que definen la configuración del sistema) que actuadores (variables de entrada) a ser controlado [9]. Los sistemas dinámicos pueden ser modelados como un número finito de ecuaciones diferenciales de primer orden acopladas, de la forma:

$$\begin{aligned} \dot{x}_1 &= F_1(t, x_1, \dots, x_n, u_1, \dots, u_p) \\ \dot{x}_2 &= F_2(t, x_1, \dots, x_n, u_1, \dots, u_p) \\ &\vdots \\ \dot{x}_n &= F_n(t, x_1, \dots, x_n, u_1, \dots, u_p) \end{aligned} \quad (1)$$

donde \dot{x}_i denota la derivada de x_i con respecto a la variable temporal t y u_1, u_2, \dots, u_p variables de entrada. Las

variables x_1, x_2, \dots, x_n se denominan variables de estado y representan la memoria que el sistema dinámico tiene de su comportamiento pasado [10].

Una representación alternativa que desacopla las variables de entrada es: $\dot{x} = f(t, x) + g(u)$, una ecuación diferencial de vectores n -dimensionales, denominada ecuación de estados.

Un caso especial de la ecuación anterior ocurre cuando la función f no depende explícitamente del tiempo, esto es $\dot{x} = f(x) + g(u)$, en cuyo caso se dice que el sistema es autónomo invariante en el tiempo.

Cuando $g(u)$, que representa la presencia de fuerzas externas que alteran la dinámica del sistema, no es de rango completo, se dice que el sistema posea algún grado de subactuación.

LQR

En términos generales un problema de control óptimo consiste en generar una ley de control u para un sistema de la forma (2)

$$\dot{x} = F(x, u, t) \quad (2)$$

de tal forma que x siga una trayectoria que optimice algún criterio de desempeño o función de costo J [11]. Usualmente, se busca minimizar o maximizar el criterio de desempeño seleccionando la entrada de control. Para cada $u(t)$ posible se asociará una trayectoria del sistema $x(t)$.

Los criterios más usuales son de posición (llevar al sistema tan cerca como sea posible a un estado terminal en un periodo de tiempo dado), de tiempo (llevar a un estado terminal en el periodo de tiempo más corto posible) o de energía (llevar de un estado a otro con la menor inversión de energía) [12]. La selección del índice de desempeño depende de la naturaleza específica del problema de control, sin embargo uno de los criterios de desempeño ampliamente utilizado que combina los mencionados anteriormente es el criterio de desempeño cuadrático, que en forma general adopta la estructura (3).

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (3)$$

donde $Q > 0$ y $R > 0$ son matrices simétricas de peso para el estado y el control, respectivamente.

El esquema de control conocido como Regulador Cuadrático Lineal (LQR) brinda una ley de control óptima para un sistema lineal con un índice de desempeño cuadrático [13].

$$\dot{x} = Ax + Bu \quad (4)$$

Considere el sistema en espacio de estados (4) con $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$ con el índice de desempeño (3). El control óptimo que minimiza J está dado por la ley de control lineal de realimentación de estados (5), con $K = R^{-1} B^T P$

$$u(t) = -Kx(t) \quad (5)$$

donde P es la única solución definida positiva de la ecuación algebraica de Riccati (6).

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (6)$$

Las matrices $Q \in \mathbb{R}^{n \times n}$ y $R \in \mathbb{R}^{p \times p}$ son los parámetros de sintonía del problema. Una elección común para R suele ser λI con $\lambda > 0$ e I la matriz identidad de $p \times p$, de acuerdo a algunos autores [12], valores “pequeños” de λ se asocian a una convergencia más rápida de $y(t) \rightarrow 0$ pero exigen valores mayores para la señal de control $u(t)$.

Sintonía de Controladores

Un problema común en el diseño de leyes de control para sistemas dinámicos es el de sintonizar dichos controladores, es decir, encontrar valores adecuados para las constantes o parámetros de diseño del mismo.

La mayoría de las estrategias de control no proporcionan una metodología para ajustar dichos parámetros de diseño y el desempeño de los controladores depende en gran medida de los mismos.

Una alternativa válida para llevar a cabo la sintonía de controladores, es considerarlo un problema de optimización [14]; para hacerlo es necesario seleccionar un índice de desempeño y un algoritmo de optimización.

Para conseguir una sintonía más fina y mejorar el desempeño del controlador, es necesario tomar en cuenta el régimen transitorio en el índice de desempeño, sin enfocarse solamente en el tiempo que el sistema tarda en alcanzar la regulación.

La mayoría de los índices de desempeño que toman en cuenta el régimen transitorio consideran el error de seguimiento $e(t)$ como la distancia del punto $x_r(t)$ respecto al valor deseado $x_d(t)$.

Un índice de desempeño utilizado comúnmente en ingeniería es la integral del error cuadrático o ISE por sus siglas en inglés (Integrated Square Error)(7)

$$ISE = \int_0^{\infty} e(t)^2 dt \quad (7)$$

En este índice de desempeño, errores grandes contribuyen a la integral de forma considerable, lo cual tiende a producir soluciones con sobreimpulsos pequeños que llegan al punto de regulación con celeridad, pero tiene la desventaja de ser insensible a errores pequeños.

II. MODELO DINÁMICO DEL WIP

Sea el espacio de configuración $Q_f \in \mathbb{R}^6$ y las coordenadas locales $q = (x, y, \alpha, \theta, \varphi_r, \varphi_l) \in Q_f$. La posición del centro del eslabón que une las ruedas izquierda y derecha del WIP en el plano horizontal es $r = (x, y)$. Los ángulos de cabeceo y guiñada son respectivamente α y θ , mientras que φ_r y φ_l representan la rotación absoluta de las ruedas derecha e izquierda respectivamente.

Considerando ambas ruedas idénticas de masa m_r y el péndulo como cuerpos rígidos homogéneos, el centro de masa de la rueda izquierda, rueda derecha y del cuerpo son respectivamente (8), (9) y (10).

$$r_l = (x(t) - d \cdot s(\theta(t)), \quad y(t) + d \cdot c(\theta(t)), \quad 0) \quad (8)$$

$$r_r = (x(t) + d \cdot s(\theta(t)), \quad y(t) - d \cdot c(\theta(t)), \quad 0) \quad (9)$$

$$\begin{aligned} r_B &= (x(t) + b \cdot s(\alpha(t))c(\theta(t)), \\ & \quad y(t) + b \cdot s(\alpha(t))s(\theta(t)), \\ & \quad b \cdot c(\alpha(t))) \end{aligned} \quad (10)$$

con $s() = \text{sen}()$, $c() = \text{cos}()$, $2d$ es la distancia entre las ruedas y b la distancia del punto central del eje de las ruedas (PCR) al centro de masa del cuerpo del péndulo.

Es posible calcular la energía potencial total como la suma de (11), (12) y (13).

$$Ep_l = m_r \cdot g \cdot r_{l3} \quad (11)$$

$$Ep_r = m_r \cdot g \cdot r_{r3} \quad (12)$$

$$Ep_B = m_B \cdot g \cdot r_{B3} \quad (13)$$

$$Ep_T = Ep_l + Ep_r + Ep_B \quad (14)$$

Donde Ep_l , Ep_r y Ep_B son las energías potenciales de la rueda izquierda, la rueda derecha y el cuerpo del péndulo, respectivamente; m_B y g son la masa del cuerpo del péndulo y la constante de aceleración gravitacional, y r_{l3} , r_{r3} y r_{B3} son el tercer elemento de las coordenadas respectivas.

Para calcular la energía cinética de rotación se define la matriz simétrica que caracteriza la inercia rotacional de cada cuerpo rígido como (15), (16) y (17).

$$J_l = \begin{bmatrix} J_{rx} & 0 & 0 \\ 0 & J_{ry} & 0 \\ 0 & 0 & J_{rz} \end{bmatrix} \quad (15)$$

$$J_r = \begin{bmatrix} J_{rx} & 0 & 0 \\ 0 & J_{ry} & 0 \\ 0 & 0 & J_{rz} \end{bmatrix} \quad (16)$$

$$J_B = \begin{bmatrix} J_{Bx} & 0 & 0 \\ 0 & J_{By} & 0 \\ 0 & 0 & J_{Bz} \end{bmatrix} \quad (17)$$

Con $J_{rx} = J_{ry}$ y $J_{Bx} = J_{By}$, por simetría; considerando además las matrices de rotación que permiten pasar los marcos de referencia ubicados en los centros de masa de los cuerpos rígidos, al marco de referencia ubicado en el PCR, pueden calcularse las velocidades angulares ω_l , ω_r y ω_B de los cuerpos y con ellas su energía cinética a partir de (18), (19) y (20).

$$Ec_l = (1/2)m_r \cdot \left\| \frac{d(r_l)}{dt} \right\|^2 + (1/2)\omega_l^T J_l \cdot \omega_l \quad (18)$$

$$Ec_r = (1/2)m_r \cdot \left\| \frac{d(r_r)}{dt} \right\|^2 + (1/2)\omega_r^T J_r \cdot \omega_r \quad (19)$$

$$Ec_B = (1/2)m_B \cdot \left\| \frac{d(r_B)}{dt} \right\|^2 + (1/2)\omega_B^T J_B \cdot \omega_B \quad (20)$$

$$Ec_T = Ec_l + Ec_r + Ec_B \quad (21)$$

Para caracterizar la dinámica del WIP se utilizó el lagrangiano del sistema \mathcal{L} , un mapeo $\mathcal{L} : Q_f \rightarrow \mathbb{R}$ que se define

como la energía cinética total menos la energía potencial total, $\mathcal{L} = Ec_T - Ep_T$, el cual se calcula utilizando (14) y (21).

El sistema físico además tiene la restricción de que puede moverse instantáneamente hacia adelante o atrás pero no lateralmente por el deslizamiento de las ruedas. Es decir, utiliza una locomoción diferencial. Dicha restricción al no poder escribirse directamente en términos de q , se clasifica como no-holonómica [15].

Esta restricción de rodamiento sin deslizamiento, sin embargo, es lineal respecto a \dot{q} por lo que de acuerdo a Delgado [8] se puede representar como (22)

$$A^T \dot{q} = 0 \quad (22)$$

donde:

$$A^T = \begin{bmatrix} -\text{sen}(\theta) & \text{cos}(\theta) & 0 & 0 & 0 & 0 \\ \text{cos}(\theta) & \text{sen}(\theta) & d & 0 & -r & 0 \\ \text{cos}(\theta) & \text{sen}(\theta) & -d & 0 & 0 & -r \end{bmatrix} \quad (23)$$

La dinámica de sistemas mecánicos con restricciones representables de la forma (22) puede ser descrita utilizando las ecuaciones de Lagrange-D'Alembert (24)

$$\frac{d}{dt}(\nabla_{\dot{q}}L) - \nabla_q L = A(q)\lambda + F_{ext} \quad (24)$$

En ausencia de fuerzas externas adicionales a la entrada del sistema τ , (24) puede escribirse en forma matricial como:

$$\tilde{M}\ddot{q} + \tilde{C}(q, \dot{q})\dot{q} + \nabla_q V(q) = A(q)\lambda + \tau \quad (25)$$

donde $\tilde{M}(q) = \tilde{M}(q)^T$ y $\tilde{M}(q) > 0$ es la matriz de inercia, el término $\tilde{C}(q, \dot{q})\dot{q}$ representa las fuerzas de coriolis y centrífugas.

La restricción se une al sistema utilizando multiplicadores de Lagrange $\lambda \in \mathbb{R}^3$ que representan la magnitud de las fuerzas restrictivas que obligan al sistema a satisfacer la mencionada restricción.

Debido a que A es una matriz de rango completo, es decir, posee rango tres, las velocidades admisibles ν de q , deben pertenecer a un subconjunto del espacio tangente de Q_f , denominado espacio tangente restringido D_q , de solamente tres dimensiones (las seis coordenadas originales menos el rango de la matriz de restricciones A), considerando q en el espacio de configuración original Q_f , ν debe satisfacer (26).

$$\dot{q} = S(q)\nu \quad (26)$$

Siendo S una matriz de rango completo tal que $A^T S = 0$ para toda $q \in Q_f$ y los valores que se muestran a continuación:

$$S(q) = \begin{bmatrix} \text{cos}\theta & 0 & 0 \\ \text{sin}\theta & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ \frac{1}{r} & 0 & \frac{d}{r} \\ \frac{1}{r} & 0 & -\frac{d}{r} \end{bmatrix} \quad (27)$$

Las velocidades admisibles de q caen en el espacio generado por las columnas de S , esto es, el espacio 3-dimensional D_q [8]. Reemplazando $\dot{q} = S\nu$ y $\ddot{q} = S\dot{\nu} + \dot{S}\nu$ en (25) y eliminando las restricciones premultiplicando por S^T se produce el sistema (28).

$$S^T \tilde{M} S \dot{\nu} + S^T (\tilde{M} \dot{S} + \tilde{C} S) \nu + S^T \nabla_q E p_T = S^T \tau \quad (28)$$

El sistema dinámico (28) puede escribirse en la forma compacta (29)

$$M_g \dot{\nu} + C_g \nu + S^T \nabla_q E p_T = S^T \tau \quad (29)$$

Dado que (26) produce $\nu = (v, \dot{\alpha}, \dot{\theta})$, con v la velocidad de avance, un conjunto de coordenadas más apropiado para el sistema restringido, se puede describir a partir de definir $\psi = (s, \alpha, \theta)$, donde s representa la longitud de arco, tal que $v = \dot{\psi}$, las ecuaciones de movimiento en el espacio restringido en su formulación Newton-Euler están dadas por:

$$M \ddot{\psi} + C \dot{\psi} + \nabla_{\psi} E p_T = Gu \quad (30)$$

con

$$M = \begin{bmatrix} c_1 & c_2 \cdot c(\alpha(t)) & 0 \\ c_2 \cdot c(\alpha(t)) & c_3 & 0 \\ 0 & 0 & c_4 - c_2 \cdot s^2(\alpha(t)) \end{bmatrix} \quad (31)$$

$$C = \begin{bmatrix} 0 & c_5 \cdot s(\alpha(t)) \dot{\alpha}(t) & c_5 \cdot s(\alpha(t)) \dot{\theta}(t) \\ c_5 \cdot s(\alpha(t)) \dot{\alpha}(t) & 0 & 0 \\ c_5 \cdot s(\alpha(t)) \dot{\theta}(t) & 0 & c_6 \cdot c(\alpha(t)) \cdot s(\alpha(t)) \dot{\alpha}(t) \end{bmatrix} \quad (32)$$

$$\nabla_{\psi} E p_T = \begin{bmatrix} 0 \\ -c_7 s(\alpha(t)) \\ 0 \end{bmatrix} \quad (33)$$

$$G = \begin{bmatrix} 1/r & 1/r \\ -1 & -1 \\ -d/r & d/r \end{bmatrix} \quad (34)$$

donde:

$$c_1 = m_B + 2(m_r + J_r y / r^2)$$

$$c_2 = b \cdot m_B$$

$$c_3 = J_B y + b^2 \cdot m_B$$

$$c_4 = J_B z + 2J_r z + 2 \cdot d^2 \cdot m_r + \frac{2 \cdot d^2 \cdot J_r y}{r^2} - (1/2) b^2 \cdot m_B$$

$$c_5 = - (1/2) b \cdot m_B$$

$$c_6 = b^2 \cdot m_B y$$

$$c_7 = c_2 \cdot g$$

III. CONTROL LQR APLICADO AL WIP

El sistema (30), dado que M (31) es una matriz invertible, puede escribirse de la forma (35)

$$\ddot{\psi} = M^{-1} (Gu - C\dot{\psi} - \nabla_{\psi} E p_T) \quad (35)$$

La representación en espacio de estados de (35) con $x(t) = (\psi, \dot{\psi}) = (s, \alpha, \theta, v, \dot{\alpha}, \dot{\theta})$ resulta natural para utilizar un controlador LQR, sin embargo, debido a la independencia del sistema respecto a la variable s y su no controlabilidad, resulta más adecuado definir el vector de estados reducido $x_r(t) = (\alpha, \theta, v, \dot{\alpha}, \dot{\theta})$.

El objetivo de control es llevar el vector de estados $x_r(t)$ al origen, más formalmente, puede expresarse como (36).

$$\lim_{t \rightarrow \infty} [x_r(t)] = [0] \quad (36)$$

Debido a la no linealidad de (35) y con el objetivo de obtener un modelo matemático lineal de la forma (4), se decidió utilizar la técnica de linealización, basada en una expansión por serie de Taylor alrededor del punto de operación $x_r(t) = 0$.

El modelo linealizado de (35) queda descrito por (37)

$$\dot{x}_r = A_0 x_r + B_0 u \quad (37)$$

con

$$A_0 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ a_1/a_2 & 0 & 0 & 0 & 0 \\ a_3/a_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (38)$$

$$B_0 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ b_1/b_2 & b_1/b_2 \\ -b_3/b_2 & -b_3/b_2 \\ b_4 & -b_4 \end{bmatrix} \quad (39)$$

y

$$a_1 = b^2 \cdot g \cdot m_B^2 \cdot r^2$$

$$a_2 = 2b^2 \cdot m_B \cdot (J_r y + m_r \cdot r^2) + j_B y \cdot (2J_r y + r^2(m_B + m_r))$$

$$a_3 = b \cdot g \cdot m_B \cdot (2J_r y + r^2(m_B + 2m_r))$$

$$b_1 = r(J_B y + b \cdot m_B(b + r))$$

$$b_2 = 2 \cdot (b^2) \cdot m_B \cdot (J_r y + m_r \cdot r^2) + J_B y (2J_r y + (m_B + 2m_r)r^2)$$

$$b_3 = 2J_r y + r(b \cdot m_B + (m_B + 2m_r)r)$$

$$b_4 = d \cdot r / ((J_B z + 2J_r y)r^2 + (2(d^2)J_r y + m_r \cdot r^2))$$

De acuerdo a la teoría de Kalman [13], el sistema es controlable, es decir, es posible transferir el sistema desde cualquier estado inicial $x_r(t_0)$ a cualquier otro en un tiempo finito, ya que la matriz de controlabilidad del sistema, $C = [B|AB|A^2B]$ es de rango completo.

Conociendo los valores de las matrices A_0 y B_0 que caracterizan el sistema cerca del punto de operación, y considerando trivial la resolución de la ecuación de Riccati para un conjunto de matrices Q y R , el paso restante es el diseño de dichas matrices.

IV. SINTONÍA DEL CONTROLADOR LQR

Para llevar a cabo el ajuste del controlador LQR, es decir, diseñar las matrices Q y R , se decidió plantear este problema como uno de optimización, debido a que el objetivo de control propuesto es llevar el vector de estados al origen, se define $e(t)$ como $\|x_r(t)\|_2$.

Como criterio de desempeño o función de aptitud, se decidió utilizar una variante de ISE que se conoce como la integral del error cuadrado ponderada en el tiempo (40) o ITSE (Integrated time-square error), que tiene la ventaja de darle mayor importancia a los errores que permanecen en el tiempo y con ello penalizar soluciones oscilatorias.

$$ITSE = \int_0^{\infty} te(t)^2 dt \quad (40)$$

Para la parte de optimización, es decir el problema de minimizar (40), se consideraron dos opciones; la primera, un método numérico, el algoritmo de Fletcher-Reeves, también conocido como gradiente conjugado, que es una variante del método clásico de descenso de gradiente. Lo anterior bajo la hipótesis de que la función de aptitud utilizada es una curva suave y diferenciable. Una condición para que esto ocurra es que el error de seguimiento respecto al tiempo sea a su vez diferenciable, ya que el único requisito para que se alcance regulación cerca del punto de operación utilizando LQR es que Q y R sean simétricas definidas positivas, la hipótesis anterior es bastante plausible.

Se implementó también el algoritmo heurístico de Nelder-Mead, conocido popularmente como el método simplex no lineal, que tiene la principal fortaleza en que no requiere que la función a optimizar sea suave ni conocer las derivadas parciales de la misma. Dicho algoritmo se desarrolló como una alternativa en caso de que la función de error no fuese lo suficientemente suave.

El algoritmo de Fletcher-Reeves mejora las propiedades de convergencia del método de descenso de gradiente al utilizar direcciones conjugadas en cada paso.

Mientras el método de descenso de gradiente se basa en mover un punto de partida en direcciones proporcionales al negativo del gradiente de la función, ya que la derivada direccional en dicha dirección es negativa a menos que se encuentre en un punto crítico. En el algoritmo de Fletcher-Reeves la dirección de movimiento en cada paso se inicializa en el negativo del gradiente, pero a este se le resta una cantidad proporcional a la dirección del paso anterior (42), produciendo direcciones conjugadas y acelerando así la convergencia [16].

$$x_n = x_{n-1} + \lambda s_n \quad (41)$$

$$s_n = -\nabla F(x_n) + \frac{\|\nabla F(x_n)\|}{\|\nabla F(x_{n-1})\|} s_{n-1} \quad (42)$$

El algoritmo de Nelder-Mead [17], por otra parte, utiliza el concepto de un simplex, a partir de un conjunto de soluciones candidatas, construye un politipo que se va modificando mediante el descarte de las soluciones menos prometedoras y la adición de puntos nuevos creados a partir de combinar los mejores puntos en el politipo.

Debido a que tanto el algoritmo de Nelder-Mead como el de Fletcher-Reeves pueden quedar atrapados en óptimos locales, una alternativa utilizada en este trabajo es reiniciar los algoritmos cada que esto ocurra utilizando distintos puntos de inicio generados aleatoriamente.

V. DETALLES DE IMPLEMENTACIÓN

Para llevar a cabo la simulación del sistema se consideraron los parámetros de la Tabla I.

| Símbolo | Descripción | Unidad |
|----------|--|------------------------|
| m_r | Masa de las ruedas | 3.86 kg |
| d | Distancia del centro del eje a las ruedas | 0.6 m |
| r | Radio de ambas ruedas | 0.166 m |
| m_B | Masa del péndulo | 130 kg |
| J_{By} | Inercia del péndulo respecto al eje y | 7.51 kg m ² |
| J_{Bz} | Inercia de cada rueda respecto al eje z | 7.68 kg m ² |
| J_{ry} | Inercia del péndulo respecto al eje y | 0.25 kg m ² |
| J_{rz} | Inercia de cada rueda respecto al eje z | 0.12 kg m ² |
| b | Distancia del centro del eje al centro de masa del péndulo | 0.15 m |
| g | Constante de gravitación | 9.81 m/s ² |

Tabla I: Parámetros utilizados para la simulación.

Considerando dichos parámetros, las matrices numéricas A_0 y B_0 son respectivamente (43) y (44)

$$A_0 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ -2,98778 & 0 & 0 & 0 & 0 \\ 23,9154 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (43)$$

$$B_0 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0,0662734 & 0,0662734 \\ -0,219677 & -0,219677 \\ 0,210064 & -0,210064 \end{bmatrix} \quad (44)$$

Las matrices Q y R se consideraron múltiplos escalares de la matriz identidad, es decir, $Q = \mu_1 I_5$ y $R = \mu_2 I_2$

El periodo considerado para la simulación fue de cinco segundos, con condiciones iniciales $x(t_0) = (1/2rad, 1/2rad, -1/2m/s, 1/2rad/s, -1/2rad/s)$, adicionalmente se limitó el par requerido para la ley de control en $50Nm$.

Se utilizó GNU Octave, Software Libre que permite la programación de alto nivel, producido originalmente para cómputo numérico. Sobre GNU Octave se implementó una versión de Runge-Kutta-Fehlberg [18] para llevar a cabo las simulaciones, con una tolerancia de 0,01, y los parámetros $\Delta t = 0,01s$, $\Delta t_{max} = 64\Delta t$ y $\Delta t_{min} = \Delta t/64$. Versiones de los algoritmos de Fletcher-Reeves y de Nelder-Mead también se implementaron para llevar a cabo el cálculo de los valores óptimos de μ_1 y μ_2 que minimicen la función de aptitud (40).

VI. RESULTADOS

La ejecución de cada algoritmo de optimización produjo distintos valores para las constantes, pero en todos los casos los algoritmos obtuvieron constantes que producen el mismo valor en la función de aptitud, lo cual indica que (40) posee múltiples óptimos globales, de manera similar a la conocida función sombrero [19]

De entre los puntos óptimos se decidió utilizar $\mu = (\mu_1, \mu_2) = (1,6868(10^2), 1,0182(10^{-2}))$ debido a que es el que posee un número de condición mayor en la matriz R , $R = \mu_2 I_2$ y por lo tanto produciría un menor error numérico al operarse. El valor encontrado por ambos algoritmos para la aptitud (40) en todos los puntos óptimos es 7,8829.

La ley de control que se produce considerando estos valores para Q y R , $u = -Kx_r$, se encuentra caracterizada por la matriz K (45)

$$K = \begin{bmatrix} -389,659 & 91,012 & -91,012 & -131,433 & 93,362 \\ -389,659 & -91,012 & -91,012 & -131,433 & -93,362 \end{bmatrix} \quad (45)$$

La regulación del sistema $\dot{x}_r = (A_0 - B_0K)x_r$ se comprueba ya que los valores propios de la matriz $(A_0 - B_0K)$ se localizan en el semiplano izquierdo del plano complejo, el conjunto de valores propios es $-42,2942, -1,6939 + 1,0617i, -1,6939 - 1,0617i, -38,2237$ y $-1,003$.

El comportamiento en simulación de los estados α , θ y v se puede apreciar en la Fig. 2, los valores requeridos por la ley de control se presentan en la Fig. 3.

VII. CONCLUSIONES Y TRABAJO FUTURO

La ley de control LQR propuesta, cumple con el objetivo de control como se aprecia en la Fig. 2 al regular los estados del sistema tipo WIP, lo hace además sin exceder el valor límite permitido para los pares tal y como se ve en la Fig. 3.

El diseño de una ley de control no lineal que pueda utilizarse para regular el sistema desde puntos lejanos al de operación así como el refinamiento del modelo matemático considerando fricción y tomando en cuenta la dinámica de los actuadores queda como trabajo futuro.

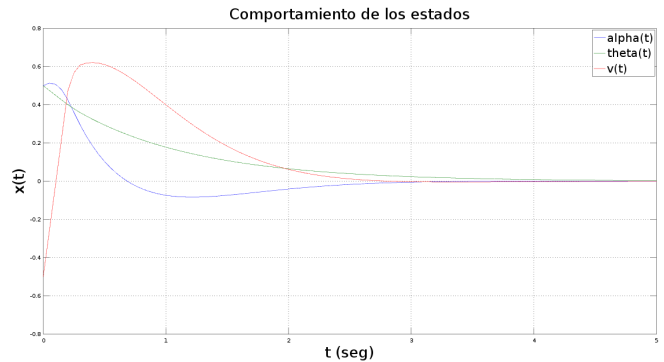


Fig. 2: Comportamiento de los estados del WIP

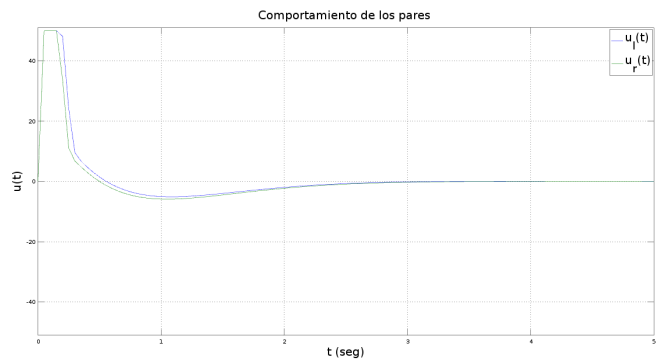


Fig. 3: Comportamiento de los pares requeridos

La metodología propuesta para optimizar los valores de las matrices de diseño del controlador, Q y R considerando el problema de control como uno de optimización parece ser adecuada al menos para este sistema, una alternativa a tener en cuenta sería utilizar una función de aptitud distinta que pondere la importancia de regular cada estado (por ej. podría ser más crítico regular α que θ).

REFERENCIAS

- [1] H. Igarashi, T. Saito, T. Kinjyo, and F. Matsuno, "Development of an autonomous inverted pendulum mobile robot for outdoor environment," *2008 SICE Annual Conference*, pp. 2282–2285, 2008. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4655044>
- [2] J. S. Yoon, "by Design and Production of a Prototype Wheeled Pendulum," 2006.
- [3] A. Castro, "Modelling and dynamic analysis of a two-wheeled inverted-pendulum," no. August, 2012.
- [4] A. Wasif, D. Raza, W. Rasheed, Z. Farooq, and S. Q. Ali, "Design and implementation of a two wheel self balancing robot with a two level adaptive control," *8th International Conference on Digital Information Management, ICDIM 2013*, pp. 187–193, 2013.
- [5] a. M. Almeshal, K. M. Goher, a. N. K. Nasir, M. O. Tokhi, and S. a. Agouri, "Fuzzy logic optimized control of a novel structure two-wheeled robotic vehicle using HSDBC, SDA and BFA: A comparative study," *2013 18th International Conference on Methods & Models in Automation & Robotics (MMAR)*, pp. 656–661, 2013. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6669988>
- [6] M.-L. Chen, "Analysis and Design of Robust Feedback Control Systems for a Nonlinear Two-Wheel Inverted Pendulum System," *2012 International Symposium on Computer,*

- Consumer and Control*, pp. 949–953, 2012. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6228229>
- [7] “Personal Transportation That Simply Moves You — Segway.” [Online]. Available: <http://www.segway.com/>
- [8] S. Delgado and P. Kotyczka, “Energy shaping for a wheeled inverted pendulum,” *IFAC-PapersOnLine*, vol. 48, no. 13, pp. 93–98, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.ifacol.2015.10.220>
- [9] J. Moreno-Valenzuela and C. Aguilar-Avelar, *Science and Engineering Motion Control of Underactuated Mechanical Systems*. Cham: Springer International, 2018.
- [10] H. K. Khalil, *Nonlinear Systems*, 2002.
- [11] R. S. Burns, *Advanced Control Engineering*, 2001.
- [12] C.-T. Chen, “Linear System Theory and Design,” p. 334, 1998.
- [13] K. Ogata, *Modern Control Engineering*, 4th ed. New Jersey: Prentice Hall, 2002.
- [14] J. A. Morales, M. A. Castro, D. Garcia, C. Higuera, and J. Sandoval, “IDA-PBC Controller Tuning Using Steepest Descent,” pp. 1–13, 2017.
- [15] Z. Li, C. Yang, and L. Fan, *Advanced Control of Wheeled Inverted Pendulum Systems*. Springer, 2013.
- [16] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 1999.
- [17] J. Nelder and R. Mead, “A simplex method for function minimization,” *The computer journal*, vol. 7, no. 4, pp. 308–313, 1965.
- [18] J. H. Mathews and K. D. Fink, *Numerical Methods Using MATLAB*, 3rd ed. Simon & Schuster, 1998.
- [19] “Function Reference: Sombbrero.” [Online]. Available: <https://octave.sourceforge.io/octave/function/sombbrero.html>

Modelado y control IDA-PBC de un robot bípedo tipo compás con fricción

Daniel H. Valles
 Tecnológico Nacional de México
 Instituto Tecnológico de La Paz
 La Paz, B.C.S., México
 ing.danielvalles@gmail.com

Jesús Sandoval
 Tecnológico Nacional de México
 Instituto Tecnológico de La Paz
 La Paz, B.C.S., México
 jsandoval@itlp.edu.mx

Víctor Santibáñez
 Tecnológico Nacional de México
 Instituto Tecnológico de La Laguna
 Torreón, Coahuila, México
 vsantiba@itlalaguna.edu.mx

Víctor De León Gómez
 CNRS in the Institut de Recherche en
 Laboratoire des Sciences du Numérique de Nantes (LS2N)
 Nantes, France
 Victor.Deleongomez@ls2n.fr

Resumen—En este trabajo se presenta un IDA-PBC (Interconexión e inyección de amortiguamiento—una extensión del Control Basado en Pasividad) con compensación de fricción dinámica para mantener la marcha de un robot bípedo de tipo compás con fricción en su coordenada actuada. Para el diseño del compensador de fricción se incorporó uno de los modelos más simples de fricción dinámica: el modelo de Dahl. Se presentan resultados de simulación sobre un modelo de robot bípedo tipo compás para validar el desempeño del controlador propuesto.

Abstract—In this paper, an IDA-PBC (Interconnection and damping injection, an extension of Passivity Based Control) control with dynamic friction compensation to hold the gait of a bipedal compass-type robot with friction in the actuated joint is presented. For the design of the friction compensator, one of the simplest models of dynamic friction was incorporated: the Dahl model. Simulation results are presented on a compass bipedal robot model to validate the performance of the proposed controller.

Index Terms—Robot bípedo tipo compás, IDA-PBC, fricción, modelo de Dahl

I. EL ROBOT BÍPEDO TIPO COMPÁS

El robot bípedo más simple está dotado de un par de piernas rígidas así como una articulación en la cadera y otra en el pie de soporte, como se muestra en la Figura 1. Este tipo de robot

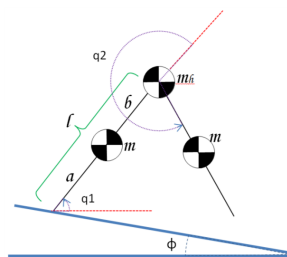


Figura 1. Diagrama del robot bípedo tipo compás.

de pares que propicien la locomoción, colocándose sobre una pendiente de ϕ grados, debido a la ganancia de energía en cada paso a razón de cambio en el potencial de la pendiente, mientras que cada impacto con la superficie disipará la energía extra, haciendo que cada paso iguale al anterior [1].

El movimiento del bípedo, está considerado únicamente en el plano sagital, por lo que al desplazarse, emulará un compás, tomando de ahí su nombre, Robot Bípedo Tipo Compás o por sus siglas RBTC.

I-A. Modelado matemático

El movimiento del bípedo se clasifica en dos etapas: Etapa de balanceo que se define como el periodo de transición previo a un paso cuando un pie no tiene contacto con la superficie; y etapa de impacto, presente cuando se tiene una o ambas piernas en contacto con la superficie. El comportamiento descrito es capturado por su modelo dinámico, el cual puede ser escrito con las ecuaciones de Euler-Lagrange, a partir del conocimiento del lagrangiano del robot: función de energía cinética menos la función de energía potencial [2]–[4]. El Robot Bípedo Tipo Compás, es uno de los robots más simples que permite un estudio sobre su dinámica que determine su marcha. Con ese fin, a continuación se presenta el modelo dinámico del robot antes mencionado.

I-A1. El robot bípedo como robot manipulador: Una manera de analizar el robot bípedo tipo compás, es haciéndolo desde la perspectiva de un robot manipulador [5], considerando las coordenadas en el pie de soporte q_1 como si se tratara de la base del robot manipulador, donde la coordenada se mide desde el eje horizontal hasta el eslabón en sentido antihorario, y en la cadera q_2 referido a la segunda articulación del robot manipulador, tomando la extensión del primer eslabón hasta el segundo, en sentido antihorario. En la Figura 2 se muestra el diagrama de cuerpo libre, en el cual se centra el análisis geométrico. En forma compacta, las ecuaciones de movimiento quedan de la siguiente manera:

$$\tilde{M}(q)\ddot{q} + \tilde{C}(q, \dot{q})\dot{q} + \tilde{g}(q) = \tilde{\tau} \quad (1)$$

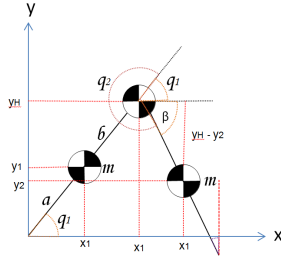


Figura 2. Diagrama de cuerpo libre del robot bípido con coordenadas en pie de soporte y cadera, q_1 y q_2 respectivamente.

donde $q = [q_1 \ q_2]^T$ es el vector de posición, mientras que las matrices $\tilde{M}(q)$, $\tilde{C}(q, \dot{q})$ y el vector de gravedad $\tilde{g}(q)$, se definen como:

$$\begin{aligned} \tilde{M}(q) &= \begin{bmatrix} a_1 & a_2 \\ a_2 & a_3 \end{bmatrix} \\ \tilde{C}(q, \dot{q}) &= \begin{bmatrix} -2mlb \sin(q_2) \dot{q}_2 & -mlb \sin(q_2) \dot{q}_2 \\ mlb \sin(q_2) \dot{q}_1 & 0 \end{bmatrix} \\ \tilde{g}(q) &= \begin{bmatrix} g_1 \\ mgb \cos(q_1 + q_2) \end{bmatrix} \end{aligned} \quad (2)$$

donde:

$$\begin{aligned} a_1 &= ma^2 + ml^2 + mb^2 + m_H l^2 + 2mlb \cos(q_2) \\ a_2 &= mb^2 + mlb \cos(q_2) \\ a_3 &= mb^2 \\ g_1 &= mga \cos(q_1) + mgl \cos(q_1) + mgb \cos(q_1 + q_2) + m_H gl \cos(q_1). \end{aligned}$$

Enseguida, el modelo de Dahl para un sistema subactuado incluyendo la fricción viscosa en las coordenadas actuadas puede ser escrito como [6]:

$$\tilde{f}(z, \dot{q}) = GG^T \Sigma_o z + GF_v G^T \dot{q} \quad (3)$$

donde $z \in \mathbb{R}^n$ es un vector de estados no medibles, $G = [I_m \ 0]^T \in \mathbb{R}^{n \times m}$, $\Psi = \text{diag}\{\frac{1}{f_{c1}}, \dots, \frac{1}{f_{cm}}\}$, es una matriz diagonal semidefinida positiva, donde f_{ci} , ($i = 1, 2, \dots, m$) denota los parámetros de Coulomb para cada una de las m coordenadas actuadas, $\Sigma_o = \text{diag}\{\sigma_{01}, \dots, \sigma_{0n}\}$ es una matriz semidefinida positiva, la cual contiene los parámetros de rigidez de las n coordenadas, $\tilde{f} \in \mathbb{R}^n$ es el vector de fuerzas debidas a la fricción, $F_v = \text{diag}\{f_{v1}, \dots, f_{vm}\}$, es una matriz diagonal, definida positiva, la cual contiene los coeficientes de fricción viscosa de las m coordenadas actuadas. Aunque el estado z no es medible, se asume que todos los parámetros del modelo son conocidos. Para el análisis se ha de considerar que la fricción está presente sólo en la coordenada actuada, lo cual es equivalente a expresar que $z = [z_1, \dots, z_m, z_{m+1}, \dots, z_n]^T$ con $z_k = 0$, para $k = m + 1, \dots, n$.

Tomando en cuenta la presencia de la fricción dinámica como se muestra en (3), la ecuación (1), queda representada como:

$$\tilde{M}(q)\ddot{q} + \tilde{C}(q, \dot{q})\dot{q} + \tilde{g}(q) = \tilde{\tau} - \tilde{f}(z, \dot{q}) \quad (4)$$

Los parámetros del robot bípido (1)-(2) y (5)-(6) han sido tomados de [7] (véase Tabla I).

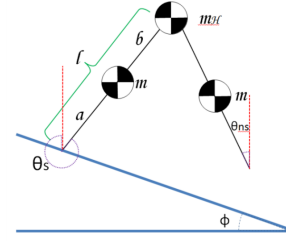


Figura 3. Diagrama del robot bípido tipo compás con coordenadas en ambos pies, θ_s y θ_{ns} .

I-A2. Modelo dinámico con coordenadas en ambos pies: La configuración clásica de análisis del RBTC consiste en definir las coordenadas generalizadas con respecto al eje vertical y el pie de *soporte*, así como al eje vertical y el pie de *no soporte*, es decir θ_s y θ_{ns} respectivamente, con ambas mediciones tomadas en sentido opuesto a las manecillas del reloj, como se muestra en la Figura 3. Tomando en cuenta la existencia de la fricción a través del modelo de fricción dinámica de Dahl, se puede obtener un modelo dinámico alterno del RBTC, esto es:

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta) = \tau - f(z, \dot{\theta}) \quad (5)$$

donde $\theta = [\theta_s \ \theta_{ns}]^T$ es el vector de posición, mientras que las matrices $M(\theta)$, $C(\theta, \dot{\theta})$, el vector $g(\theta)$ y el modelo de fricción dinámica de Dahl $f(z, \dot{\theta}) = GG^T \Sigma_o z + GF_v G^T \dot{\theta}$ se definen como:

$$\begin{aligned} M(\theta) &= \begin{bmatrix} ma^2 + ml^2 + m_c l^2 & mlb \cos(\theta_s - \theta_{ns}) \\ -mlb \cos(\theta_s - \theta_{ns}) & mb^2 \end{bmatrix} \\ C(\theta, \dot{\theta}) &= \begin{bmatrix} 0 & -mlb \sin(\theta_s - \theta_{ns}) \dot{\theta}_{ns} \\ mlb \sin(\theta_s - \theta_{ns}) \dot{\theta}_s & 0 \end{bmatrix} \\ g(\theta) &= \begin{bmatrix} -mga \sin(\theta_s) - mgl \sin(\theta_s) - m_c gl \sin(\theta_s) \\ mgb \sin(\theta_{ns}) \end{bmatrix} \\ f(z, \dot{\theta}) &= \sigma_{01} z_1 + f_{v1} \dot{\theta}_s \end{aligned} \quad (6)$$

Se considera la fricción como fuerza opuesta para este modelo con coordenadas en ambos pies (θ_s y θ_{ns}), por lo que su expresión matemática queda definida como:

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta) = \tau - (\sigma_{01} z_1 + f_{v1} \dot{\theta}_s) \quad (7)$$

I-A3. Relación entre dos modelos dinámicos del robot bípido tipo compás: A pesar de tratarse del mismo robot bípido, dos modelos dinámicos pueden representar las ecuaciones de movimiento del robot y la diferencia entre ellas es

Tabla I
PÁRAMETROS DEL ROBOT BÍPEDO TIPO COMPÁS

| Párametro | Valor [Unidad] | Descripción |
|-------------|--------------------------|---|
| m | 5 [Kg] | Masa de la pierna |
| m_H | 10 [Kg] | Masa de la cadera |
| a | 0.5 [m] | Distancia entre pie y centro de masa |
| b | 0.5 [m] | Distancia entre centro de masa y cadera |
| $l = a + b$ | 1 [m] | Distancia total de la pierna |
| g | 9.81 [m/s ²] | Aceleración debido a la gravedad |

la selección de las coordenadas generalizadas q y θ , como se muestra en las ecuaciones (4) y (5), respectivamente. Se puede observar que el modelo obtenido con las coordenadas en ambos pies (θ_s y θ_{ns}) es más simple en comparación con el modelo obtenido con las coordenadas en pie de soporte y cadera (q_1 y q_2). Sin embargo, se dificulta asignar en el modelo (5) los pares en el pie de soporte y cadera, teniendo en cuenta que puede tratarse como un sistema completamente actuado. Por tal motivo en este epígrafe, se presenta una matriz de transformación para expresar las coordenadas θ en términos de las coordenadas q , así como una matriz de transformación para asignar los pares u en pie de soporte y cadera, que se definirá matriz B , haciendo uso del modelo con coordenadas en θ (ver más detalles en [7]). Con este fin, se reescribe el modelo (5) como:

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + g(\theta) = Bu - f(z, \dot{\theta}) \quad (8)$$

donde la matriz B se define como:

$$B = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} \quad (9)$$

la cual más adelante se explica cómo se obtiene. Al obtener el modelo dinámico del RBTC con las coordenadas θ_s y θ_{ns} , no es posible obtener directamente el par aplicado en pie de soporte y la cadera; sino en el pie de soporte y en el pie de balanceo, a diferencia de cuando se usan las coordenadas q_1 y q_2 .

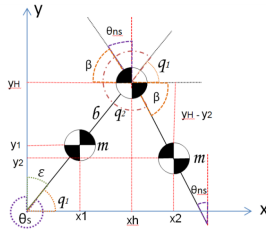


Figura 4. Diagrama de cuerpo libre que relaciona las coordenadas θ y las coordenadas q .

Un análisis geométrico, muestra la relación existente entre θ y q , donde $\theta = [\theta_s \ \theta_{ns}]^T$ y $q = [q_1 \ q_2]^T$ al analizar la Figura 4, se obtiene:

$$q_1 = \theta_s - \frac{3\pi}{2} \quad (10)$$

$$q_2 = \theta_{ns} - \theta_s + 3\pi. \quad (11)$$

Al obtener las derivadas temporales de (10) y (11), se puede expresar en notación matricial la siguiente relación:

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \dot{\theta}_s \\ \dot{\theta}_{ns} \end{bmatrix} \quad (12)$$

donde se define la matriz de transformación

$$A = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}$$

que permite expresar las coordenadas θ en términos de las coordenadas q y viceversa. La derivada temporal de (12) es:

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_s \\ \ddot{\theta}_{ns} \end{bmatrix} \quad (13)$$

por tanto, se tiene que $\dot{q} = A\dot{\theta}$ y $\ddot{q} = A\ddot{\theta}$. Finalmente, podemos reescribir (4) de la siguiente manera:

$$\widetilde{M}(q_1, q_2)\ddot{q} + \widetilde{C}(q_1, q_2, \dot{q}_1, \dot{q}_2)\dot{q} + \widetilde{g}(q_1, q_2) = \widetilde{\tau} - \widetilde{f}(z, \dot{q}_1, \dot{q}_2)$$

o bien,

$$\begin{aligned} & \widetilde{M}(\theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi)A\ddot{\theta} + \\ & \widetilde{C}(\theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi, \dot{\theta}_s, \dot{\theta}_{ns} - \dot{\theta}_s)A\dot{\theta} + \\ & \widetilde{g}(\theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi) = \\ & \widetilde{\tau} - \widetilde{f}(z_1, z_2, \theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi). \end{aligned} \quad (14)$$

Se puede observar que $\widetilde{M}A$ no es una matriz simétrica, por lo que para cumplir con las propiedades de simetría y positividad de la matriz M , se premultiplicará la ecuación por A^T , con la finalidad de obtener la simetría buscada, esto es,

$$\begin{aligned} & A^T \widetilde{M}(\theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi)A\ddot{\theta} + \\ & A^T \widetilde{C}(\theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi, \dot{\theta}_s, \dot{\theta}_{ns} - \dot{\theta}_s)A\dot{\theta} + \\ & A^T \widetilde{g}(\theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi) = \\ & A^T \widetilde{\tau} - A^T \widetilde{f}(z_1, z_2, \theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi) \end{aligned} \quad (15)$$

la cual resulta ser equivalente al modelo planteado en la ecuación (8). Esto permite concluir que el modelo planteado en (4) es equivalente al presentado en la forma tradicional (5), donde se tiene que:

$$\begin{aligned} M(\theta) &= A^T \widetilde{M}(\theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi)A\ddot{\theta} \\ C(\theta, \dot{\theta}) &= A^T \widetilde{C}(\theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi, \dot{\theta}_s, \dot{\theta}_{ns} - \dot{\theta}_s)A\dot{\theta} \\ g(\theta) &= A^T \widetilde{g}(\theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi) \\ f(z, \dot{\theta}) &= A^T \widetilde{f}(z_1, z_2, \theta_s - \frac{3\pi}{2}, \theta_{ns} - \theta_s + 3\pi) \end{aligned}$$

así como también resultan las siguientes identidades:

$$\begin{aligned} B &= A^T \\ u &= \widetilde{\tau} \end{aligned}$$

donde B corresponde a la matriz definida en (9), esto es,

$$B = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}$$

la cual permite usar el modelo (5) con los pares propuestos en el modelo (4).

II. DISEÑO DE UN IDA-PBC CON COMPENSACIÓN DE FRICCIÓN DINÁMICA PARA EL ROBOT BÍPEDO TIPO COMPÁS.

En el presente epígrafe se muestra el diseño de un IDA-PBC para el control de un robot bípido tipo compás con fricción en la coordenada actuada, el cual está basado en una extensión del método IDA-PBC que incorpora uno de los modelos más simples de fricción dinámica: El modelo de Dahl. En un sistema completamente actuado, todas las coordenadas generalizadas del sistema, se encuentran sujetas a un actuador, sin embargo al tener menos actuadores, este

sistema se denomina subactuado. Para el robot bípedo tipo compás sujeto a análisis, la coordenada subactuada es q_2 , que corresponde a la cadera. La fricción a considerar será la presente en el actuador, mas no en las articulaciones propias del RBTC, por tal motivo, en q_2 la fricción se considera nula.

Para el diseño, se explota el hecho de que la matriz de inercia M puede depender de la coordenada subactuada, lo cual facilita la solución de las ecuaciones diferencias parciales que es uno de los principales desafíos del método IDA-PBC. Por tanto, con la finalidad de aprovechar el modelo dinámico del robot bípedo analizado como robot manipulador (4), donde la matriz de inercia está en función de la coordenada subactuada q_2 , se considerará que el actuador del bípedo estará únicamente en el pie de soporte, es decir q_1 . El diseño del controlador que se muestra enseguida está basado en los trabajos reportados en [6]–[8].

II-A. Diseño de la ley de control

A continuación se muestra el diseño del controlador IDA-PBC, con énfasis en la obtención de M_d y J_2 . El modelo dinámico de la parte mecánica (1) puede ser escrito con las siguientes matrices $M(q)$, G y la función de energía potencial $V(q)$:

$$\begin{aligned} M(q) &= \begin{bmatrix} C_1 + C_2 + 2C_3 \cos(q_2) & C_2 + C_3 \cos(q_2) \\ C_2 + C_3 \cos(q_2) & C_2 \end{bmatrix} \\ G &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ V(q) &= C_4 g \sin(q_1) + C_5 g \sin(q_1 + q_2) \end{aligned}$$

donde:

$$\begin{aligned} C_1 &= ma^2 + ml^2 + m_H l^2 \\ C_2 &= mb^2 \\ C_3 &= mlb \\ C_4 &= m_H l + ma + ml \\ C_5 &= mb \end{aligned}$$

y $G \in \mathbb{R}^{n \times m}$ es la matriz de distribución de pares. La extensión del método IDA-PBC con fricción dinámica inicia con la función de energía total dada por:

$$H(q, p) = \frac{1}{2} p^T M(q)^{-1} p + V(q) \quad (16)$$

Ahora bien, tomando el modelo de fricción de Dahl (3), las ecuaciones de movimiento del sistema completo, puede escribirse como:

$$\begin{bmatrix} \dot{q} \\ \dot{p} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & I_n & 0 \\ -I_n & -GF_v G^T & -GG^T \\ 0 & [GG^T]^T & -G\Psi G^T \end{bmatrix} \begin{bmatrix} \nabla_q H \\ \nabla_p H \\ \nabla_z W \end{bmatrix} + \begin{bmatrix} 0 \\ Gu \\ 0 \end{bmatrix} \quad (17)$$

donde $u \in \mathbb{R}^m$ es el vector de entradas de control; y

$$W(z) = \frac{1}{2} z^T \Sigma_o z.$$

II-B. Dinámica deseada en lazo cerrado y ley de control

Siguiendo con el procedimiento del método IDA-PBC, una de sus principales características consiste en asignar una

estructura deseada en lazo cerrado con una función deseada dada por:

$$H_d(q, p) = \frac{1}{2} p^T M_d(q)^{-1} p + V_d(q) \quad (18)$$

donde $M_d = M_d^T > 0$ y V_d son las matrices deseadas, de inercia y energía potencial, respectivamente; por lo que el sistema en lazo cerrado queda expresado como:

$$\begin{bmatrix} \dot{q} \\ \dot{p} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & M^{-1} M_d & 0 \\ -M_d M^{-1} & J_2 - GK_v G^T & -GG^T \\ 0 & [GG^T]^T & -G\Psi G^T \end{bmatrix} \begin{bmatrix} \nabla_q H_d \\ \nabla_p H_d \\ \nabla_z W_d \end{bmatrix} \quad (19)$$

donde $J_2 = -J_2^T$ y $K_v = K_v^T > 0$, son matrices libres, $\tilde{z} = z - \hat{z}$, siendo \hat{z} la estimación del estado z , así como la introducción de una función semidefinida positiva

$$W_d(\tilde{z}) = \frac{1}{2} \tilde{z}^T \Sigma_o \tilde{z} \quad (20)$$

Uno de los principales desafíos del método IDA-PBC, es la solución del siguiente conjunto de Ecuaciones Diferenciales Parciales (EDP), llamadas *Matching Equations*:

$$\begin{aligned} G^\perp \{ \nabla q (p^T M^{-1} p) - M_d M^{-1} \nabla q (p^T M_d^{-1} p) + \\ 2J_2 M_d^{-1} p \} &= 0 \\ G^\perp \{ \nabla q V - M_d M^{-1} \nabla q V_d \} &= 0 \end{aligned} \quad (21)$$

donde $G^\perp \in \mathbb{R}^{(n-m) \times m}$ es una matriz aniquiladora por la izquierda de G , ya que $G^\perp G = 0$. Es importante resaltar que la compensación de fricción incluye una nueva inyección de amortiguamiento dada por $u = u_{es} + u_{di} + u_{fric}$, donde u_{es} es el control por moldeo de energía, u_{di} es la inyección de amortiguamiento y u_{fric} es la compensación de fricción [6].

Finalmente si existen soluciones M_d y V_d de (21), la ley de control que al sustituirla en (17) conduce a (19), es la siguiente:

$$u = \underbrace{[G^T G]^{-1} G^T [\nabla_q H - M_d M^{-1} \nabla_q H_d + J_2 M_d^{-1} p]}_{u_{di}} + \underbrace{F_v G^T M^{-1} p + G^T \Sigma_o \hat{z}}_{u_{fric}} + \underbrace{F_v G^T M^{-1} p + G^T \Sigma_o \hat{z}}_{u_{es}}$$

Debido a que la función de energía potencial no será moldeada para el robot bípedo, se tiene que $V_d = V$, así como tampoco se considera inyección de amortiguamiento, es decir, la matriz K_v es considerada nula ya que el objetivo de control no es llevar el sistema a un punto deseado sino generar un ciclo de marcha estable, y por tanto $u_{di} = 0$, la estructura final de la ley de control es:

$$u = \underbrace{[G^T G]^{-1} G^T [\nabla_q H - M_d M^{-1} \nabla_q H_d + J_2 M_d^{-1} p]}_{u_{fric}} + \underbrace{F_v G^T M^{-1} p + G^T \Sigma_o \hat{z}}_{u_{es}} \quad (22)$$

donde

$$J_2 = \begin{bmatrix} 0 & \tilde{p}^T \alpha(q_2) \\ -\tilde{p}^T \alpha(q_2) & 0 \end{bmatrix} = \begin{bmatrix} 0 & J_a \\ -J_a & 0 \end{bmatrix}$$

con

$$J_a = J_{\alpha_1} + J_{\alpha_2}$$

$$J_{\alpha_1} = -\left(\frac{r_1 p_2 - C_2 p_1}{K C_2 r_2}\right) (K-1) r_3 (-r_1 + K C_3 \cos(q_2))$$

$$J_{\alpha_2} = \left(\frac{r_1 p_1 + [K r_2 - r_1^2] p_2}{K C_2 r_2}\right) r_3 (K-1)$$

Ya definida la matriz J_2 , M_d queda finalmente como:

$$M_d = \begin{bmatrix} \frac{-K r_2 + r_1^2}{C_2} & C_2 + C_3 \cos(q_2) \\ C_2 + C_3 \cos(q_2) & C_2 \end{bmatrix}$$

donde:

$$r_1 = C_2 + C_3 \cos(q_2)$$

$$r_2 = C_3^2 \cos(q_2)^2 - C_1 C_2$$

$$r_3 = C_3 \sin(q_2)$$

II-C. Observador de fricción

En vista de que el modelo de Dahl es dinámico y tiene un estado no medible z , basado en el trabajo de [6] podemos diseñar un observador de fricción con la siguiente estructura:

$$\dot{\hat{z}} = [GG^T]^T [M^{-1}p - M_d^{-1}p] - G\Psi G^T \Sigma_o \hat{z} \quad (23)$$

el cual proporciona una estimación del estado no medible z . Una ventaja de la inclusión del modelo Dahl (3) en (17) es la simplicidad para diseñar el observador. Para aclarar este procedimiento, considere que, a partir de la definición $\tilde{z} = z - \hat{z}$, la derivada temporal puede escribirse como:

$$\dot{\tilde{z}} = \dot{z} - \dot{\hat{z}} \quad (24)$$

por tanto, el observador propuesto (23) se logra sustituyendo \dot{z} y $\dot{\hat{z}}$ respectivamente en (24). Finalmente, el observador de fricción del robot bípodo de tipo compás propuesto queda:

$$\dot{\hat{z}} = \frac{1}{\Delta} \left[[a_3 p_1 - a_2 p_2] - \frac{1}{K} [a_2 p_1 - a_2 p_2] \right] - \frac{\dot{q}_1}{f_{c1}} \sigma_{01} \hat{z}_1.$$

III. SIMULACIÓN Y RESULTADOS

El objetivo de control está dirigido en mantener la marcha del robot bípodo, a pesar de los efectos indeseables de la fricción. Con la finalidad de validar la ley de control propuesta, se llevaron a cabo simulaciones usando Matlab presentando resultados sobre el comportamiento de la marcha con diferentes valores de los parámetros. Para ello, primero se evalúa el comportamiento únicamente moldeando la función de energía cinética con diferentes valores de la ganancia K del controlador. Luego se considera la fricción y diferentes valores de K , sin incorporar el esquema de compensación de fricción dinámica. Por último se consideran los mismos parámetros del robot y el controlador donde la marcha se ve afectada debido a la fricción en la articulación actuada del robot, pero ahora aplicando el esquema de control con compensación de fricción dinámica con la finalidad de mantener una marcha estable. Los parámetros a modificar son: el coeficiente de fricción viscosa f_{v1} , el parámetro de fricción de Coulomb f_{c1} , el cual es estrictamente positivo de acuerdo a (I-A1) y el parámetro de rigidez σ_{01} .

III-A. IDA-PBC sin fricción viscosa

La primera simulación consiste en reproducir los resultados reportados en la literatura, por ejemplo [9]–[11], que tal como se ha mencionado, no contemplan fricción, sólo se realiza el moldeo de la función de energía cinética y son mostrados en la Figura 5, en donde se contemplaron tres valores de K . En el recuadro izquierdo, se consideró un valor de $K = 0.5$. En el recuadro central, el valor de $K = 0.9$. Y por último en el recuadro derecho $K = 1.5$.

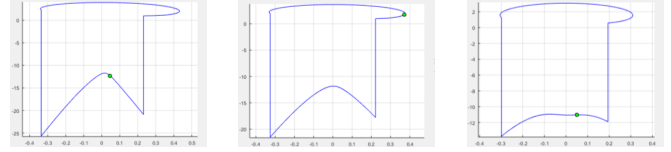


Figura 5. Ciclos límite del robot bípodo sin fricción viscosa. q_1 (eje horizontal) vs p_1 (eje vertical). El círculo en verde indica la condición inicial.

III-B. IDA-PBC con fricción dinámica

En este epígrafe se considera el mismo modelo del robot bípodo, pero ahora se incorpora el modelo de fricción dinámica sólo en la articulación actuada del robot, usando el modelo de fricción de Dahl. Cabe mencionar que si sólo se considerara fricción viscosa en la articulación actuada, el controlador lo pudiera cancelar de manera directa y recuperar el mismo comportamiento obtenido cuando la fricción viscosa está ausente en dicha articulación. Sin embargo, es conocido que en la mayoría de los sistemas mecánicos, la fricción viscosa no aproxima el comportamiento real a bajas velocidades. Por tal motivo, en este trabajo se considera aproximar la fricción en la articulación actuada lo mejor posible con el modelo de fricción de Dahl. En la Tabla II, se aprecia un conjunto de valores de los parámetros del modelo de Dahl de la articulación actuada y la ganancia correspondiente al controlador.

Tabla II
PARÁMETROS DEL MODELO DE FRICCIÓN DE DAHL DEL ROBOT BÍPEDO TIPO COMPÁS Y LA GANANCIA DEL CONTROLADOR (22)

| Caso | f_{v1} | f_{c1} | σ_{01} | K |
|------|----------|----------|---------------|-----|
| (a) | 0.01 | 0.5 | 5 | 0.5 |
| (b) | 0.1 | 0.9 | 10 | 0.9 |
| (c) | 0.33 | 1 | 15.4 | 1.5 |

En los casos (a) y (b) de la Tabla II, la marcha se mantiene estable, mas no la periodicidad del paso del robot. Esto se traduce como una marcha con pasos diferentes conforme avanza el robot (ver Figura 6). Al comparar las gráficas de la Figura 6 en el recuadro izquierdo ($K = 0.5$) y el recuadro derecho ($K = 0.9$), se puede apreciar que la longitud de los pasos se reduce en función del incremento de los valores de los parámetros del modelo de Dahl y en especial de la ganancia K del controlador. Las gráficas se presentan en coordenadas q y p . En la gráfica de la Figura 6 en el recuadro derecho, cuando $K = 1.5$, se muestra un movimiento inicial mínimo por parte de las piernas, indicando que los pasos no sólo son cortos,

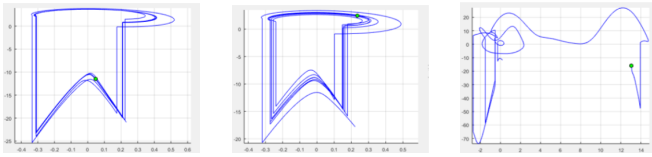


Figura 6. Ciclo límite de la marcha del robot bípedo, con fricción dinámica sólo en la coordenada actuada. q_1 (eje horizontal) vs p_1 (eje vertical). El círculo en verde indica la condición inicial.

sino también lentos, lo que ocasiona que el robot pierda su marcha y finalmente caiga.

En la Figura 7, se aprecia un instante en que el robot bípedo no logra mantener la marcha.

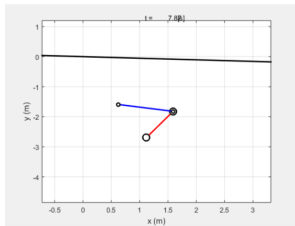


Figura 7. Imagen del instante en la animación en que el robot bípedo pierde la marcha por los efectos de la fricción

III-C. IDA-PBC con compensación de fricción dinámica

En el epígrafe anterior se mostraron resultados de simulación, cuando la fricción en la coordenada actuada se aproximaba con el modelo de fricción de Dahl y el controlador IDA-PBC no compensa dicha fricción. Por tanto, en este epígrafe se muestran los resultados cuando la fricción dinámica en la articulación actuada es compensada con el controlador (22). Es importante destacar que este controlador (22) regula la velocidad y longitud de los pasos, haciendo que se logre una similitud con la marcha que no contempla fricción, reproduciendo un conjunto de condiciones iniciales muy cercanos entre sí. Esto se puede ver en la Figura 8, compuesta por la descripción gráfica de las tres condiciones de K que se manejaron anteriormente. Se observa que la ley de control (22) logra mantener la marcha del robot con fricción dinámica en la articulación actuada. En la Figura 9, se muestra una imagen de un instante en la animación en la que el robot avanza aun cuando se considera fricción en la coordenada actuada.

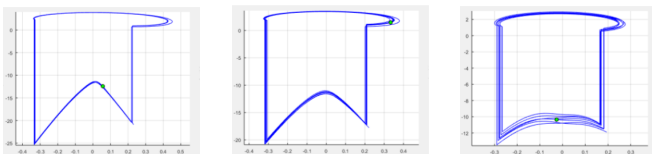


Figura 8. Ciclo límite de la marcha del robot bípedo, con compensación de fricción dinámica sólo en la coordenada actuada. q_1 (eje horizontal) vs p_1 (eje vertical). El círculo en verde indica la condición inicial.

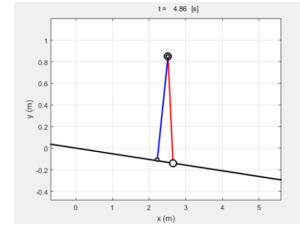


Figura 9. Instante de la animación de la marcha del robot bípedo con control con compensación de fricción dinámica en la articulación actuada.

IV. CONCLUSIONES

Se ha presentado un esquema de control IDA-PBC con compensación de fricción dinámica de un modelo de robot bípedo tipo compás para mantener la marcha, cuando se considera fricción dinámica en la articulación actuada del robot. La fricción dinámica es representada con el modelo de fricción de Dahl. Resultados de simulación ilustran el desempeño del esquema de control propuesto. La evidencia mostrada en las simulaciones resalta cómo la fricción capturada por el modelo de Dahl ocasiona que el robot pueda caer cuando el controlador no compensa la fricción, mientras que al ser considerada la compensación, se logra mantener la marcha con una ligera variación en la longitud y velocidad del paso del robot.

REFERENCIAS

- [1] McGeer T., *Passive dynamic walking*, International Journal of Robotics Research, Vol. 9, No. 2, pp. 62-82, 1990.
- [2] Contreras L. y Vargas L., *Generación de modelos de caminata bípeda a través de diversas técnicas de modelamiento*, Revista Científica y Tecnológica de la Facultad de Ingeniería, Universidad Distrital Francisco José de Caldas, Bogotá, Colombia, Vol. 11, No. 2, pp. 4-13, 2006
- [3] Duarte J., Montero B., Ospina-Henao P. y González E., *Mecánica lagrangiana para el modelado dinámico y simulación mediante Simulink-Matlab de un péndulo de Furuta*, Congreso Internacional de Ciencias Básicas e Ingeniería, Villavicencio, Colombia, 2016.
- [4] Gómez-Estern F., *Control de sistemas no lineales basado en la estructura hamiltoniana*, Tesis Doctoral, Universidad de Sevilla, 2002.
- [5] De León Gómez V., Santibáñez V., Moreno J., *A procedure to find equivalences among dynamic models of planar biped robots*. Simulation modelling practice and theory, 75, pp. 48-66, 2017.
- [6] Sandoval J., Kelly R. and Santibáñez V., *Interconnection and damping assignment passivity based control of a class of underactuated mechanical systems with dynamic friction*, International Journal of Robust and Nonlinear Control, pp. 738-751, 2011.
- [7] De León Gómez V., *Control de robots bípedos*, Tesis Doctoral, Tecnológico Nacional de México-Instituto Tecnológico de la Laguna, Torreón, Coahuila, México, 2014.
- [8] De León V., Santibáñez V. and Sandoval J., *Interconnection and damping assignment passivity based control for a compass-like biped robot*, International Journal of Advanced Robotics Systems, pp. 1-18, July-August 2017.
- [9] Goswami A., Thuilot B. and Espiau B., *Compass-like biped robot part I: Stability and bifurcation of passive gaits*, Institut National de Recherche en Informatique et en Automatique (INRIA), 1996.
- [10] Grizzle J., Hurst J., Morris B., Park H.W. and Sreenath K., *Mabel, a new robotic bipedal walker and runner*, American Control Conference, St. Louis, USA, pp. 2030-2036, 2009.
- [11] Spong M.W., *Passivity based control of the compass gait biped*, IFAC World Congress, Beijing, China, pp. 12-16, 1999.

Modelado y Comparativa de un Controlador LQR y un Controlador Difuso en un Péndulo de Furuta

J.A. Coronel¹, J. L. Gómez¹, J. A. Morales¹, J. Moyrón¹, I. Santillán¹.

¹División de Estudios de Posgrado e Investigación, Instituto Tecnológico de La Paz, La Paz, B.C.S, México

Resumen—En este documento se compara el desempeño de dos controladores aplicados a un péndulo de Furuta, un Regulador Cuadrático Lineal (LQR, por sus siglas en inglés) y un Controlador de lógica difusa (FLC, por sus siglas en inglés). El péndulo de Furuta es un sistema conformado por un brazo rotatorio en el cual se coloca un péndulo que rota libremente en un plano perpendicular al brazo. La complejidad del mismo radica en que las ecuaciones que rigen el comportamiento dinámico del péndulo no son lineales, además de que el sistema cuenta únicamente con un actuador ubicado en el brazo rotatorio. El desempeño es evaluado de acuerdo a la rapidez con que se logra llevar el péndulo a una posición vertical desde el reposo, y qué tanto tiempo les toma a los controladores llevar las velocidades del brazo y péndulo a cero.

Index Terms—Péndulo, Furuta, Sistemas dinámicos, Control.

I. INTRODUCCIÓN

El péndulo de Furuta o péndulo invertido rotatorio ([1], [2], [3]), es un buen ejemplo para comprobar la teoría del control moderno. Ha sido utilizado para ilustrar ideas en control lineal como la estabilidad de sistemas inestables [4]. Fue diseñado por el Doctor Katsuhisa Furuta del Instituto Tecnológico de Tokio en 1992, siendo un mecanismo diferente al péndulo invertido, pues este no presentaba las limitantes físicas de los rieles sobre los cuales se desliza el carro. Actualmente, sistemas como el que se estudia tienen aplicación en robótica, como se observa en [5].

Este sistema sub-actuado permitió realizar la comparación de dos leyes de control propuestas en el documento, una obtenida por un controlador LQR, y con base en esta, una ley de control producto de un controlador difuso (FLC).

Se presenta el modelo dinámico obtenido a partir de las ecuaciones de movimiento de Lagrange, considerando el sistema formado por una cadena cinemática abierta, y las articulaciones rotacionales y traslacionales, para, posteriormente, realizar las simulaciones correspondientes con cada controlador y evaluar su desempeño.

II. MODELO DINÁMICO

Llámesse al brazo rotario y al péndulo eslabón 1 y eslabón 2 respectivamente, con parámetros l_1 , m_1 , l_2 , m_2 y dos ángulos q_1 y q_2 que describen la posición del eslabón 1 y eslabón 2 (Figura 1). Observe además que hay dos marcos de referencia, O_{1xyz} y $O_{2x'y'z'}$, con los cuales se describe la posición de los

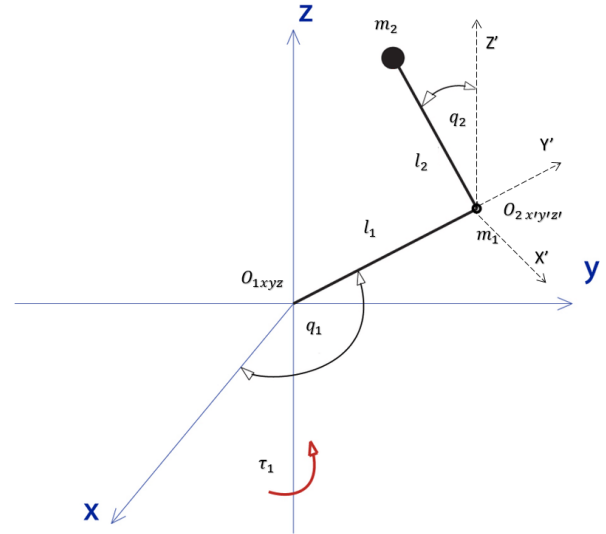


Figura 1: Diagrama del Péndulo de Furuta

centros de masa (ubicado en el extremo de los eslabones por simplicidad) de los eslabones 1 y 2.

$$\vec{r}_1 = \begin{pmatrix} l_1 \cos(q_1) \\ l_1 \sin(q_1) \\ 0 \end{pmatrix} \quad (1)$$

$$\vec{r}_{2/O_2} = \begin{pmatrix} 0 \\ l_2 \sin(q_2) \\ l_2 \cos(q_2) \end{pmatrix} \quad (2)$$

El marco $O_{2x'y'z'}$, anclado en la articulación de los eslabones, experimentará una traslación y rotación en consecuencia del movimiento del eslabón 1, por ello, se expresarán las coordenadas del centro de masa del eslabón 2 con respecto al marco O_{1xyz} . Ya que la rotación es con respecto al eje z del marco O_{1xyz} , se multiplica $R_z(q_1)$ por el vector con las coordenadas del centro de masa del eslabón 2.

$$R_z(q_1) = \begin{pmatrix} \cos(q_1) & -\sin(q_1) & 0 \\ \sin(q_1) & \cos(q_1) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3)$$

$$\vec{r}_2 = \vec{r}_1 + R_z \vec{r}_{2/O_2} \quad (4)$$

$$\vec{r}_2 = \begin{pmatrix} l_1 \cos(q_1) - l_2 \sin(q_1) \sin(q_2) \\ l_1 \sin(q_1) + l_2 \cos(q_1) \sin(q_2) \\ l_2 \cos(q_2) \end{pmatrix} \quad (5)$$

El *Lagrangiano* $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})$ del sistema es la diferencia de su energías cinética K y potencial U :

$$\mathcal{L}(\mathbf{q}(t), \dot{\mathbf{q}}(t)) = \mathcal{K}(\mathbf{q}(t), \dot{\mathbf{q}}(t)) - \mathcal{U}(\mathbf{q}(t)) \quad (6)$$

La energía cinética $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}})$ viene dada por la suma de las energías cinéticas $\mathcal{K}_1(\mathbf{q}_1, \dot{\mathbf{q}}_1)$ y $\mathcal{K}_2(\mathbf{q}_1, \mathbf{q}_2, \dot{\mathbf{q}}_1, \dot{\mathbf{q}}_2)$ correspondientes a los eslabones 1 y 2.

$$\mathcal{K}(\mathbf{q}(t), \dot{\mathbf{q}}(t)) = \frac{1}{2}m_1\|\dot{\mathbf{r}}_1\|^2 + \frac{1}{2}m_2\|\dot{\mathbf{r}}_2\|^2 \quad (7)$$

La energía potencial \mathcal{U} del eslabón 1 es nula, ya que este rota en el plano xy , mientras que para el eslabón 2, la energía potencial es:

$$\mathcal{U}(\mathbf{q}(t)) = gl_2m_2 \cos q_2 \quad (8)$$

Sustituyendo y simplificando mediante identidades trigonométricas, el *Lagrangiano* $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})$ del sistema es:

$$\begin{aligned} \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = & l_1^2 \dot{q}_1^2 (m_1 + m_2) + \frac{l_2^2 m_2 q_2^2}{2} \\ & + \frac{l_2^2 m_2 \sin(q_2)^2}{2} - gl_2 m_2 \cos(q_2) \\ & + l_1 l_2 m_2 \dot{q}_1 \dot{q}_2 \cos(q_2) \end{aligned} \quad (9)$$

Las ecuaciones de movimiento de Lagrange para un mecanismo de cadena cinemática abierta de dos grados de libertad viene dada por:

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} \right] - \frac{\partial \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} = \boldsymbol{\tau} \quad (10)$$

Donde $\boldsymbol{\tau}$ es el vector de fuerzas y pares externos en cada articulación. Por (10) tenemos:

$$\begin{aligned} \tau_1 = & (m_1 + m_2)l_1^2 \ddot{q}_1 + 2m_2 l_2^2 \dot{q}_1 \dot{q}_2 \sin(q_2) \cos(q_2) \\ & + m_2 l_2^2 \ddot{q}_1 \sin^2(q_2) + m_2 l_1 l_2 \ddot{q}_2 \cos(q_2) \\ & - m_2 l_1 l_2 \dot{q}_2^2 \sin(q_2) \end{aligned} \quad (11)$$

$$\begin{aligned} \tau_2 = & m_2 l_1 l_2 \dot{q}_1 \cos(q_2) - l_2^2 m_2 \dot{q}_1^2 \sin(q_2) \cos(q_2) \\ & + l_2^2 m_2 \ddot{q}_2 - m_2 l_2 g \sin(q_2) \end{aligned} \quad (12)$$

Las cuales se pueden escribir de modo compacto como:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \quad (13)$$

$$\mathbf{M}(\mathbf{q}) = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \quad (14)$$

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} \quad (15)$$

$$\mathbf{g}(\mathbf{q}) = \begin{pmatrix} 0 \\ -gl_2 m_2 \sin(q_2) \end{pmatrix} \quad (16)$$

$$M_{11} = (m_1 + m_2) l_1^2 + m_2 l_2^2 \sin^2(q_2)$$

$$M_{12} = l_1 l_2 m_2 \cos(q_2)$$

$$M_{21} = l_1 l_2 m_2 \cos(q_2)$$

$$M_{22} = l_2^2 m_2$$

$$C_{11} = l_2^2 m_2 \dot{q}_2 \cos(q_2) \sin(q_2)$$

$$C_{12} = l_2^2 m_2 \dot{q}_2 \cos(q_2) \sin(q_2) - l_1 l_2 m_2 \dot{q}_2 \sin(q_2)$$

$$C_{21} = -l_2^2 m_2 \dot{q}_1 \cos(q_2) \sin(q_2)$$

$$C_{22} = 0$$

Cabe mencionar que debido a la ausencia de fuerzas externas en el eslabon 2, $\tau_2 = 0$.

A partir de (13), se obtiene el vector $\ddot{\mathbf{q}}$.

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}[\boldsymbol{\tau} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q})] \quad (17)$$

$$\ddot{\mathbf{q}} = \begin{pmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{pmatrix} \quad (18)$$

$$\begin{aligned} \ddot{q}_1 = & \frac{-1}{\alpha[1 - \cos^2(q_2)] + \beta} [m_2 l_2^2 \sin(q_2) \cos(q_2) (\dot{q}_1 \dot{q}_2 + q_2^2) \\ & + l_1 l_2 \sin(q_2) (m_1 \dot{q}_1^2 \cos^2(q_2) - m_2 \dot{q}_2^2) \\ & + gl_1 m_1 \sin(q_2) \cos(q_2) - \tau] \end{aligned}$$

$$\begin{aligned} \ddot{q}_2 = & \frac{l_2 \sin(2q_2) \dot{q}_1^2 + 2g \sin(q_2) (b_1 + a_3 \sin(q_2)^2)}{2l_2 (\alpha \sin(q_2)^2 + \beta)} \\ & - \frac{l_1 \cos(q_1) [b_2 (\dot{q}_2 - 1) + a_2 \sin(q_2) \dot{q}_2^2 + \tau]}{(a_1 l_2 + a_3 l_2) \sin(q_2)^2 + \beta l_2} \end{aligned}$$

Con las constantes:

$$\begin{aligned} \alpha = l_1^2 m_1 + l_2^2 m_2 & \quad \beta = l_1^2 m_2 \\ a_1 = m_1 l_1^2 & \quad a_2 = l_1 l_2 m_2 & \quad a_3 = m_2 l_2^2 \\ b_1 = l_1^2 + m_1 + l_1^2 m_2^2 & \quad b_2 = -0,5 m_2 \sin(2q_2) l_2^2 \dot{q}_2 \end{aligned}$$

Definiendo \mathbf{x} como vector de estados:

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \quad (19)$$

$$x_1 = q_1 \quad x_2 = q_2 \quad x_3 = \dot{q}_1 \quad x_4 = \dot{q}_2$$

Es posible representar (18) como:

$$\frac{d}{dt} \mathbf{x} = \begin{pmatrix} \dot{x}_3 \\ \dot{x}_4 \\ \ddot{x}_3 \\ \ddot{x}_4 \end{pmatrix} \quad (20)$$

$$\dot{x}_3 = \ddot{q}_1 \quad \dot{x}_4 = \ddot{q}_2$$

III. PUNTOS DE EQUILIBRIO Y OBJETIVO DE CONTROL

Según [6], un vector $\mathbf{x}_e \in \mathbb{R}$ constante es un equilibrio o estado del sistema si $f(t, \mathbf{x}_e) = \mathbf{0}$.

Con $\dot{q}_1 = \dot{q}_2 = \ddot{q}_2 = 0$ y (18) se tiene:

$$\ddot{q}_1 \Big|_{\mathbf{x}_e} = \sin(q_2)[l_1 l_2 m_1 \dot{q}_1^2 \cos^2(q_2) + g l_1 m_1 \cos(q_2)] \quad (21)$$

$$\ddot{q}_2 \Big|_{\mathbf{x}_e} = l_2^2 m_2 \dot{q}_1^2 \sin(q_2) \cos(q_2) + g l_2 m_2 \sin(q_2) \quad (22)$$

Si en (22), $\sin q_2 = 0$, entonces $q_2 = n\pi; \forall n \in \mathbb{Z}, q_1 \in \mathbb{R}$.

Obsérvese que dichos valores también cumplen la condición en (21).

Por lo tanto, el vector de equilibrio \mathbf{x}_e es:

$$\mathbf{x}_e = \begin{pmatrix} k_1 \\ n\pi \\ 0 \\ 0 \end{pmatrix} \forall k_1 \in \mathbb{R}, n \in \mathbb{Z} \quad (23)$$

Se plantea como objetivo de control llevar el sistema al reposo ($\dot{q}_1 = 0, \dot{q}_2 = 0$) en el punto de equilibrio $q_2 = 0$.

IV. CONTROLADOR LQR

Se linealiza (20) alrededor del punto de equilibrio \mathbf{x}_e y un $\tau_e = (0)$, considerando $k_2 = 0$. Sea $f_1 = x_3, f_2 = x_4, f_3 = \dot{q}_1, f_4 = \dot{q}_2$, tenemos:

$$\begin{aligned} \frac{\partial f_1}{\partial x_1} \Big|_{\mathbf{x}_e} &= 0 & \frac{\partial f_2}{\partial x_1} \Big|_{\mathbf{x}_e} &= 0 & \frac{\partial f_3}{\partial x_1} \Big|_{\mathbf{x}_e} &= 0 & \frac{\partial f_4}{\partial x_1} \Big|_{\mathbf{x}_e} &= 0 \\ \frac{\partial f_1}{\partial x_2} \Big|_{\mathbf{x}_e} &= 0 & \frac{\partial f_2}{\partial x_2} \Big|_{\mathbf{x}_e} &= 0 & \frac{\partial f_3}{\partial x_2} \Big|_{\mathbf{x}_e} &= a_1 & \frac{\partial f_4}{\partial x_2} \Big|_{\mathbf{x}_e} &= a_2 \\ \frac{\partial f_1}{\partial x_3} \Big|_{\mathbf{x}_e} &= 1 & \frac{\partial f_2}{\partial x_3} \Big|_{\mathbf{x}_e} &= 0 & \frac{\partial f_3}{\partial x_3} \Big|_{\mathbf{x}_e} &= 0 & \frac{\partial f_4}{\partial x_3} \Big|_{\mathbf{x}_e} &= 0 \\ \frac{\partial f_1}{\partial x_4} \Big|_{\mathbf{x}_e} &= 0 & \frac{\partial f_2}{\partial x_4} \Big|_{\mathbf{x}_e} &= 1 & \frac{\partial f_3}{\partial x_4} \Big|_{\mathbf{x}_e} &= 0 & \frac{\partial f_4}{\partial x_4} \Big|_{\mathbf{x}_e} &= 0 \\ \frac{\partial f_1}{\partial \tau} \Big|_{\tau_e} &= 0 & \frac{\partial f_2}{\partial \tau} \Big|_{\tau_e} &= 0 & \frac{\partial f_3}{\partial \tau} \Big|_{\tau_e} &= b_1 & \frac{\partial f_4}{\partial \tau} \Big|_{\tau_e} &= b_2 \end{aligned}$$

$$\begin{aligned} a_1 &= \frac{-m_2 g}{l_1 m_1} & a_2 &= \frac{g(m_1 + m_2)}{l_2^2 m_1} \\ b_1 &= \frac{1}{l_1^2 m_1} & b_2 &= \frac{-1}{l_1 l_2 m_1} \end{aligned}$$

Dicha linealización puede escribirse de la forma:

$$\frac{d}{dt} \mathbf{x} = \mathbf{A} \mathbf{x} + \mathbf{B} \tau \quad (24)$$

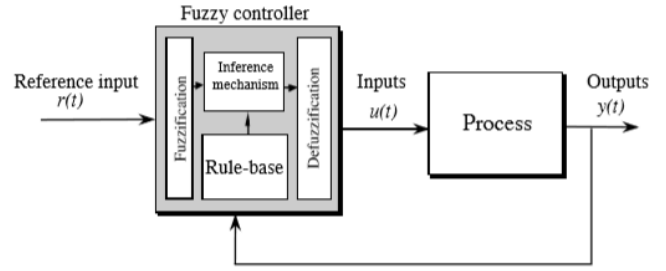
Con:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & a_1 & 0 & 0 \\ 0 & a_2 & 0 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ b_1 \\ b_2 \end{bmatrix} \quad (25)$$

Donde [7] las denomina como \mathbf{A} la matriz de estados y \mathbf{B} la matriz de entradas.

De acuerdo a [8], una vez obtenido el sistema lineal de (24) se determina la matriz \mathbf{K} que haga que $\mathbf{x} = 0$ sea

Figura 2: Controlador difuso



asintóticamente estable y que minimice el índice de desempeño cuadrático dado por:

$$J = \int_0^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt \quad (26)$$

Donde \mathbf{Q} y \mathbf{R} son matrices cuadradas, simétricas y definidas positivas.

$$\mathbf{K} = \mathbf{R}^{-1} \mathbf{B} \mathbf{P} \quad (27)$$

La matriz \mathbf{K} que cumple lo anterior es igual a $[-1,0000 \quad -18,3774 \quad -1,3169 \quad -2,7794]$, con $\mathbf{R} = 1$ y \mathbf{Q} siendo la matriz identidad de 4×4 .

V. CONTROLADOR DIFUSO

La arquitectura propuesta por [9] para un controlador difuso se muestra en la Figura 2.

1. Reglas (Rule-base): Contiene el conocimiento de la planta como un grupo de reglas, que describen la mejor manera de controlar el sistema.
2. Mecanismo de inferencia (Inference mechanism): Evalúa las reglas que son relevantes en el estado actual del sistema y decide qué entrada debería aplicarse a la planta.
3. Fuzzificación (Fuzzification): Modifica las entradas del controlador para ser interpretadas y comparadas por las reglas establecidas.
4. Defuzzificación (Defuzzification): Convierte las conclusiones del mecanismo de inferencia en entradas para la planta.

V-A. Reglas

Las variables que se consideran para la formulación de las reglas del controlador serán e y $\frac{d}{dt}e$, donde $e = \theta_{deseada} - \theta_{real}$, siendo $\theta_{deseada} = 0$ el cual representa el objetivo de control.

En la Tabla 1 se muestran el conjunto de reglas a utilizar para el sistema, en donde de manera lingüística, EGN representa error grande negativo, EPN error pequeño negativo, EPP error pequeño positivo, EGP error grande positivo, EP error pequeño, MRM muy rápido negativo, RN rápido negativo, L lento, RP rápido positivo, MRP muy rápido positivo, PGP par grande positivo, PPP par pequeño positivo, PMP par muy pequeño, PPN par pequeño negativo y PGN par grande negativo. El signo representa el sentido en el que abre el ángulo θ , siendo positivo cuando este abra en sentido antihorario.

Tabla 1: Reglas para el péndulo de Furuta.

| Par τ | $\frac{d}{dt}e$ | | | | |
|------------|-----------------|-----|-----|-----|-----|
| | MRN | RN | L | RP | MRP |
| e | EGN | PGP | PGP | PPP | PMP |
| | EPN | PGP | PGP | PPP | PPN |
| | EP | PGP | PPP | PMP | PPN |
| | EPP | PPP | PMP | PPN | PGN |
| | EGP | PMP | PPN | PGN | PGN |

V-B. Mecanismo de inferencia

En [10] se explica que las funciones de membresía constituyen un método utilizado para representar los conjuntos difusos, de modo que a cada punto en el espacio se le asigna un valor o grado de pertenencia a mencionado conjunto. Según [9], estas pueden ser clasificadas por su forma, gaussiana, triangular, trapezoidal y sigmoideal. Las funciones de membresía triangulares se definen la siguiente manera:

$$\mu^L(u) = \begin{cases} 1 & u \leq c^L \\ \max(0, 1 - \frac{c^L - u}{0,5w}) & u > c \end{cases} \quad (28)$$

$$\mu^C(u) = \begin{cases} \max(0, 1 + \frac{u-c}{0,5w}) & u \leq c \\ \max(0, 1 + \frac{c-u}{0,5w}) & u > c \end{cases} \quad (29)$$

$$\mu^R(u) = \begin{cases} \max(0, 1 + \frac{u-c^R}{0,5w^R}) & u \leq c^R \\ 1 & u > c^R \end{cases} \quad (30)$$

Donde c representa el centro del triángulo, w el ancho de la base, c^L y c^R el punto de saturación, w^L y w^R el tamaño del dominio de μ^L o μ^R que es diferente a la unidad y diferente de cero. Las siguientes ecuaciones componen el mecanismo de inferencia, y se muestran en las figuras 2, 3 y 4.

$$\mu_{EGN}(e) = \begin{cases} 1 & e \leq -\frac{\pi}{4} \\ \max(0, 1 - \frac{0,25\pi+e}{0,125\pi}) & e > -\frac{\pi}{4} \end{cases} \quad (31)$$

$$\mu_{EPN}(e) = \begin{cases} \max(0, 1 + \frac{e+0,125\pi}{0,125\pi}) & e \leq -\frac{\pi}{8} \\ \max(0, 1 - \frac{0,125\pi+e}{0,125\pi}) & e > -\frac{\pi}{8} \end{cases} \quad (32)$$

$$\mu_{EP}(e) = \begin{cases} \max(0, 1 + \frac{e}{0,04166\pi}) & e \leq 0 \\ \max(0, 1 - \frac{e}{0,4166\pi}) & e > 0 \end{cases} \quad (33)$$

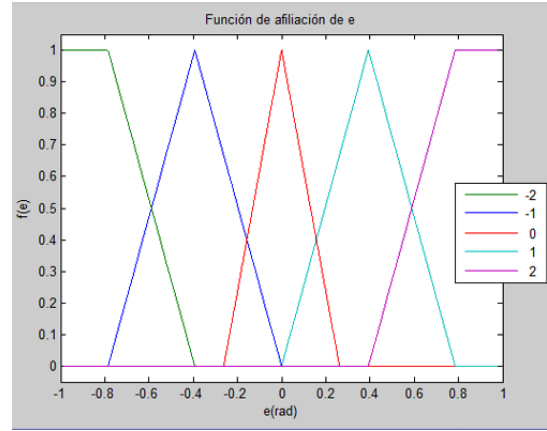
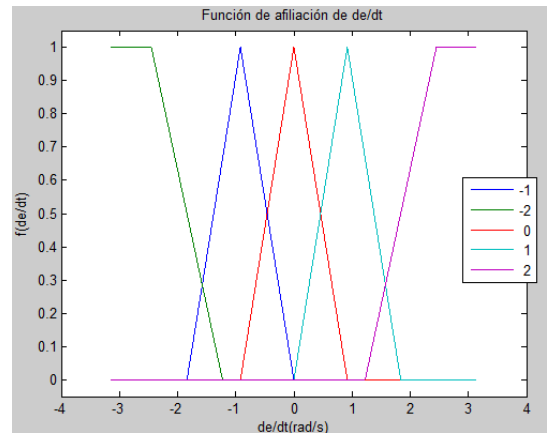
$$\mu_{EPP}(e) = \begin{cases} \max(0, 1 + \frac{e-0,125}{0,125\pi}) & e \leq \frac{\pi}{8} \\ \max(0, 1 + \frac{0,125\pi-e}{0,125\pi}) & e > \frac{\pi}{8} \end{cases} \quad (34)$$

$$\mu_{EGN}(e) = \begin{cases} \max(0, 1 + \frac{e-0,25}{0,125\pi}) & e \leq \frac{\pi}{4} \\ 1 & e > \frac{\pi}{4} \end{cases} \quad (35)$$

$$\sigma_{MRN}(\frac{de}{dt}) = \begin{cases} 1 & \frac{de}{dt} \leq -\frac{\pi}{4} \\ \max(0, 1 - \frac{0,25\pi-\frac{de}{dt}}{0,125\pi}) & e > -\frac{\pi}{4} \end{cases} \quad (36)$$

$$\sigma_{RN}(\frac{de}{dt}) = \begin{cases} \max(0, 1 + \frac{\frac{de}{dt}+0,125\pi}{0,125\pi}) & \frac{de}{dt} \leq -\frac{\pi}{8} \\ \max(0, 1 - \frac{0,125\pi-\frac{de}{dt}}{0,125\pi}) & \frac{de}{dt} > -\frac{\pi}{8} \end{cases} \quad (37)$$

$$\sigma_L(e) = \begin{cases} \max(0, 1 + \frac{\frac{de}{dt}}{0,125\pi}) & \frac{de}{dt} \leq 0 \\ \max(0, 1 - \frac{\frac{de}{dt}}{0,125\pi}) & \frac{de}{dt} > 0 \end{cases} \quad (38)$$

 Figura 3: Funciones de membresía e .

 Figura 4: Funciones de membresía $\frac{de}{dt}$.


$$\sigma_{RP}(\frac{de}{dt}) = \begin{cases} \max(0, 1 + \frac{\frac{de}{dt}-0,125}{0,125\pi}) & \frac{de}{dt} \leq \frac{\pi}{8} \\ \max(0, 1 + \frac{0,125\pi-\frac{de}{dt}}{0,125\pi}) & \frac{de}{dt} > \frac{\pi}{8} \end{cases} \quad (39)$$

$$\sigma_{MRP}(\frac{de}{dt}) = \begin{cases} \max(0, 1 + \frac{\frac{de}{dt}-0,25}{0,125\pi}) & \frac{de}{dt} \leq \frac{\pi}{4} \\ 1 & \frac{de}{dt} > \frac{\pi}{4} \end{cases} \quad (40)$$

$$\xi_{PGN}(\tau) = \begin{cases} \max(0, 1 + \frac{\tau+0,3}{0,2}) & \tau \leq -0,3 \\ \max(0, 1 - \frac{0,3+\tau}{0,2}) & \tau > -0,3 \end{cases} \quad (41)$$

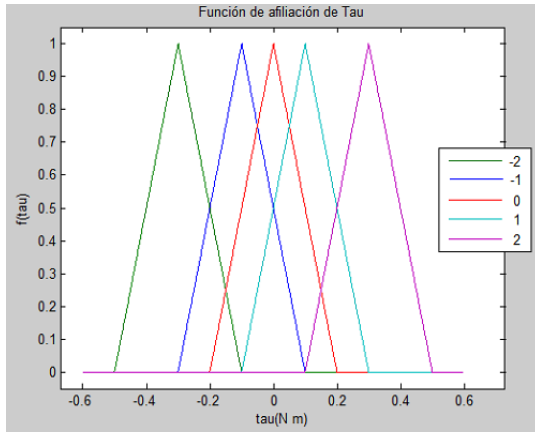
$$\xi_{PPN}(\tau) = \begin{cases} \max(0, 1 + \frac{\tau+0,1}{0,2}) & \tau \leq -0,1 \\ \max(0, 1 - \frac{0,1+\tau}{0,2}) & \tau > -0,1 \end{cases} \quad (42)$$

$$\xi_{PMP}(\tau) = \begin{cases} \max(0, 1 + \frac{\tau}{0,2}) & \tau \leq 0 \\ \max(0, 1 - \frac{\tau}{0,2}) & \tau > 0 \end{cases} \quad (43)$$

$$\xi_{PPP}(\tau) = \begin{cases} \max(0, 1 + \frac{\tau-0,1}{0,2}) & \tau \leq 0,1 \\ \max(0, 1 + \frac{0,1-\tau}{0,2}) & \tau > 0,1 \end{cases} \quad (44)$$

$$\xi_{PGP}(\tau) = \begin{cases} \max(0, 1 + \frac{\tau-0,3}{0,2}) & \tau \leq 0,3 \\ \max(0, 1 + \frac{0,3-\tau}{0,3}) & \tau > 0,3 \end{cases} \quad (45)$$

Estas funciones de membresía fueron obtenidas con el estudio previo del comportamiento del sistema con el control LQR.

Figura 5: Funciones de membresía τ .


V-C. Fuzzificación

Con base en las reglas mostradas en Tabla 1 y las funciones de membresía enlistadas, se cuantifica el valor de cada premisa posible, formadas por las intersecciones de las funciones de membresía que contengan el punto de entrada (sea este e o de/dt), mediante (46). Nótese que, en la Figura 3 y Figura 4, la máxima cantidad de premisas posibles es cuatro, debido a las combinaciones de las funciones de membresía que contengan a e y de/dt dentro de su dominio.

$$\mu_{premise} = \min \left[\mu(e), \sigma \left(\frac{de}{dt} \right) \right] \quad (46)$$

Una vez determinadas las premisas que intervendrán en la conclusión, se asigna una función de membresía τ a cada una de ellas.

V-D. Defuzzificación

Para encontrar una salida del controlador que considere todas las conclusiones de las premisas formadas se utiliza el método centro de gravedad (COG) [9]. Siendo b_i el centro de la función de membresía ξ_i y μ_i la función de membresía de la premisa i , se tiene:

$$u_{conc} = \frac{\sum_i b_i \int \mu_i}{\sum_i \int \mu_i} \quad (47)$$

Ya que las funciones de membresía son triángulos equiláteros, se puede determinar $\int \mu_i$ con la siguiente expresión:

$$\int \mu_i = w \left(h - \frac{h^2}{2} \right) \quad (48)$$

Con w siendo la base y h la altura del triángulo.

VI. SWING-UP

En [4], [10], [11] se presenta el control por energía del *swing up*, que consiste en añadir energía al péndulo mediante movimientos oscilatorios del brazo. Como se menciona en [4], la principal ventaja de este método es que el sistema recibe una cantidad de energía determinada por la diferencia de energías del péndulo en la posición vertical ($q_2 = 0$) y la energía actual.

La ley de control que considera la energía actual del sistema y agrega energía en la dirección correcta es la siguiente:

$$u = \text{sat}(k_v(E - E_0)) \text{sign}(\dot{q}_2 \cos q_2) \quad (49)$$

E representa la energía actual del sistema, mientras que E_0 la energía en la posición deseada. La función sat limitará el par, resultado de la ecuación (49), al rango del motor que se utilice. La parte de la ecuación $\text{sign}(\dot{q}_2 \cos q_2)$ indica la posición y la dirección del movimiento de q_2 . El coeficiente k_v es para establecer qué tan rápido el péndulo alcanza la proximidad a la posición de equilibrio inestable.

VII. RESULTADOS

Los parámetros utilizados en la simulación son los siguientes:

$$m_1 = m_2 = 0,1 \text{ kg} \quad l_1 = l_2 = 0,2 \text{ m} \quad g = 9,81 \text{ m/s}^2$$

Además de un par de entrada máximo de 3 mNm.

Se utilizó en ambos casos una ley de control que conmuta el controlador Swing-Up para llevar el péndulo a una posición cercana al equilibrio y dentro de dicha zona los controladores LQR y difuso propuestos, para satisfacer el objetivo de control.

La simulación de la dinámica del sistema, considerando la ley de control LQR propuesta, se muestra en la Figura 6. Como puede observarse, el sistema llega al origen y con ello satisface el objetivo de control, en aproximadamente ocho segundos, con condiciones iniciales $q_2 = \pi$ y $q_1 = \dot{q}_1 = \dot{q}_2 = 0$, y conmutando a dicha ley cuando q_2 es menor que $3\pi/8$.

Después de tres oscilaciones del péndulo (q_2) este logra equilibrarse y permanecer vertical, mientras el brazo sigue en movimiento hasta que finalmente se detiene a los ocho segundos. Se puede apreciar que utilizando la ley de control LQR, el brazo se logra estabilizar antes de completar una revolución. La velocidad máxima alcanzada por el brazo es de $17,36 \text{ rad/s}$ y $-21,35 \text{ rad/s}$ por el péndulo.

Con la ley de control difuso (Figura 7) la estabilidad se logra a los 15 segundos aproximadamente considerando una conmutación a dicha ley cuando q_2 es menor a $\pi/7$. A diferencia de la ley anterior, el brazo (q_1) necesitó más de cuatro vueltas en el mismo sentido para llegar al equilibrio y lograr una velocidad nula en el mismo, sin embargo, el péndulo se estabilizó y se mantuvo en una posición y velocidad constantes a los 2.57 segundos. La velocidad máxima alcanzada por el brazo fue de $-14,86 \text{ rad/s}$ y $18,33 \text{ rad/s}$.

VIII. CONCLUSIÓN Y TRABAJO A FUTURO

Se logró diseñar un controlador LQR y uno difuso para el péndulo de Furuta. Los resultados muestran que ambos controladores pueden satisfacer el objetivo de control. Las simulaciones presentan algunas ventajas al utilizar el controlador difuso ya que los picos de velocidad son menores, y el tiempo en que se lleva el péndulo a cero es menor comparado con el controlador LQR.

Sin embargo, el controlador difuso alcanza el equilibrio en el brazo en un tiempo mayor, además de que su zona de operación es menor a la del controlador LQR.

Figura 6: Simulación de controlador LQR+Swing-Up.

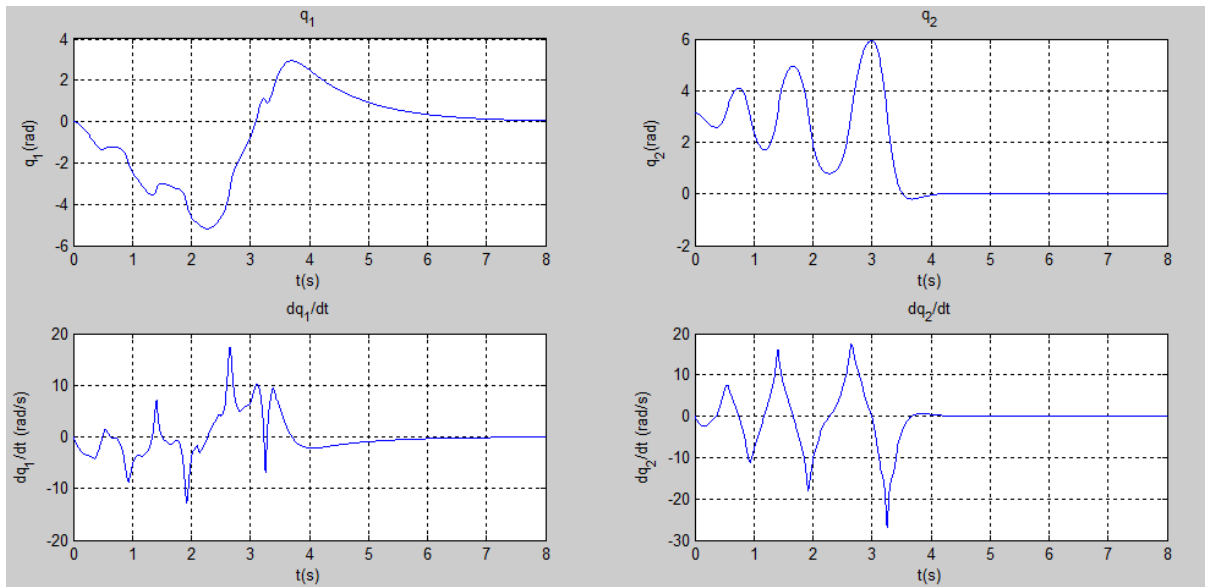
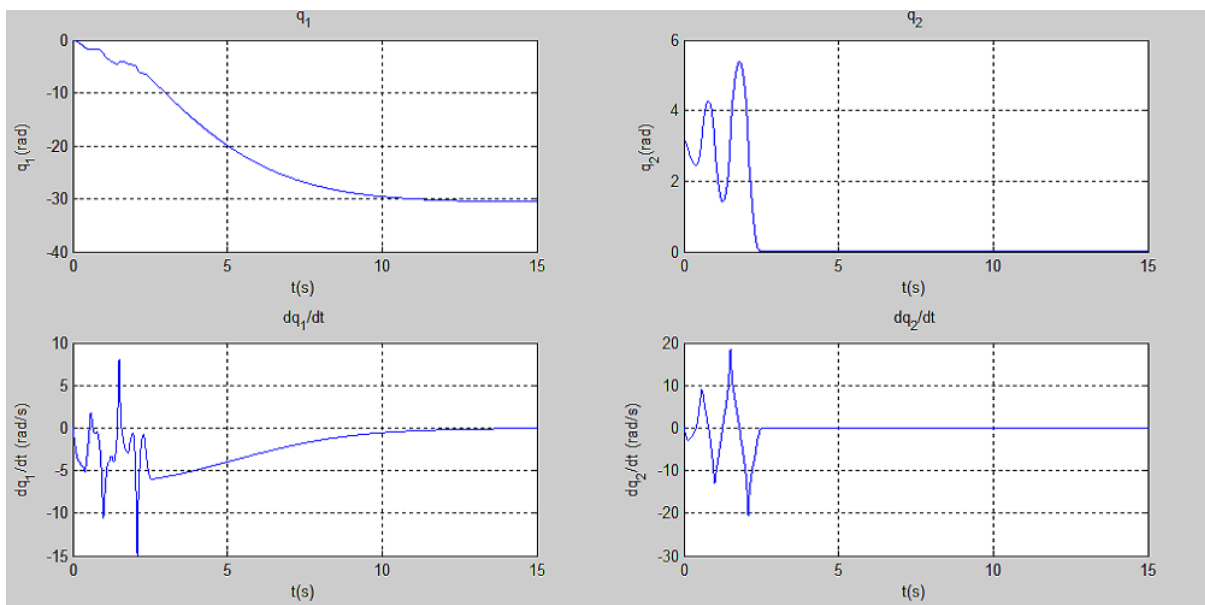


Figura 7: Simulación de controlador difuso+Swing-Up.



Como trabajo a futuro se implementarán las leyes de control propuestas a un péndulo de Furuta físico y se corroborará los resultados obtenidos por la simulación. Además, se podría desarrollar un controlador que utilice técnicas de control no lineal.

AGRADECIMIENTOS

Al Consejo Nacional de Ciencia y Tecnología (CONACyT), por la beca que me fue otorgada en el periodo Agosto 2017 a Julio 2018

REFERENCIAS

- [1] N. D. Quyen, N. V. Thuyen, N. Q. Hoc, and N. D. Hien, "Rotary inverted pendulum and control of rotary inverted pendulum by artificial neural network," *International Conference Theoretical Physics*, vol. 37, pp. 243–249, 2012. [Online]. Available: <http://www.iop.vast.ac.vn/theor/conferences/nctp/proc/37/243-249.pdf>
- [2] N. S. Bhangal, "Design and Performance of LQR and LQR based Fuzzy Controller for Double Inverted Pendulum System," *Journal of Image and Graphics*, vol. 1, no. 3, pp. 143–146, 2013.
- [3] O. D. M. Giraldo, J. G. V. Hernández, and D. G. Buitrago, "Control global del Péndulo Rotacional Invertido empleando modelos de energía," *Scientia et Technica*, vol. 1, no. 52, pp. 16–25, 2012. [Online]. Available: <http://200.21.217.140/index.php/revistaciencia/article/view/7831>

- [4] K. J. Åström and K. Furuta, "Swinging up a pendulum by energy control," *Automatica*, vol. 36, no. 2, pp. 287–295, 2000.
- [5] A. G. Moisés and A. Einstein, "Informática y de Telecomunicación Diseño y construcción de un robot auto-balanceado mediante Arduino Grado en Ingeniería Eléctrica y Electrónica Trabajo Fin de Grado," 2017.
- [6] R. Kelly and V. Santibáñez, *Control de movimiento de robots manipuladores*, 2003. [Online]. Available: <http://dialnet.unirioja.es/servlet/libro?codigo=176687>
- [7] K. Ogata, *Ingeniería de Control Moderna*, 2013, vol. 53, no. 9.
- [8] J. E. Machado, "Modelación y Control de un Robot Bípedo," 2014.
- [9] K. M. Passino and S. Yurkovich, *Fuzzy Control*, 1998, vol. 163.
- [10] J. G. Valenzuela H., O. D. Montoya ., and D. G. Buitrago, "Lógica Difusa Aplicada al Control Local del Péndulo Invertido con Rueda de Reacción," *Scientia et Technica*, vol. 18, no. 4, pp. 623–632, 2013.
- [11] P. Seman, B. Rohal'-Ilkiv, M. Juhás, and M. Salaj, "Swinging up the furuta pendulum and its stabilization via model predictive control," *Journal of Electrical Engineering*, vol. 64, no. 3, pp. 152–158, 2013.

Clasificador difuso para la señal de referencia de un controlador PI para regular el inventario en las cadenas de suministro

Pablo M. Ayllon-Lorenzo, Nohe R. Cazarez-Castro, Selene L. Cardenas-Maciel,
Jorge A. Lopez-Renteria

Tecnológico Nacional de México - Instituto Tecnológico de Tijuana - CONACYT,
Calz. Tecnológico, S/N, 22414, Tijuana, Baja California, México.
pablo.ayllon17@tectijuana.edu.mx, nohe@ieee.org, lilettecardenas@ieee.org,
jorge.lopez@tectijuana.edu.mx
<http://tectijuana.edu.mx/>

Abstract—En el presente trabajo se propone un clasificador difuso para la señal de referencia de un controlador PI para regular el inventario en las cadenas de suministro, estructurada en nodos productores y no productores. El clasificador descrito tiene como objetivo asignarle un modelo de inventario a los eslabones de la cadena de suministro atendiendo al impacto de la demanda del cliente final tanto en volumen como en importe. Como resultado de las diversas simulaciones realizadas se obtuvo una señal de referencia óptima que le permite al controlador proporcional integrador resolver el problema de regulación planteado. En esta investigación se considera una cadena de suministro como un conjunto instalaciones en una conexión en serie y/o en paralelo incluyendo centros de distribución y plantas de producción entre ellos.

I. INTRODUCCIÓN

La administración de una cadena moderna de suministro concierne conocimientos de manufactura, compras, distribución, teoría de control e inteligencia computacional. De ahí que la efectividad en la gestión de los inventarios constituya uno de los mayores retos de las cadenas de suministros en el siglo XXI. En una economía global cada vez más interdependiente e interconectada, el proceso de trasladar los suministros y los bienes terminados de un lugar a otro tiene lugar gracias a una apabullante innovación tecnológica, matemáticas avanzadas y softwares cada vez más potentes [1].

El hecho de que un sistema pueda seguir señales de referencia es uno de los problemas fundamentales en la teoría de control, por lo tanto, es un tema que se ha estudiado profundamente y para el que se han diseñado diferentes métodos que dan solución a este problema. Uno de estos métodos es la teoría de regulación, con la cual se puede lograr el seguimiento asintótico de trayectorias y al mismo tiempo rechazar perturbaciones [2].

Los sistemas de inferencia han sido utilizados como una estructura de representación del conocimiento impreciso o incierto, dichos sistemas están basados en los conceptos de conjuntos difusos, reglas lingüísticas de la forma Si-Entonces y en un sistema de razonamiento basado en la lógica difusa.

Los sistemas de inferencia difusa consisten en tres etapas: la fusificación, la base de reglas difusas y la defusificación [3].

La mezcla de estas áreas del conocimiento está encaminada a lograr el éxito en la gestión integrada de los procesos de la cadena, principalmente en la planificación de los inventarios.

La presente investigación combina elementos de control e inteligencia computacional con el fin de acotar el inventario en las cadenas de suministro, logrando optimizar los costos de mantenimiento de inventario e incrementando por ende la rentabilidad de las empresas de la red. Dicho informe está estructurado por 5 secciones, comenzando por el modelado matemático de la planta y la formulación del controlador hasta llegar a los resultados de la simulación, sin obviar el planteamiento de los elementos necesarios para desarrollar el clasificador difuso.

II. MODELADO MATEMÁTICO Y PLANTEAMIENTO DEL OBJETIVO DE CONTROL

En esta sección se detalla el modelo matemático que describe la dinámica de la planta y se exponen los elementos considerados para la implementación del controlador proporcional integrador.

A. Modelado de la Planta

De acuerdo al modelado de la planta, la tasa de cambio del inventario en todos los nodos de la cadena de suministro está determinada por la siguiente ecuación de balance de materia:

$$\frac{dN_i}{dt} = \lambda_i - \lambda_{i+1} \quad (1)$$

donde N_i es el nivel de inventario del nodo i , λ_i es la tasa de entrada al nodo i , λ_{i+1} es la tasa de salida del nodo i , todos estos parámetros $\in \mathbb{R}^+$ y en este artículo están dadas en unidades físicas; y siguen la secuencia mostrada en la Fig. 1.

Para la simplificación del análisis, son considerados nodos productores a aquellos eslabones que le agregan valor al producto comercializado (plantas de manufactura), mientras que

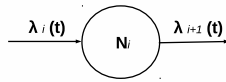


Fig. 1. Secuencia del Flujo Material

el resto de los elementos serán considerados no productores. Los nodos productores incluyen tienen implícitos otras variables que determinan su comportamiento y son modeladas según la siguiente ecuación:

$$\frac{d\lambda_i}{dt} = \frac{1}{T} [W_i(t) - \lambda_i(t)] \quad (2)$$

donde T es el tiempo de adaptación al cambio del nodo i expresado en unidades de tiempo, W_i es la política de producción del nodo i dado en unidades físicas [4].

B. Planteamiento del Objetivo de Control

El controlador usado en este trabajo contempla lo planteado en [5]. Donde la señal de control para los nodos no productores se rige por la siguiente función de saturación:

$$\lambda_i = \begin{cases} \lambda_{max} & \text{si } u(t) \geq \lambda_{max} \\ u(t) & \text{si } 0 < u(t) < \lambda_{max} \\ 0 & \text{si } u(t) \leq 0 \end{cases} \quad (3)$$

cuya ley de control está dada por:

$$u(t) = -k_p(N_i - N_{plan}) - k_i \int (N_i - N_{plan}) dt + \lambda_{i+1} \quad (4)$$

donde: λ_{max} es la tasa máxima de entrada del nodo i expresada en unidades físicas, N_{plan} es el nivel de inventario deseado expresado en unidades físicas, $u(t)$ es la señal del controlador PI asociada al control saturado, k_p y k_i son las ganancias proporcional e integradora respectivamente, λ_{max} , N_{plan} , k_p y $k_i \in \mathbb{R}^+$ y $u(t) \in \mathbb{R}$.

El sistema está en lazo cerrado cuando $0 < u(t) < \lambda_{max}$ y reemplazando (1), (3) y (4) se obtiene:

$$\dot{N}_i = -k_p(N_i - N_{plan}) - k_i \int (N_i - N_{plan}) dt. \quad (5)$$

la señal de control para los nodos productores se rige por la siguiente función de saturación:

$$W_i = \begin{cases} \lambda_{max} & \text{si } u(t) \geq \lambda_{max} \\ u(t) & \text{si } 0 < u(t) < \lambda_{max} \\ 0 & \text{si } u(t) \leq 0 \end{cases} \quad (6)$$

cuya ley de control está dada por:

$$u(t) = -k_p(N_i - N_{plan}) - k_p \int (N_i - N_{plan}) dt + \lambda_{i+1} \quad (7)$$

donde: λ_{max} es la tasa máxima de entrada del nodo i expresada en unidades físicas, N_{plan} es el nivel de inventario deseado

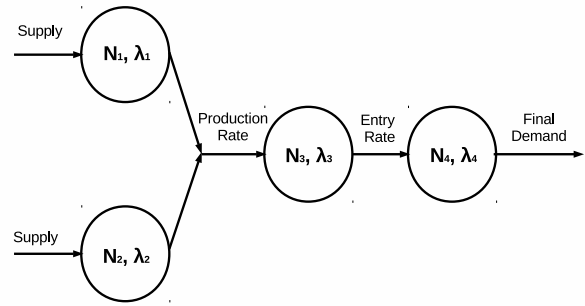


Fig. 2. Arquitectura de la Cadena de Suministro

expresado en unidades físicas, $u(t)$ es la señal del controlador PI asociada al control saturado, k_p y k_i son las ganancias proporcional e integradora respectivamente, λ_{max} , N_{plan} , k_p y $k_i \in \mathbb{R}^+$ y $u(t) \in \mathbb{R}$.

El sistema está en lazo cerrado cuando $0 < u(t) < \lambda_{max}$ y reemplazando (1), (2) y (7) se obtiene:

$$\dot{\lambda}_i = \frac{1}{T} (-k_p(N_i - N_{plan}) - k_i \int (N_i - N_{plan}) dt - \lambda_i + \lambda_{i+1}) \quad (8)$$

Siendo el objetivo de control planteado:

$$\lim_{t \rightarrow \infty} |N_i - N_{plan}| = 0 \quad (9)$$

III. METODOLOGÍA PARA EL DISEÑO DEL CLASIFICADOR DIFUSO

En esta sección se establecen las principales condiciones para el diseño del clasificador propuesto. El mismo tiene como objetivo asignarle un modelo de gestión de inventario a cada uno de los eslabones de la cadena de suministro (N_{plan}), con base en la relevancia que tiene la demanda del cliente final de la cadena en la rentabilidad de las empresas que la integran. Para ello se empleó la metodología planteada en [6].

Se tomó como referencia un segmento de una cadena de suministro de filtros de aceite automotrices compuesta por dos centros de distribución primeramente, seguidos por una planta ensambladora hasta llegar al punto de venta final de la red. La Fig. 2 ilustra la arquitectura de la cadena analizada.

Se consideró para este caso que una demanda diaria de este producto superior a las 200 unidades es significativamente alta para la cadena. De estar comprendida entre las 100 y 200 unidades su impacto es mediano en los niveles de ventas, mientras que de ser inferior a las 100 unidades se contempla como baja. Suponiendo que este producto tiene un precio de venta final de 264 MX, el impacto monetario de la demanda para la cadena será de alto, medio y bajo para aquellos importes mayores que 39,600 MX, los comprendidos

TABLE I
CLASIFICACIÓN DE LA DEMANDA SEGÚN EL VOLUMEN REQUERIDO

| Demandada Diaria (UF) | Clase |
|-----------------------|-------|
| $D \geq 200$ | Alta |
| $100 \leq D < 200$ | Media |
| $100 \geq D$ | Baja |

TABLE II
CLASIFICACIÓN DE LA DEMANDA SEGÚN EL IMPORTE REQUERIDO

| Importe Diario (MX) | Clase |
|--------------------------|-------|
| $I \geq 39,600$ | Alta |
| $39,600 \leq I < 19,800$ | Media |
| $19,800 \geq I$ | Baja |

entre 39,600 MX y 19,800 MX y los menores de 19,800 MX respectivamente. De igual forma se contempló que la demanda máxima que puede asimilar la cadena atendiendo a sus capacidades es de 300 filtros diarios. Las tablas I y II muestran detalladamente estos criterios.

El clasificador está estructurado por un sistema difuso tipo Takagi-Sugeno, para separar la demanda final atendiendo a los criterios expuestos anteriormente y asignarle un modelo de gestión de inventario a los nodos de la cadena, atendiendo a la relación entre los valores de entradas.

A. Modelado Difuso T-S

El diseño del sistema difuso es similar para las dos variables, la variación está determinada por la diferencia en el soporte de las funciones de membresías seleccionadas. Claramente, un análisis de la demanda en volumen tiene un impacto directo en el balance de las capacidades de almacenamiento, mientras que un análisis del importe diario solicitado tiene una connotación financiera.

1) *Fusificación*: Fueron diseñadas funciones de membresía triangulares para este proceso tal y como se muestra en las Figs. 3 y 4.

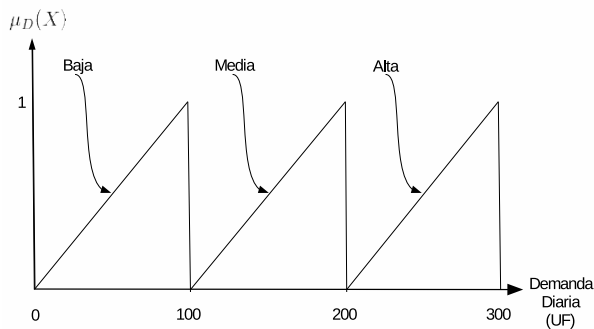


Fig. 3. Función de membresía para la demanda en volumen

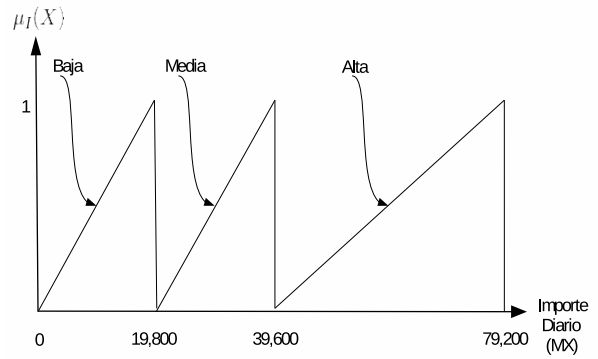


Fig. 4. Función de membresía para la demanda en importe

Es importante destacar que como el la señal de referencia del controlador se expresa en unidades físicas, todos los modelos de gestión estarán condicionados al volumen solicitado por el cliente final. Sin embargo, la entrada monetaria juega un papel imprescindible a la hora de formular las reglas de inferencia.

2) *Formato de las Reglas*: Se propusieron tres tipos de sistemas de gestión de inventario atendiendo a la combinación de los valores de las entradas.

1) Lote Óptimo Básico (EOQ por sus siglas en inglés):

$$N_{plan} = \sqrt{\frac{2 \times C_l \times D}{C_p}} \tag{10}$$

donde:

N_{plan} es el tamaño del Lote Óptimo expresado en unidades físicas y sería la señal de referencia del controlador,

C_l el el coste de lanzar un pedido expresado en unidades monetarias,

D es la demanda anual del producto expresada en unidades físicas/año,

C_p es el costo de mantenimiento de inventario expresado en unidades monetarias/unidad al año [7].

Al trabajar con la demanda diaria, se modifica la ecuación (10):

$$N_{plan} = \sqrt{\frac{730 \times C_l \times D}{C_p}} \tag{11}$$

2) Método Doble Lote:

$$N_{plan} = 2 \times L \times D \tag{12}$$

donde:

D es la demanda diaria del producto expresada en unidades físicas,

L es el ciclo de gestión del pedido expresado en unidades de tiempo.

TABLE III
PARÁMETROS DE LOS MODELOS DE GESTIÓN DE INVENTARIOS

| Variable | Valor |
|----------|---------|
| C_l | 50 MX |
| C_p | 15 MX |
| L | 10 días |

TABLE IV
MOTOR DE INFERENCIA

| Demanda(UF) | Importe(MX) | SGI | Modelo Sugeno |
|-------------|-------------|-------------|---------------------|
| Alta | Alta | Lote Óptimo | $z = 49.32 + 0 + 0$ |
| Media | Alta | Lote Óptimo | $z = 49.32 + 0 + 0$ |
| Baja | Alta | Doble Lote | $z = 20 + 0 + 0$ |
| Alta | Media | Lote Óptimo | $z = 49.32 + 0 + 0$ |
| Media | Media | Doble Lote | $z = 20 + 0 + 0$ |
| Baja | Media | Doble Lote | $z = 20 + 0 + 0$ |
| Alta | Baja | Doble Lote | $z = 20 + 0 + 0$ |
| Media | Baja | Lote a Lote | $z = 10 + 0 + 0$ |
| Baja | Baja | Lote a Lote | $z = 10 + 0 + 0$ |

3) Método Lote a Lote:

$$N_{plan} = L \times D \tag{13}$$

donde:

D es la demanda diaria del producto expresada en unidades físicas,

L es el ciclo de gestión del pedido expresado en unidades de tiempo [8].

Los elementos de los modelos de gestión requeridos para el desarrollo de las simulaciones se muestra en la tabla III

conforme a lo planteado en [9], se elaboraron las reglas de inferencia mostradas en la tabla IV

IV. RESULTADOS

Se consideró una demanda diaria de 217 filtros y se llevó a cabo la simulación en el toolbox Simulink de Matlab obteniéndose los siguientes resultados para la cadena de suministro propuesta:

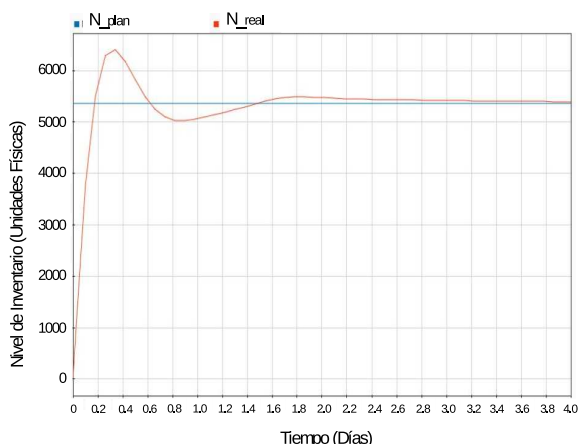


Fig. 5. Dinámica del nivel de inventario en los nodos 1 y 2

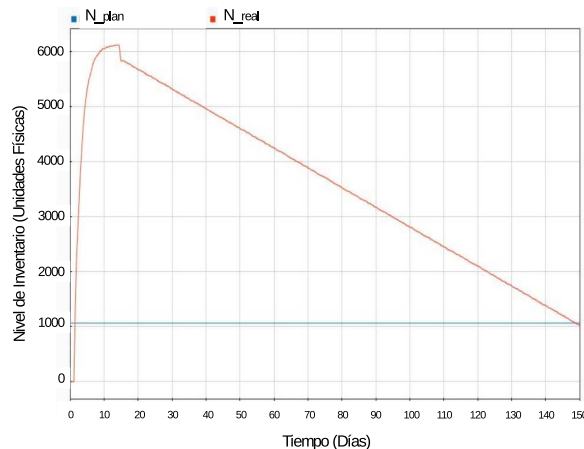


Fig. 6. Dinámica del nivel de inventario en el nodo 3

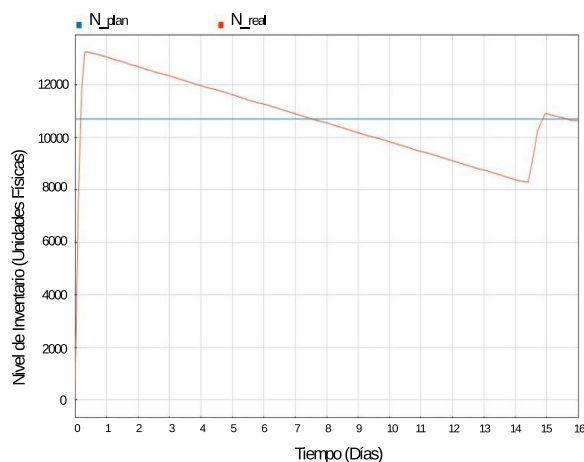


Fig. 7. Dinámica del nivel de inventario en el nodo 4

En la Fig.5 se ilustra el comportamiento de la señal de control para los nodos 1 y 2. El nivel de inventario para estos eslabones de la cadena tiene un estado transitorio relativamente corto pues logra alcanzar el nivel deseado (el resultante del clasificador) en un periodo inferior a los 4 días.

El nodo productor al depender del tiempo de adaptación al cambio para ajustar su política de producción, demora un tiempo considerablemente largo (superior a los 145 días) en alcanzar el estado estable, tal y como se muestra en la Fig. 6.

El nodo final de la cadena a pesar de no tener incorporada ninguna línea de producción, es el que tiene mayor responsabilidad ante el cliente final y por lo tanto debe ofrecer un tiempo de respuesta corto ante la variación de la demanda, para no perder clientes de su segmento de mercado. Según se muestra en la Fig. 7 el controlador propuesto garantiza una dinámica positiva para este eslabón de cara al mercado destino.

V. CONCLUSIONES

- 1) Se propuso un controlador capaz de resolver el problema de regulación del inventario en las cadenas de suministro.
- 2) Se diseñó un clasificador difuso para asignarle un modelo de gestión de inventario a los diferentes nodos de la cadena de suministro con base en el impacto de la demanda final de la misma.
- 3) Se asignaron valores aleatorios de demanda y se simuló para evaluar la viabilidad del clasificador elaborado, resultando positivos los resultados alcanzados.
- 4) Con la metodología propuesta se permite optimizar los costos de manejo de inventario en las cadenas de suministros de bienes y servicios al controlar los volúmenes de mercancía en sus almacenes.

ACKNOWLEDGMENT

Este proyecto fue parcialmente financiado por el Tecnológico Nacional de México con el proyecto 6351.17-P.

REFERENCES

- [1] Chase, Richard B., and F. Robert Jacobs. Administración de operaciones: producción y cadena de suministros. McGraw Hill México, 2010.
- [2] Martínez, Rosalío Farfan, et al. "Control Difuso Takagi-Sugeno Aplicado al Equipo Didáctico de la Bola y la Viga."
- [3] S. C. Ponce Flores, P. M. Quintero Flores and J. L. Hernández Corona. "Modelado de un sistema de inferencia difusa tipo Mamdani para el diagnóstico de envejecimiento de motores eléctricos".International Journal of Innovation and Scientific Research, pp. 34-42, 2016.
- [4] T. Nagatani, D. Helbing and A. Rodríguez-Angeles. "Stability Analysis and Strategies for linear Supply Chains". Institute for Economics and Traffic, Dresden University of Technology, 2004.
- [5] D. A. Trejo Rodríguez, A. Rodríguez Angeles. "Control acotado para la regulación de inventario en cadenas de suministro". Memorias del Congreso Nacional de Control Automático, Queretaro, 2016.
- [6] I. Guadalupe Gaytan, S. Cardenas Maciel and N. Cazarez Castro, *Clasificador de Agua Tratada Local*, Ensenada, Mexico: Memorias del Congreso de Ciencias Computacionales, CiCOMp, 2016.
- [7] Juan, Ángel A. and Rafael Garcé -Martín. "Gestión de stocks: modelos deterministas." Recuperado 4.17 (2013): 2015.
- [8] Ballou, Ronald H. Logística: Administración de la cadena de suministro. Pearson Educación, 2004.
- [9] J. S. R. Jang, C. T. Sun, E. Mizutani, Neuro-Fuzzy and Soft Computing , Prentice-Hall, 1997.

Reporte Lingüístico Automático para el Proceso de Compostaje

Andrea M. de Anda ^{*}, Marco A. Castro L. ^{*}, Iliana Castro L. ^{*}, Joaquín Gutiérrez J. [†]

^{*}Tecnológico Nacional de México

Instituto Tecnológico de La Paz, La Paz, B.C.S. B

Email: deandaMSC16@itlp.edu.mx, mcastro@itlp.edu.mx, icaastro@itlp.edu.mx

[†]Centro de Investigaciones Biológicas del Noroeste, La Paz, B.C.S.

Email: joaquin04@cibnor.mx

Resumen—Mediante el uso de el lenguaje natural se pueden generar reportes entendibles y prácticos para los humanos acerca de algún fenómeno o proceso. En el trabajo presente se implementó el Modelo Granular Lingüístico de un Fenómeno para desarrollar una aplicación que genera reportes lingüísticos del proceso de compostaje el cual requiere de una constante supervisión para producir una composta de calidad.

I. INTRODUCCIÓN

La tecnología actual permite el almacenamiento automatizado de datos acerca de un fenómeno, los cuales por lo general son numéricos. Para que estos datos sean útiles necesitan ser representados de forma entendible. Una alternativa es el uso del Lenguaje Natural (LN) que permite una descripción sucinta del estado actual, tendencia o comportamiento de un fenómeno.

El LN permite a los humanos hacer representaciones imprecisas pero prácticas acerca de un fenómeno complejo ya que permite la granularidad al ocultar ciertos detalles de poca importancia o resaltar otros que si lo sean[1].

Lo anterior motivó a Lotfi Zadeh para desarrollar la Teoría Computacional de Percepciones (TCP) [2] que permite razonar con percepciones en lugar de hacerlo con medidas numéricas exactas haciendo uso de la Lógica Difusa (LD) para tal propósito. La LD maneja de manera eficiente la imprecisión y permite la generación de descripciones en LN [3].

Gracian Triviño y sus colaboradores tomaron como base la TCP para desarrollar el Modelo Granular Lingüístico de un Fenómeno (MGLF) [4] el cual es una metodología que permite desarrollar una aplicación que genere una descripción lingüística en tiempo real del estado y evolución de un fenómeno.

Algunas de sus aplicaciones incluyen el análisis del tráfico [5][6], el análisis financiero[7], el análisis de estructuras de la superficie marciana[8], en el aprendizaje [9] y en la evaluación de diversas actividades humanas [10][11].

En este trabajo se aplica el MGLF para generar un reporte lingüístico del proceso de elaboración de composta o proceso de compostaje.

En las instalaciones del Centro de Investigaciones Biológicas del Noroeste (CIBNOR) actualmente se trabaja en la elaboración de composta mediante diversas metodologías.

Sensores recolectan los datos numéricos de la temperatura de un montículo de composta.

El presentar la información recolectada como una lista de números no es fácil de entender por cualquier persona. Con los datos recopilados se puede inferir información del estado de la composta, por ejemplo si el proceso ha sido estable y si se ha comportado de acuerdo a lo esperado. Toda esta información se puede presentar en forma de mensajes en LN para ser mejor entendidos.

II. MODELO GRANULAR LINGÜÍSTICO DE UN FENÓMENO

Un concepto relacionado a la TCP es la granulación. La granulación permite agrupar los datos de entrada en unidades de información llamados gránulos. Un gránulo es una colección de objetos, ya sean físicos o abstractos que se mantienen agrupados por diversos criterios como similitud, proximidad, indistinción o funcionalidad [12]. La granulación puede ser rígida o difusa. Por ejemplo, la edad puede ser un gránulo rígido si se habla de la edad en años. Sin embargo será un gránulo difuso si se habla de la edad como joven, mediana edad o viejo. La granulación difusa es más parecida a la manera en como el humano razona y toma decisiones en ambientes que son imprecisos o donde hay incertidumbre y la verdad es parcial [13].

La granulación difusa es una manera de agrupar información en clases o conjuntos difusos. Esta manera de agrupar información juega un papel importante en algunos procesos mentales como el reconocimiento de patrones, la comunicación y la abstracción [14].

Un conjunto difuso tiene una etiqueta llamada variable lingüística. Las variables lingüísticas son aquellas cuyos valores son palabras o sentencia en LN [15]. Por ejemplo, Temperatura es una variable lingüística y sus valores podrían ser baja, mediana o alta. La granulación de conceptos es dependiente del contexto o del fenómeno en cuestión.

II-A. Percepción Computacional

Una Percepción Computacional (PC) es un modelo de una unidad de información o gránulo del fenómeno que consta de una tupla (\bar{A}, \bar{W}) donde:

- $\vec{A} = (a_1, a_2, \dots, a_n)$ es un vector de expresiones lingüísticas en LN y pueden ser simples como por ejemplo $a_i = \text{"La temperatura es alta"}$, o compleja como por ejemplo $a_i = \text{"En los últimos tres días el estado de la composta ha sido estable"}$.
- $\vec{W} = (w_1, w_2, \dots, w_n)$ Es un vector con los grados de validación $w_i \in [0, 1]$ asociados a cada a_i . Estos valores dependen de cada aplicación y del contexto.

Cada PC representa un punto de vista particular del fenómeno ya que cubren aspectos específicos con cierto grado de detalle o granularidad.

II-B. Mapeo de Percepciones

Cada Mapeo de Percepciones (MP) es usado para crear y agregar una PC. Un MP es una tupla que consta de cuatro componentes (\vec{U}, y, g, T) donde:

- $\vec{U} = (u_1, u_2, \dots, u_n)$ es el vector de n PC de entradas con cada $u_i = (\vec{A}_{ui}, \vec{W}_{ui})$. Se le llama mapeo de percepciones de primer orden ($1 - MP$), cuando se mapea las entradas directamente de una base de datos, de una imagen o de sensores.
- $y = (\vec{A}_y, \vec{W}_y)$ es la salida PC.
- $g()$ es una función de agregación que se utiliza para generar el vector de grados difusos $\vec{W}_y = (W_1, W_2, \dots, W_{ny})$ de las entradas PC. Las funciones que se utilizan son funciones de membresía de la LD.
- T es el algoritmo generador del texto que permite generar las sentencias en A_y . T puede ser una plantilla lingüística, por ejemplo, $\text{"La temperatura es \{baja / media / alta\}"}$.

II-C. Modelo Granular Lingüístico del Fenómeno

El Modelo Granular Lingüístico de un Fenómeno presenta la información de forma granulada. A través de una red de MP, la información se va presentando dependiendo del detalle que se requiera. Una entrada (PC) a un nivel, es la salida de un nivel anterior y se puede tener tantos niveles como sea necesario. La entrada z del primer nivel $1 - MP$ es directa de una fuente externa, ya sea un sensor, una imagen o una base de datos.

La figura 1 muestra la arquitectura propuesta para el sistema de reporte automático del proceso de compostaje basada en la arquitectura presentada por Triviño. Los módulos principales son el módulo de adquisición de datos, el módulo de validación y el módulo de generación de reportes.

II-C1. Módulo de Adquisición de Datos: Este módulo se encarga de capturar los datos directos del sistema físico o fenómeno. Estos provienen de una fuente externa y puede ser de sensores, de internet, de una base de datos o de una imagen. Estos datos son las entradas para los mapeos de primer nivel (1-PM) con los que se producen los primeros PC.

II-C2. Módulo de Validación: En este módulo se hace uso del MGLF propuesto para el fenómeno. Cuando llega una entrada, esta es validada usando la función de agregación para calcular su grado de pertenencia a cada PC. Las salidas de este

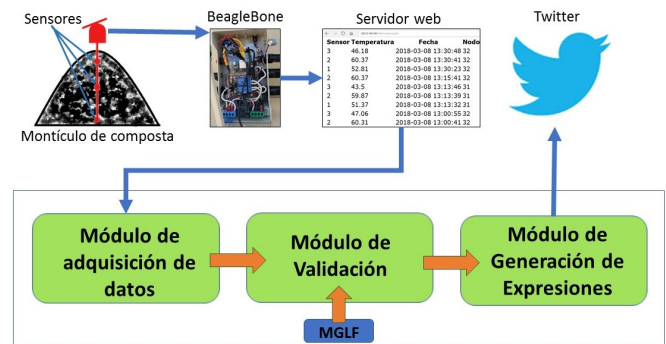


Figura 1. Arquitectura propuesta del MGLF para el proceso de compostaje

módulo son sentencias lingüísticas con sus respectivos grados de validación.

II-C3. Módulo de Generación de Expresiones: En el módulo de generación de reportes se combinan todos los PC de salida para generar un reporte lingüístico por medio de plantillas.

III. EL PROCESO DE COMPOSTAJE

En la actividad agrícola cada vez es más frecuente el uso de composta como abono orgánico para obtener una buena calidad en las cosechas y un menor impacto en el medio ambiente, sobre todo en zonas donde las condiciones del suelo no son tan favorables para la agricultura.

La composta es el producto final de los procesos bioquímicos que sufre la basura orgánica al descomponerse y se reincorpora al ciclo natural que sigue toda materia orgánica por lo que no representa un daño al ambiente[16]. Su costo de producción es bajo y como se sabe la basura representa un gran problema en la actualidad así que un beneficio adicional es el manejo de la misma.

Durante el proceso se generan cambios físicos y bioquímicos y en el intervienen varios factores como la humedad, la aereación, el pH, la relación nitrógeno-carbono, el tamaño de los desechos, el tiempo y la temperatura, siendo este último el factor que más cambia durante todo el proceso por lo tanto es importante conocerlo en todo momento. Su estado en una determinada etapa puede ser indicador de que el proceso va bien o mal.

Como se ve en la figura 2 en el proceso de compostaje se reconocen tres etapas principales en base a los cambios de temperatura, más una etapa de maduración de tiempo variable [17]. Las tres etapas principales son:

1. Mesófila
2. Termófila
3. Enfriamiento

La etapa Mesófila tiene una duración de entre 2 y 5 días con una temperatura que va desde la temperatura ambiente hasta los 40 o 50°C.

La etapa Termófila puede durar entre 1 y 3 semanas dependiendo de los materiales usados y las condiciones del clima entre otros factores. Aquí se alcanzan la temperatura

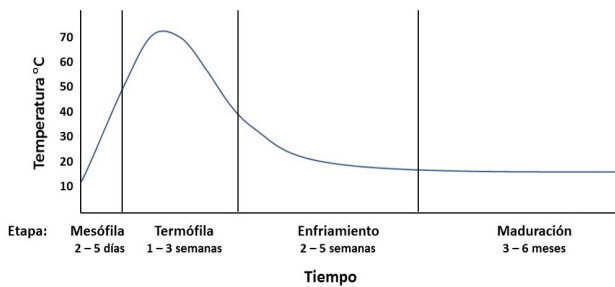


Figura 2. Etapas del proceso de compostaje.

más alta que se presentará durante todo el proceso llegando a estar entre los 50 y 70°C. Esta temperatura es la consecuencia de la actividad microbiana y se alcanza entre los primeros 3 y 7 días de iniciado el proceso. Sin embargo una temperatura mayor a los 70°C por más de 3 días puede matar las bacterias necesarias para esta etapa.

La etapa de enfriamiento requiere de varias semanas de duración, entre 2 y 5 semanas. Los microorganismos presentes en esta etapa no elevan tanto la temperatura de hecho la temperatura va bajando hasta llegar a igualar la del ambiente.

La etapa de maduración puede durar meses y la temperatura presente es la del medio ambiente.

El tiempo es otro factor importante. Para que una composta pueda ser utilizada dependerá del manejo de todos los factores y esto determinará el tiempo y la calidad de la composta. Si un proceso de compostaje tiene un adecuado manejo, la composta estará lista en un tiempo mínimo de 3 meses en sitios con pocas lluvias y de climas cálidos.

IV. DESCRIPCIÓN LINGÜÍSTICA DEL PROCESO DE COMPOSTAJE

IV-1. Módulo de Adquisición de Datos: Una de las dos entradas a este módulo es la temperatura (T). Ésta se lee de un servidor web que a su vez proviene de los tres sensores instalados en el nodo, uno en la parte inferior, otro en el centro y otro más en la parte superior. Este nodo se instala en el montículo de composta desde que se inicia su elaboración. La medición se hace cada 15 o 30 minutos. De los datos medidos el más significativo es el que lee el sensor del centro ya que en el centro del montículo se mantienen las temperaturas más estables. Con el paso del tiempo el montículo va reduciendo su tamaño y el sensor que está en la parte superior del nodo queda expuesto al ambiente. El sensor de la parte inferior está cerca del suelo por lo que las temperaturas medidas serán menores.

La segunda entrada es el tiempo (t) de medición de la temperatura. Esto indica en qué etapa se encuentra el proceso.

IV-2. Módulo de Validación: El MGLF propuesto para el proceso de compostaje se muestra en la figura 3.

Las particiones difusas de temperatura son cinco: muy_baja, baja, media, alta y muy_alta. Las particiones de tiempo son las tres principales del proceso de compostaje: mesófila, termófila

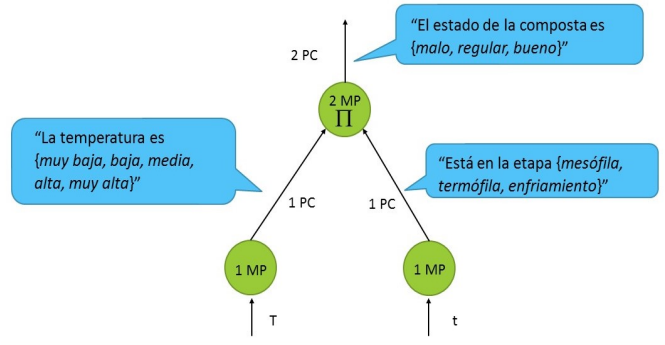


Figura 3. MLGF propuesto para el proceso de compostaje

y enfriamiento (figura 4). Las particiones de estado son tres: malo, regular, bueno (figura 5).

El conjunto de reglas son del tipo:

“IF temperatura IS {muy_baja/baja/media/alta/muy_alta} AND tiempo IS {mesofila/termofila/enfriamiento} THEN estado IS {malo/regular/bueno}”.

IV-3. Módulo de Generación de Expresiones: El reporte lingüístico generado por el sistema puede ayudar a los encargados de vigilar el proceso de compostaje a no estar presentes en el lugar de manera tan constante. Los reportes generados son del tipo específicos que se generan al momento de hacer la medición de la temperatura.

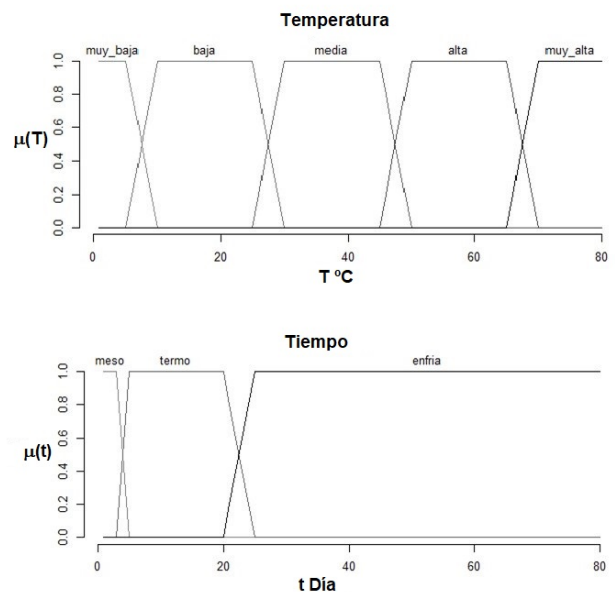


Figura 4. Función de membresía y las etiquetas lingüísticas para temperatura (T) y tiempo (t) que son las entradas a 1-MP.

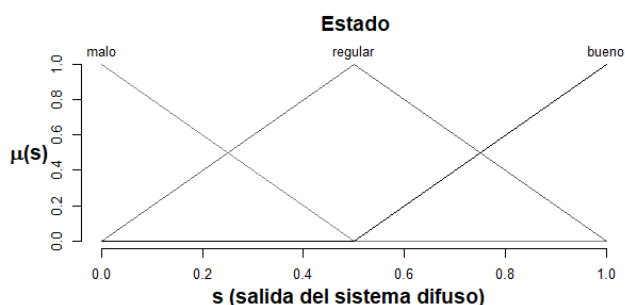


Figura 5. Particiones del estado de la composta.

V. PRUEBAS Y RESULTADOS

El sistema se diseñó tomando como entrada los datos que se recolectaron de una composta que se elaboró en junio del 2017 en el CIBNOR y se programó en el lenguaje R. Fue necesario hacer ajustes en las particiones de temperatura y tiempo.

Al momento de escribir este artículo el sistema se encuentra en marcha recolectando, analizando y almacenando los datos de una composta en tiempo real con varias mediciones durante el día para una posterior evaluación del sistema.

La figura 6 muestra el montículo de composta que se está utilizando para pruebas del sistema. En él se instalaron dos nodos con tres sensores cada uno.

La figura 7 muestra un mensaje generado por el sistema usando la aplicación Twitter como medio de visualización de los reportes. La temperatura recolectada fué de 60.87°C en el día 22 de iniciado el proceso. La temperatura es alta y la etapa es termófila. El reporte generado es “El estado de la composta es bueno”, según lo validado por el MGLF.

La figura 8 muestra el mensaje generado después de dar las entradas 70°C para la temperatura en el tercer día de iniciado el proceso. La temperatura es alta y la etapa es mesófila, así la salida es “El estado de la composta es malo”.



Figura 6. Montículo de composta con dos nodos de sensores en las instalaciones del CIBNOR.



Figura 7. Mensaje generado por el sistema el día 22 del proceso con una temperatura de 60.87°C.



Figura 8. Mensaje generado por el sistema el día 3 del proceso con una temperatura de 70°C.

VI. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha presentado un sistema para generar automáticamente reportes sobre el estado de un proceso. El reporte es una oración en LN que se transmite mediante una App de mensajería, lo que convierte al sistema en una herramienta fiable, práctica y de fácil acceso, donde el usuario final pueda enterarse de las condiciones actuales del proceso y entonces tomar la acción correspondiente.

Los mensajes en LN son específicos en el instante. En un trabajo futuro se podrían generar mensajes que mostraran el comportamiento de la composta durante todo un período de tiempo, por ejemplo durante toda la mañana, durante toda la tarde o durante todo el día, incluso mostrar la tendencia.

Para generar un modelo granular más amplio se podría agregar al sistema otras entradas por ejemplo los datos de cualquier otro factor que intervenga en el proceso.

Se pretende que el trabajo actual sirva como base para el desarrollo de otras aplicaciones que generen una descripción lingüística de diversos cultivos experimentales que se realizan en el CIBNOR actualmente.

REFERENCIAS

- [1] Gracian Trivino, Michio Sugeno, *Toward linguistic description of a phenomena*, International Journal of Approximate Reasoning 54 (2013) 2234.
- [2] L. A. Zadeh, *From Computing with Numbers to Computing with Words-From Manipulation of Measurements to Manipulation of Perceptions*, Fundam. THEORY Appl. IEEE Trans. FUZZY Syst. IEEE Trans. CIRCUITS Syst. Fundam. THEORY Appl., vol. 45, no. 1, pp. 103111, 1999.
- [3] T. J. Ross, *Fuzzy Logic with Engineering Applications*, Thrid Edit. Wiley, 2010.
- [4] C. Martinez-Cruz, A. Van Der Heide, D. Sanchez, and G. Trivino, *An approximation to the computational theory of perceptions using ontologies*, Expert Syst. Appl., 2012.
- [5] Gracian Trivino, Angel Sanchez, Antonio S. Montemayor, Juan J. Pantrigo, Raul Cabido and Eduardo G. Pardo, *Linguistic description of traffic in a roundabout*, In Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ), Spain, 2010.
- [6] Alberto Alvarez-Alvarez, Daniel Sanchez-Valdes, Gracian Trivino, Angel Sanchez, Pedro D. Suarez, *Automatic linguistic report of traffic evolution in roads*, Expert Systems with Applications 39 (2012) 1129311302.
- [7] Mendez, *Combining semantic web technologies and computational theory of perceptions for text generation in financial analysis*, 2010.
- [8] Daniel Sanchez-Valdes, Alberto Alvarez-Alvarez, Gracian Trivino, *Automatic linguistic description about circular structures of the Mars surface*, Applied Soft Computing 13 (2013) 47384749.

- [9] M. Gloria Sanchez-Torrubia, Carmen Torres-Blanc, Gracian Trivino, *An approach to automatic learning assessment based on the computational theory of perceptions*, Expert Systems with Applications 39 (2012) 1217712191.
- [10] Daniel Sanchez-Valdes, *Periodical report about personal physical activities*, Spain, 2012.
- [11] Clemente Rubio-Manzano, Gracian Trivino *Improving player experience in Computer Games by using players' behavior analysis and linguistic descriptions*, Int. J. Human-Computer Studies 95(2016)2738.
- [12] Lotfi A. Zadeh, *Fuzzy sets and information granularity*, Advances in Fuzzy Set Theory and Applications, 1979.
- [13] Lotfi A. Zadeh, *Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic*, Fuzzy Sets and Systems, vol. 90, 1997.
- [14] Lotfi A. Zadeh, *Fuzzy sets*, 1965.
- [15] Lotfi A. Zadeh, *The concept of a linguistic variable and its application to approximate reasoning - I*, Information Sciences, 1975.
- [16] A. Nieto Garibay y otros, *La composta. Importancia, elaboración y uso agrícola*, Trillas, 2013.
- [17] Pilar Román, María M. Martínez, Alberto Pantoja, *Farmers Compost Handbook. Experiences in Latin America*, Food and Agriculture Organization of the United Nations Regional Office for Latin America and the Caribbean Santiago, 2015.

Sistema de clasificación de documentos basado en categorías de Wikipedia

Alan Ramírez-Noriega, José Mendivil-Torres, Jorge Luis Almaral-Sánchez,
Yobani Martínez-Ramírez, J. Francisco Figueroa Pérez
Facultad de Ingeniería Mochis, Universidad Autónoma de Sinaloa
Fuente de Poseidón y Ángel Flores s/n, Col. Jiquilpan. Los Mochis, Sinaloa, México
{alandramireznoriega,jose.mendivil,jalmaral,yobani.juanfco.figueroa}@uas.edu.mx

Resumen—La información en ésta época ha ganado mucha importancia para las personas y para las empresas, sin embargo, el exceso de información puede convertirse en un problema cuando no se organiza de una forma adecuada. Es común descargar documentos de Internet y tenerlos en distintos lugares de la computadora, al cabo de un tiempo hay muchos archivos desorganizados. La propuesta de esta investigación emplea las categorías y documentos de Wikipedia para clasificar documentos de texto en las mismas categorías que la enciclopedia provee. Se hizo un experimento para encontrar al mejor clasificador de documentos bajo las condiciones de Wikipedia, el mejor rendimiento lo tuvo la máquina de soporte vectorial. Después, en otro experimento se clasificaron documentos con el sistema prototipo desarrollado obteniendo un rendimiento de 0.84. Este trabajo establece que los artículos de Wikipedia son una fuente para clasificar documentos con un rendimiento confiable.

I. INTRODUCCIÓN

El conocimiento es un recurso de importancia estratégica para las organizaciones; su generación, codificación, gestión y divulgación aportan al proceso de innovación. Todos estos aspectos se incluyen en lo que se llama gestión del conocimiento. La cantidad de documentos de diversos tipos disponibles en una organización es enorme y continúa creciendo cada día. Estos documentos, más que las bases de datos, son a menudo un repositorio fundamental del conocimiento [1], [2]. La humanidad produce tanta información que su catalogación manual ya no es posible; esto obliga al desarrollo de herramientas automatizadas, apoyando a las personas en el procesamiento de la información [3].

La tarea de clasificación automática de texto consiste en utilizar documentos de texto con clases o categorías predefinidas para organizar nuevos documentos en estas mismas categorías [4], [5], [2]. Esta actividad ha acumulado un estatus importante en el campo de los sistemas de información debido a la mayor disponibilidad de documentos en formato digital, por otra parte, es necesario acceder a ellos de manera fácil [5].

Un típico problema de clasificación puede ser declarado como sigue: dado un conjunto de ejemplos etiquetados pertenecientes a dos o más clases (datos de entrenamiento), clasificamos una nueva muestra de prueba a una clase con la mayor similitud. Los sistemas de recuperación de documentos, enrutamiento y filtrado, a menudo pueden ser vistos como un problema de clasificación de dos clases donde un

documento es etiquetado como relevante o no relevante [4]. En la actualidad, la categorización de texto se aplica en muchos contextos, desde la indexación de documentos dependiendo de un vocabulario hasta el filtrado de documentos, creación automática de metadatos, vaguedad en el sentido de la palabra, entre otras aplicaciones [5].

La clasificación automática de documentos ayuda a las organizaciones a deshacerse de la clasificación manual pudiendo ser costosa y consumir mucho tiempo. La precisión de los modernos sistemas de clasificación de textos se ha convertido en un competidor para personas capacitadas profesionalmente. Estos sistemas son el resultado de una combinación de tecnologías de recuperación de información y tecnologías de aprendizaje automático [2].

La categorización de documentos necesita fuentes fiables de información para basar su aprendizaje. Un repositorio importante de información es la enciclopedia en línea Wikipedia¹. Esta enciclopedia se ha convertido en el mayor repositorio de conocimientos electrónicos de la web con millones de artículos aportados en colaboración por voluntarios [6], siendo un rica fuente de información lingüística para la categorización de documentos y otras actividades relacionadas con el procesamiento del lenguaje natural[7].

Los recursos de Wikipedia pueden ser aplicados a distintos problemas de clasificación, sin embargo, la misma Wikipedia puede utilizar su conocimiento para clasificar futuras redacciones de artículos. El proceso de clasificación del contenido de Wikipedia es realizado por los editores del artículo. Un editor, que modifica un artículo, indica manualmente a qué categoría debe asignarse el artículo. Esa tarea requiere un conocimiento de la estructura de Wikipedia y su sistema de categorías, pero que con frecuencia está más allá de la competencia típica del editor. La falta de este conocimiento conduce a los errores causados por el hombre omitiendo o asignando incorrectamente la categoría del artículo [3]. A su vez, se podrían detectarse casos de artículos asignados de forma malintencionada a categorías erróneas.

Por otra parte, la clasificación de documentos puede ser de gran utilidad para usuarios comunes [8]. La descarga de documentos se ha vuelto una tarea habitual en los usuarios de Internet; sin embargo, puede ocurrir el caso de no organizar

¹https://en.wikipedia.org/wiki/Main_Page/

adecuadamente los documentos de texto descargados o generados por los mismos usuarios. Emplear tiempo para clasificar los documentos manualmente podría ser un opción, no obstante, se requiere tiempo y esfuerzo extra, un clasificador de documentos sería de gran ayuda para estos casos.

Este trabajo tiene como objetivo desarrollar un sistema clasificador de documentos; el estudio se basa en artículos de Wikipedia para entrenar el algoritmo de clasificación, posteriormente clasificar los documentos de forma automática en alguna de las categorías de documentos. Aunque podemos aplicar el sistema a la clasificación automática de artículos en la misma Wikipedia, también la propuesta permite organizar grandes cantidades de documentos clasificándolos en el nivel de especialización que el usuario desee. Los documentos podrían clasificarse en las categorías principales de Wikipedia de forma general, o clasificarlos en categorías más específicas, por ejemplo solo documentos del área de matemáticas.

Este trabajo se organiza de la siguiente forma. La Sección 2 presenta los trabajos relacionados. La sección 3 muestra los conceptos relacionados a la clasificación de documentos así como los clasificadores más comunes. La Sección 4 presenta la arquitectura del sistema de clasificación. La sección 5 presenta un experimento para definir el clasificador que mejor se adapta a las características del problema. La sección 6 realiza otro experimento con documentos sin relación a Wikipedia para determinar su nivel de precisión. La sección 7 describe la discusión de los experimentos. Por último, se muestran las conclusiones y referencias de la investigación.

II. TRABAJOS RELACIONADO

La clasificación de documentos es una actividad con mucho avances; existen varios trabajos relacionados a nuestra investigación, donde se consideran los conceptos relacionados: Clasificación de documentos y Wikipedia.

Weale presentó en [9] una técnica para integrar Wikipedia como una base de conocimiento de amplia cobertura para su uso en la clasificación de documentos. Este autor esbozó un algoritmo para integrar las categorías de Wikipedia obtenidas de entidades nombradas en los artículos.

El artículo [3] presenta una aplicación para la clasificación automática de documentos de Wikipedia. Los resultados de los experimentos realizados sobre datos hechos a mano muestran que el proceso de clasificación de la Wikipedia puede ser parcialmente automatizado. El enfoque propuesto puede utilizarse para construir un sistema de apoyo a la decisión que sugiera a los editores las mejores categorías que se ajusten a los nuevos contenidos ingresados en Wikipedia.

El clasificador propuesto en [8] toma como base la Wikipedia para la creación del corpus que define una categoría mediante técnicas de Procesado de Lenguaje Natural (PLN) que analizan sintácticamente los textos a clasificar. El resultado final del sistema propuesto presenta un alto porcentaje de acierto, incluso cuando se compara con los resultados obtenidos con técnicas alternativas de Aprendizaje Automático.

Estos trabajos utilizan el corpus de Wikipedia para clasificar documentos, sin embargo, no definen si el clasificador emplea-

Tabla I: Algoritmos de clasificación mas comunes

| Clasificador | 1 | 2 | 3 | 4 | 5 |
|------------------------------|---|---|---|---|---|
| Arboles de decisión | x | x | x | x | x |
| Bayes Ingenuo | x | x | x | x | x |
| Inducción de reglas | x | | | | |
| Redes neuronales | x | x | x | x | x |
| Vecino más cercano | x | x | x | x | |
| Máquina de soporte vectorial | | x | x | x | |
| Algoritmo de Rocchio | | x | | | |
| Correlación difusa | | x | | | |
| Algoritmos genéticos | | x | | | |
| Método de Roche | | | x | | |
| Red growing | | | x | | |
| Método de subespacio | | | | | x |

do es el mejor para las características de Wikipedia. Por otra parte, los artículos revisados no desarrollan una herramienta para organizar documentos fuera de la web, la cual es nuestra propuesta principal. A diferencia de los artículos mencionados, esta investigación utiliza un conjunto de datos con artículos de Wikipedia, para entrenar y para probar el rendimiento de los clasificadores.

III. CONCEPTOS RELACIONADOS

A continuación se muestran los conceptos relacionados a nuestra investigación. Esta teoría permite entender los procesos y los métodos empleados a través del estudio, por tanto se consideró importante incluirlos.

III-A. Clasificación de documentos

Los documentos pueden clasificarse por tres vías:

- Métodos no supervisados: Los métodos no supervisados no consideran la variable clase, así que los atributos continuos son discretizados independientemente [10].
- Supervisados: Un método de aprendizaje supervisado es aquel que requiere como entrada una serie de ejemplos etiquetados para derivar un modelo predictivo. Cada ejemplo consiste en un vector de características, y la etiqueta consiste en la clase (o valor numérico) correcto para el ejemplo [11].
- Semi-supervisados: La clasificación semi-supervisada se refiere al uso de una gran cantidad de documentos sin etiquetar junto con una cantidad menor de documentos etiquetados en el proceso de aprendizaje [12].

Recientemente se proponen muchas técnicas y algoritmos para la agrupación y clasificación de documentos electrónicos. Este trabajo se centra en las técnicas de clasificación supervisada. A partir de los últimos años, la tarea de la clasificación automática de textos ha sido ampliamente estudiada y se observa un progreso rápido en esta área [12].

La Tabla I muestra los algoritmos de clasificación más comunes de acuerdo a la literatura revisada en el primer renglón. Aún existen varios algoritmos, los más comunes son: Árboles de decisión, Bayes ingenuo, redes neuronales, vecino más cercano y máquina de soporte vectorial. Los artículos revisados fueron [5], [12], [13], [2], [4]:

III-A1. Árboles de decisión: Los árboles de decisión son modelos que realizan comprobaciones incrementales de elementos vectoriales para decidir la clase de un objeto. El algoritmo encuentra el elemento más distintivo del vector y divide basado en ese elemento. Entonces, recursivamente encuentra el elemento más distintivo de sus subconjuntos, hasta que se encuentra un nivel suficiente de generalización. Los árboles de decisión tienen muchos beneficios, entre los que destacan: los resultados son fáciles de interpretar, fácilmente reproducibles por el hombre y pueden proporcionar generalizaciones con pocos datos duros. Sin embargo, pueden ser propensos a un exceso de ajuste [9].

III-A2. Bayes ingenuo: Bayes ingenuo se utiliza regularmente en aplicaciones de clasificación de texto y experimentos debido a su facilidad y eficacia [5]. El clasificador Bayes ingenuo es uno de los algoritmos que estiman probabilidades a posteriori. Este clasificador asume, para una muestra x , que sus atributos x_1, x_2, \dots, x_n presentan una independencia condicional dado el valor de la clase, por lo que la probabilidad condicional puede expresarse como el producto de funciones de probabilidad condicional de cada atributo por separado [14].

III-A3. Redes neuronales: La clasificación de redes neuronales es una red de unidades, donde las unidades de entrada suelen representar palabras y las unidades de salida representan una clase o la etiqueta de clase. Para clasificar un documento de prueba, se determina el peso de las palabras para las unidades de entrada y la activación de estas unidades se realiza a través de la propagación directa en la red y el valor de la unidad de salida se determina como resultado de la decisión de las clases. Algunas investigaciones utilizan perceptrón de una sola capa, ya que su implementación es simple y el perceptrón de múltiples capas que es tan complejo requiere una implementación extensiva para la clasificación. El uso de un método eficaz de selección de características para reducir la dimensionalidad mejora la eficiencia en este método [2].

III-A4. Vecino más cercano: El algoritmo del vecino más cercano es un algoritmo simple que almacena todos los casos disponibles y clasifica nuevos casos con base en una medida de similitud. Este algoritmo ha sido usado en la estimación estadística y reconocimiento de patrones [4]. El algoritmo del vecino más cercano supone que los datos están en un espacio métrico, es decir, los puntos de datos están en un espacio métrico. Los datos pueden ser escalares o posiblemente vectores multidimensionales. Puesto que los puntos están en el espacio de la característica, tienen una noción de la distancia; ésta no necesita ser necesariamente la distancia euclidiana aunque es comúnmente utilizada. Cada uno de los datos de entrenamiento consta de un conjunto de vectores y etiquetas de clase asociadas con cada vector. En el caso más simple, será positivo o negativo. Pero el algoritmo puede funcionar igual de bien con un número arbitrario de clases [15].

III-A5. Máquina de soporte vectorial: El método máquina de soporte vectorial (SVM por sus siglas en inglés Support Vector Machine) es uno de los métodos de aprendizaje supervisados utilizados para la clasificación y regresión. El método

de clasificación SVM se basa en la teoría del aprendizaje aritmético basada en el principio de minimización del riesgo estructural. La idea de este principio es encontrar una hipótesis que garantice el menor error. La SVM requiere dos conjuntos de entrenamiento positivo y negativo que es inusual para otros métodos de clasificación. Este conjunto de entrenamiento positivo y negativo es necesario para la SVM para buscar un nivel de decisión con el fin de separar los datos positivos y negativos dentro del espacio n -dimensional de la mejor manera que se llama hiperplano. Por lo tanto, la SVM crea un hiperplano o un conjunto de grandes superficies en un espacio de alta dimensiones [2].

III-B. Procesamiento de los datos

Es preciso contar con una forma consistente de representar cada documento (su contenido), para ello se utilizará el modelo espacio vectorial [12], que es uno de los más empleados. El modelo vectorial básicamente consiste en que cada documento puede ser representado mediante un vector de términos. Cada término lleva asociado un coeficiente o peso que trata de expresar la importancia o grado de representatividad de ese término en ese vector o documento. Este peso puede calcularse de forma automática a partir de diversos elementos: basándose en la frecuencia de términos, tanto en toda la colección de documentos con que se trabaje, como dentro de cada documento en particular [16].

A continuación, explicamos las fases consideradas en este trabajo:

- Selección de texto: Esta etapa elige solamente el texto en formato plano del artículo de Wikipedia, esto no incluye las referencias de los artículos, ni formato html.
- Tokenización: Cada artículo es transformado en un listado de términos llamados tokens. Esta representación de datos también es conocida como bolsa de palabras (bag of words). Los tokens son cadenas de caracteres entre espacios en blanco o puntuación, pero no siempre es así, como por ejemplo en el caso de las abreviaturas. El conjunto total de palabras utilizadas, distintas y únicas, es el vocabulario del corpus.
- Palabras de parada (Stop-Words): Consisten en palabras comunes tales como: Determinadores (a, the, this), preposiciones (in, from, into, of, on), conjunciones (and, but, after, since, as) [17], [18]. Este tipo de palabras aparecen siempre en las primeras posiciones en las frecuencias de palabras ordenadas de un texto [19]. En el enfoque no lingüístico que se maneja en esta investigación se pueden omitir este tipo de palabras que se consideran innecesarias.
- Stemming o lematización: Esta técnica sirve para reducir variantes morfológicas de las formas de una palabra a raíces comunes o lexemas [18]. Básicamente, este consiste en remover el plural, el tiempo, o los atributos finales de la palabra. Por ejemplo cars=car, musical=music.
- Vectorización: El próximo paso consiste en generar una matriz (sparse matrix), transformando cada lista de palabras en un vector en un espacio Euclideo, donde cada

columna es una característica. Las características son principalmente palabras del vocabulario extraído de la tokenización. De esta forma, podemos considerar cada palabra del vocabulario como una columna, o podemos obtener representaciones mas detalladas si consideramos como posibles características secuencias de palabras, llamadas n-gramas, que han ocurrido en el texto. Los n-gramas pueden ser secuencias de una palabra (uni-gramas), de dos palabras (bigramas), de tres palabras (trigramas), y así sucesivamente.

- Frecuencia. La representación de las palabras en los artículos de Wikipedia con valores binarios produce demasiada información y no captura algunas diferencias importantes. Para estos casos es útil la suma de todas las ocurrencias o el número de veces que aparece un término en un documento. A este tipo de frecuencia de aparición también se la denomina "Frecuencia de aparición relativa" por que atañe a un documento en concreto y no a toda la colección.
- TF-IDF (término frecuencia-inverso de la frecuencia del documento): Es un peso que se utiliza a menudo en la recuperación de la información y la minería de texto. Este peso es una medida estadística utilizada para evaluar la importancia de una palabra para un documento en una colección o corpus. La importancia aumenta proporcionalmente al número de veces que una palabra aparece en el documento, pero es compensada por la frecuencia de la palabra en el corpus. Las variaciones del esquema de ponderación TF-IDF son a menudo utilizadas por los motores de búsqueda como una herramienta central para anotar y clasificar la relevancia de un documento dada una consulta del usuario [20].

III-C. Validación cruzada

La validación se utiliza para probar el rendimiento del clasificador. La validación cruzada genera aleatoriamente una partición en K bloques de tamaño N/K . El entrenamiento se lleva a cabo empleando $K-1$ bloques, mientras que el subconjunto restante es empleado como prueba. Este procedimiento es repetido K veces, eligiendo en cada iteración una parte diferente para prueba. El rendimiento general del clasificador se calcula con un promedio del rendimiento de las iteraciones [21].

III-D. Métricas de evaluación

Las métricas típicas de recuperación de información son precisión y exhaustividad [22]. Precisión (Ecuación 1) es el número de documentos correctamente clasificados sobre el total de documentos clasificados, para una categoría dada. Exhaustividad (Ecuación 2) es el número de documentos clasificados correctamente sobre el total de documentos de esa categoría.

$$precision = \frac{|\{Obj_relevantes\} \cap \{Obj_recuperados\}|}{|\{Obj_recuperados\}|} \quad (1)$$

$$exhaustividad = \frac{|\{Obj_relevantes\} \cap \{Objetos\}|}{|\{Obj_relevantes\}|} \quad (2)$$

$$F_{\beta} = (1 + \beta^2) * \frac{precision * exhaustividad}{(\beta^2 * precision) + exhaustividad} \quad (3)$$

La medida F1 (Ecuación 3) es una media armónica ponderada de exhaustividad y precisión; donde, β es una variable para dar preferencia a la exhaustividad o a la precisión, cuando $\beta > 1$ entonces la preferencia se da a la precisión, cuando $\beta < 1$ entonces la preferencia se da para la exhaustividad.

IV. ENFOQUE

La propuesta se basa en dos módulos principales: procesos de soporte y clasificador; un módulo secundario, preprocesamiento; y la enciclopedia Wikipedia. A esto se le agregan la entrada (documentos) y la salida (documentos organizados por categorías).

El proceso inicia con la entrada de un conjunto de documentos. A continuación, el módulo de soporte solicita los archivos de entrenamiento de Wikipedia, el usuario elige la categoría considerando el tipo de documentos que se manejan, los documentos se clasificaran en las subcategorías dentro de la categoría elegida. Antes de llegar al algoritmo de clasificación, los artículos de Wikipedia y los documentos de entrada son sometidos a un proceso de limpieza (procesamiento), una vez ejecutado el proceso anterior, el clasificador de documentos se encarga de categorizar los documentos de acuerdo a las subcategorías de Wikipedia pertenecientes a la categoría principal. La clasificación de documentos es enviada al módulo de soporte para crear las carpetas y ubicar los documentos de entrada en su respectiva categoría.

A continuación se detallan las funciones de cada modulo

- Procesos de soporte: Es el modulo que se encarga de coordinar todo el proceso del sistema, recibe los datos y los distribuye por el resto de los módulos y secciones. Además se encarga de generar el producto del sistema.
- Clasificador: Como su nombre lo indica clasifica los documentos en la categoría correspondiente, se basa en un set de datos de entrenamiento para determinar la pertenencia el documento seleccionado.
- Preprocesamiento: Se encarga de procesar el texto para que los algoritmos puedan tratarlo como números en lugar de palabras. Considera los procesos de: eliminación de palabras de parada [17], [18], aplicación del algoritmo TF-IDF [20], [23] y aplicación de stemming [24].
- Wikipedia: Esta parte del sistema se encarga de proveer los artículos y las categorías a donde pertenece, estos son el conjunto de entrenamiento para el clasificador.

V. SELECCIÓN DEL CLASIFICADOR

Esta sección compara los algoritmos: Arboles de decisión, Bayes ingenuo, redes neuronales, vecino más cercano y máquina de soporte vectorial, para detectar el clasificador con el mejor rendimiento, el cual será el clasificador más idóneo para tratar artículos de Wikipedia.

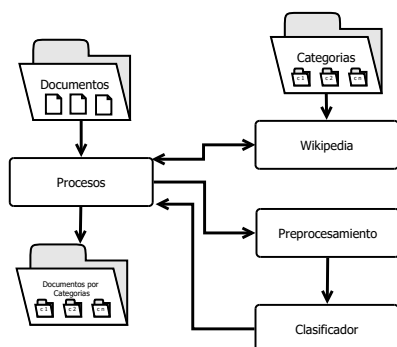


Figura 1: Arquitectura general del sistema de clasificación

Tabla II: Conjunto de artículos por categorías de Wikipedia (Main topic classifications)

| Categorías | Artículos |
|------------------------|-----------|
| Geography | 1085 |
| Health | 2530 |
| History | 1620 |
| Humanities | 2281 |
| Mathematics | 1221 |
| Nature | 536 |
| People | 297 |
| Philosophy | 955 |
| Reference works | 1425 |
| Science and technology | 585 |
| Total | 12535 |

V-A. Experimento

V-A1. Conjunto de datos: El objetivo es clasificar documentos en categorías de Wikipedia, por tanto se tomó un conjunto de datos de Wikipedia de la versión de enero de 2017. Se consideró la categoría principal de Wikipedia (Main topic classifications) en inglés compuesta de 11 subcategorías generales (ver Tabla II), se consideraron todos los artículos a un nivel de profundidad de cada subcategoría, en total se obtuvieron 12535 artículos. Estos artículos se tomaron como el set entrenamiento y de prueba, considerando las mismas subcategorías para definir la clase del artículo.

V-A2. Proceso: El proceso del experimento consistió en darle como entrada a los clasificadores el conjunto de datos de Wikipedia, se utilizó una validación cruzada con 5 divisiones, por tanto el promedio del rendimiento de cada validación fue el resultado final del clasificador.

V-A3. Implementación de los algoritmos: Los algoritmos fueron implementados a través del lenguaje de programación Python² con apoyo de la librería open source de aprendizaje automático Scikit-Learn³. Cada algoritmo consideró parámetros para obtener el mejor rendimiento de acuerdo a las necesidades de la tarea, estos parámetros fueron optimizados a través de la herramienta Grid-Search de Scikit-Learn.

²<https://www.python.org/>

³<http://scikit-learn.org/stable/index.html>

V-B. Resultados

Los resultados de los clasificadores se muestran en la Tabla III. El eje x representa a las 5 divisiones de la validación cruzada. El eje y representa la precisión obtenida por cada clasificador, cada uno de los cuales está representado en la figura por un símbolo distinto. La gráfica muestra de forma general que los clasificadores mejor evaluados fueron la máquina de soporte vectorial y Bayes ingenuo. Por otra parte, el algoritmo del vecino más cercano fue el que brindó los puntajes más bajos.

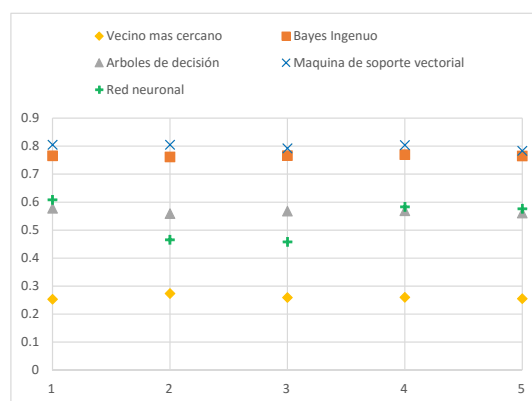


Figura 2: Resultado de las iteraciones de las pruebas.

La Tabla III muestra el valor promedio y la desviación estándar de la precisión en la clasificación de los documentos. La máquina de soporte vectorial obtuvo un valor cercano al 80 % de precisión siendo el promedio más alto. El algoritmo del vecino más cercano tuvo un promedio del 26 % siendo el promedio más bajo.

Tabla III: Media y desviación estándar de los clasificadores.

| Clasificador | Media | ds |
|------------------------------|----------|----------------|
| Vecino más cercano | 0.260249 | ± 0.007046 |
| Bayes ingenuo | 0.765593 | ± 0.002776 |
| Árboles de decisión | 0.566994 | ± 0.006743 |
| Máquina de soporte vectorial | 0.797652 | ± 0.008677 |
| Redes neuronales | 0.538116 | ± 0.063244 |

V-C. Discusión

Los artículos de Wikipedia se caracterizan porque un documento puede pertenecer a más de una categoría, de esta forma pueden existir ambigüedades. Los clasificadores tuvieron que lidiar con esta problemática, algunos como máquina de soporte vectorial y Bayes ingenuo fueron los algoritmos que mejor clasificaron los documentos, otros como el algoritmo vecino más cercano tuvo muchos problemas para la clasificación por tal motivo sus resultados fueron bajos.

Como lo muestra la Figura 2, el algoritmo mejor ubicado fue la máquina de soporte vectorial, con una precisión de 0.797 y una desviación estándar de ± 0.008 . Este clasificador demostró adaptarse mejor a las características de los documentos de Wikipedia que los otros algoritmos de acuerdo al experimento

realizado. Por tanto, el sistema de clasificación de documentos propuesta tiene como base a la máquina de soporte vectorial.

VI. CLASIFICACIÓN DE DOCUMENTOS

Esta sección emplea la herramienta de software para clasificar documentos que no pertenecen a Wikipedia con el objetivo de determinar el rendimiento ante documentos distintos al de entrenamiento.

VI-A. Experimento

VI-A1. Conjunto de datos: El conjunto de datos está formado por 26 documentos de texto de diversos temas y distintas fuentes. La Tabla IV en la columna soporte representa el total de documentos por categorías. Todos los documentos están formados solamente de texto, no contienen imágenes.

VI-A2. Sistema de clasificación: La interfaz del prototipo está formada por los siguientes elementos (ver Figura 3):

1. Ruta origen: Ruta donde se encuentran los artículos para organizar.
2. Ruta destino: Ruta donde se colocarán las carpetas generadas y los artículos categorizados.
3. Categoría de Wikipedia: Corresponde al nombre de la categoría en que se basara la categorización de documentos, se consideran las subcategorías que están contenidas en la categoría principal.
4. Clasificar: Inicia el proceso de construcción de carpetas, clasificación y organización de los documentos.

Figura 3: Pantalla de interacción con el usuario.

VI-A3. Proceso: El proceso de prueba consistió en las siguientes etapas:

1. Los documentos fueron puestos en una carpeta sin indicar la categoría a la que pertenecen.
2. La ruta de los documentos sin categorizar fue ingresada al sistema de clasificación de documentos (ruta origen).
3. La ruta destino fue ingresada al sistema de clasificación de documentos.
4. La categoría de Wikipedia elegida fue "Main topic classifications"(categoría principal).
5. El sistema de clasificación de documentos creó las carpetas correspondientes a las subcategorías de la categoría principal de Wikipedia elegida,
6. Los documentos sin categorizar fueron ubicados en la carpeta determinada por el sistema, la cual representa la categoría.
7. Se aplicaron las métricas de evaluación para determinar la precisión de la clasificación.

Tabla IV: Métricas para las pruebas de clasificación de documentos

| Categoría | Precisión | Exhaustividad | Medida F | Soporte |
|--------------------|-------------|---------------|-------------|-----------|
| Geography | 0.67 | 0.50 | 0.57 | 4 |
| History | 0.80 | 0.80 | 0.80 | 5 |
| Mathematics | 1.00 | 1.00 | 1.00 | 4 |
| Nature | 0.67 | 0.80 | 0.73 | 5 |
| People | 1.00 | 0.80 | 1.00 | 5 |
| Philosophy | 1.00 | 1.00 | 1.00 | 3 |
| Prom./total | 0.85 | 0.85 | 0.84 | 26 |

Tabla V: Matriz de confusión de la clasificación de documentos

| Categorías | Geography | History | Mathematics | Nature | People | Philosophy |
|-------------|-----------|---------|-------------|--------|--------|------------|
| Geography | 2 | 0 | 0 | 2 | 0 | 0 |
| History | 1 | 4 | 0 | 0 | 0 | 0 |
| Mathematics | 0 | 0 | 4 | 0 | 0 | 0 |
| Nature | 0 | 1 | 0 | 4 | 0 | 0 |
| People | 0 | 0 | 0 | 0 | 5 | 0 |
| Philosophy | 0 | 0 | 0 | 0 | 0 | 3 |

VI-B. Resultados

Las pruebas de clasificación de documentos mostraron los resultados en la Tabla IV. La tabla muestra las métricas de precisión, exhaustividad y la medida F, una columna extra (soporte) muestra el total de documentos para cada categoría. La categoría más problemática fue *geography* con un precisión de 0.67 y una exhaustividad de 0.50. Tres categorías obtuvieron resultados perfectos (*mathematics*, *people* y *philosophy*). En promedio se obtuvo un precisión de 0.85, un exhaustividad de 0.85 y una medida F de 0.84, sobre 26 documentos en 6 categorías.

La matriz de confusión representada en la Tabla V muestra con más exactitud los aciertos y errores de la clasificación. La primer columna y renglón representan las categorías, el caso ideal sería que la intersección entre cada categoría tuviera el total de casos por categoría, sin embargo, existen documentos mal clasificados.

VI-C. Discusión

Los resultados muestran una medida F de 0.84, este valor superó los resultados de clasificación obtenidos en la Tabla I, donde se obtuvo 0.79. Esto se debe en gran medida a la cantidad de documentos empleados en cada prueba, para la prueba de esta sección se emplearon 26 documentos tratando de ejemplificar el uso del sistema, además, considerando que un usuario promedio no tendrá miles de documentos desordenados.

La matriz de confusión (Tabla V) muestra algunos documentos mal clasificados de acuerdo a las categorías asignadas en las pruebas, sin embargo, los documentos pueden tener más de una clasificación de acuerdo a su temática. Un documento que habla sobre los primeros juegos olímpicos de la historia fácilmente podría entrar en la categoría de deportes y también en la de historia. La categoría *geography* obtuvo una medida F, sin embargo, si analizamos sus resultados vemos que dos de los documentos fueron clasificados en la categoría *nature*, claramente los documentos podrían ser catalogados en esa categoría, ya que la geografía maneja aspectos de la naturaleza.

Los resultados muestran que la clasificación de documentos de forma automática con base en información de Wikipedia obteniendo resultados confiables.

VII. CONCLUSIONES

Esta investigación desarrolló un prototipo de software para clasificar documentos de forma automática considerando categorías y documentos de Wikipedia. La investigación realizó un experimento para determinar al mejor clasificador de los más comúnmente empleados en la literatura. Esto fue con el objetivo de definir el clasificar que mejor se adapte a las características de la problemática e información de Wikipedia. Posteriormente se desarrolló el prototipo probándolo con documentos no relacionados con Wikipedia.

Los resultados muestran resultados cercanos al 80% de efectividad, considerando que un documento solo puede estar en una sola categoría, sin embargo, si consideramos múltiples categorías podríamos incrementar el valor de efectividad.

Los principales aportes de esta investigación fueron:

- Un conjunto de datos de prueba alternativo a los ya existentes en la literatura para clasificar documentos.
- Se determinó la máquina de soporte vectorial como el algoritmo que mejor se adapta a la clasificación de documento de acuerdo al conjunto de datos empleado.
- Un sistema prototipo de clasificación de documentos basado en categorías de Wikipedia.

Esta investigación hizo algunas aportaciones, sin embargo, aún quedan aspectos por mejorar, para el trabajo futuro se plantean los siguientes aspectos:

- Detección de documentos en carpetas y subcarpetas en una ruta definida.
- Realizar experimentos considerando que un artículo puede estar en más de una categoría y estar correctamente clasificado.
- En versiones mejoradas el algoritmo deberá analizar el corpus de los documentos a clasificar y determinar una categoría a la que pertenecen los documentos. Por el momento indicamos manualmente la categoría.

REFERENCIAS

- [1] M. A. Pérez and C. A. Cardoso, "Minería de texto para la categorización automática de documentos," 2010, pp. 11–45.
- [2] N. Vafsisisi, M. Reza, and F. Derakhshi, "Text Classification with Machine Learning Algorithms," *Journal of Basic and Applied Scientific Research*, vol. 3, pp. 31–35, 2013.
- [3] J. Szymański, "Towards Automatic Classification of Wikipedia Content," in *Intelligent Data Engineering and Automated Learning – IDEAL 2010: 11th International Conference*, C. Fyfe, P. Tino, D. Charles, C. Garcia-Osorio, and H. Yin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 102–109. [Online]. Available: https://doi.org/10.1007/978-3-642-15381-5_13
- [4] Y. H. Li and a. K. Jain, "Classification of Text Documents," *The Computer Journal*, vol. 41, no. 8, pp. 537–546, 1998.
- [5] D. Bhavani Dasari and V. Venu Gopala Rao. K, "Text Categorization and Machine Learning Methods: Current State Of The Art," *Global Journal of Computer Science and Technology*, vol. 12, no. 11-C, 2012.
- [6] X. Hu, X. Zhang, C. Lu, E. K. Park, and X. Zhou, "Exploiting Wikipedia As External Knowledge For Document Clustering," *International Conference On Knowledge Discovery And Data Mining*, pp. 389–396, 2009.
- [7] P. Wang and C. Domeniconi, "Building semantic kernels for text classification using wikipedia," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Las Vegas, Nevada, USA: ACM, 2008, pp. 713–721.
- [8] J. M. Quinteiro-González, E. Martel-Jordán, P. Hernández-Morera, J. A. Ligeró-Fleitas, and A. López-Rodríguez, "Clasificación de textos en lenguaje natural usando lawikipedia," *RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao*, vol. 2011, no. 8, pp. 265–281, 2011.
- [9] T. Weale, S. Spielberg, G. Lucas, and P. Jackson, "Utilizing Wikipedia Categories for Document Classification," *Evaluation*, p. 4, 2006.
- [10] L. Sucar, "Introduction to Bayesian Networks and Influence Diagrams," in *Decision Theory Models for Applications in Artificial Intelligence: Concepts and Solutions*, 2012, pp. 9–32.
- [11] L. P. D. Bosque and S. E. Garza, "Prediction of Aggressive Comments in Social Media an Exploratory Study," vol. 14, no. 7, pp. 3474–3480, 2016.
- [12] B. Baharudin, L. H. Lee, and K. Khan, "A Review of Machine Learning Algorithms for Text-Documents Classification," *Journal of Advances in Information Technology*, vol. 1, no. 1, pp. 4–20, 2010.
- [13] T. Golub, "The analysis of text documents classifiers constructing methods," in *13th International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET)*, 2016, pp. 742–745.
- [14] Y. González-Meneses, B. Pedroza-Méndez, F. López-Briones, C. Pérez-Corona, and J. Ramírez-Cruz, "Implementación del clasificador Naive Bayes para la Acentuación Automática de Palabras Ambiguas del Español." *Congreso Internacional de Cómputo en Optimización y Software*, vol. 6, pp. 322–336, 2011.
- [15] A. Rettinger, U. Lösch, V. Tresp, C. D'Amato, and N. Fanizzi, "Mining the semantic web: Statistical learning for next generation knowledge bases," *Data Mining and Knowledge Discovery*, vol. 24, no. 3, pp. 613–662, 2012.
- [16] R. Vences, J. Gómez, and V. Menéndez, "A Document Recommendation System Using a Document-Similarity Ontology," *IEEE Latin America Transactions*, vol. 14, no. 7, pp. 3329–3334, 2016.
- [17] C. Fox, "A stop list for general text," *ACM SIGIR Forum*, vol. 24, no. 1-2, pp. 19–21, 1989.
- [18] R. Bhowmik, "Keyword extraction from abstracts and titles," in *Conference Proceedings - IEEE SOUTHEASTCON*, 2008, pp. 610–617.
- [19] G. K. Zipf, "Human behavior and the principle of least effort," *Journal of Clinical Psychology*, vol. 6, no. 3, p. 306, 1950.
- [20] L. Zhiqiang, S. Werimin, and Y. Zhenhua, "Measuring Semantic Similarity between Words Using Wikipedia," *2009 International Conference on Web Information Systems and Mining*, pp. 251–255, 2009.
- [21] J. López-de Uralde, I. Ruiz, I. Santos, A. Zubillaga, P. G. Bringas, A. Okariz, and T. Guraya, "Automatic Morphological Categorisation of Carbon Black Nano-aggregates," in *Database and Expert Systems Applications: 21th International Conference, DEXA 2010*, P. G. Bringas, A. Hameurlain, and G. Quirchmayr, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 185–193.
- [22] A. Nenkova and K. McKeown, "Automatic Summarization," in *Foundations and Trends in Information Retrieval*, 2011, vol. 5, pp. 103–233.
- [23] R. P. Biuk-aghai and K. K. Ng, "A method for Automated Document Classification Using Wikipedia-Derived Weighted Keywords," in *Data and Software Engineering (ICODSE), 2014 International Conference on*. Bandung: IEEE, 2014, pp. 1–6.
- [24] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 40, no. 3, pp. 211–218, 1980.

Detección de Alteraciones en Imágenes mediante Watermarking basado en DWT y Cómputo Paralelo

Javier Molina-García, Volodymyr Ponomaryov, Rogelio Reyes-Reyes,
Clara Cruz-Ramos

Instituto Politécnico Nacional, ESIME Unidad Culhuacán
Av. Santa Ana No. 1000, Col. San Francisco Culhuacán, CP. 04430, México, D.F.
Tel. (55) 5729-6000 Ext. 73264 Fax (55) 56562058 email: jmolinag1001@alumno.ipn.mx, rreyesre@ipn.mx

Resumen— El uso de herramientas de edición de imágenes como Photoshop® y Gimp® ha estado en constante crecimiento durante los últimos años, mediante estas herramientas es posible realizar modificaciones en imágenes digitales sin que el espectador final note la presencia de las alteraciones realizadas. Esto se vuelve un problema cuando dichas modificaciones son realizadas con el fin de dañar de manera económica o social a una persona o empresa. Durante los últimos años han surgido propuestas las cuales realizan la detección de alteraciones en imágenes digitales, la principal desventaja de estos algoritmos es que están diseñados para trabajar de manera serial, desaprovechando así el poder de procesamiento multi-núcleo que ofrecen distintos equipos como computadoras de escritorio o dispositivos móviles. El presente trabajo muestra un algoritmo paralelo para la detección de alteraciones en imágenes digitales, basado en la inserción de una marca de agua semi-frágil en el dominio de la Transformada Discreta de Wavelet (*DWT por sus siglas en inglés*), este algoritmo es desarrollado para trabajar de manera paralela sobre dispositivos con procesadores multi-núcleo. Los resultados obtenidos demuestran que las imágenes protegidas no son degradadas visualmente durante la inserción de la marca de agua y el algoritmo es capaz de detectar alteraciones en imágenes con distintos formatos, así mismo se tiene una reducción significativa de tiempo de procesamiento al ejecutar el algoritmo en equipos con procesadores multi-núcleo.

keywords— Autenticación de imagen, Detección de alteraciones, DWT, Marca de Agua, Watermarking, Cómputo paralelo.

I. INTRODUCCIÓN

Debido al crecimiento vertiginoso de las tecnologías de la información, las imágenes digitales han tomado un rol importante en diversas áreas como son: medicina, seguridad, navegación autónoma, análisis forense, etc. Así mismo, las imágenes digitales o fotográficas pueden ser empleadas como evidencia en casos legales, donde la alteración de estas imágenes puede definir el resultado de un caso legal.

De esta manera surge la necesidad de realizar la detección de alteraciones en imágenes digitales, estas alteraciones pueden ser divididas en dos tipos: alteraciones intencionales y alteraciones no intencionales. En el primer caso se encuentran las modificaciones hechas con el fin de alterar el contenido que es detectado visualmente por el Sistema Visual Humano (*HVS, por sus siglas en inglés*), ejemplos de este tipo de alteración son: el intercambio de color, ataque de collage, inserción de información extra, etc., en el segundo caso se encuentran las modificaciones que no alteran la información que es captada

por el espectador final en la imagen digital, ejemplo de estos tipos de ataque se encuentran la compresión JPEG, la cual modifica la información de los píxeles pero no altera de manera significativa el contenido visual de la imagen.

Uno de los métodos utilizados para la detección de alteraciones de contenido digital son las marcas de agua, las cuales consisten en insertar información dentro del contenido digital a proteger [1-5]. Las técnicas de marcado de agua pueden ser clasificadas de acuerdo a tres características: robustez, imperceptibilidad y capacidad de inserción, las cuales están ligadas entre sí, es decir, al momento de modificar una característica las otras se pueden ver afectadas.

Chetan et al. [6] propone un sistema, el cual utiliza una marca de agua frágil basada en la Transformada Discreta Contourlet, para la detección de alteraciones en documentos digitales, esta propuesta al ser frágil presenta falsas detecciones al aplicarse ataques no intencionales como compresión JPEG. Musaed Alhussein [7] Propone un método para la detección de alteraciones basado en un descriptor de textura local y en máquinas de aprendizaje, este sistema genera vectores de patrones binarios los cuales representan las texturas de la imagen, con estos patrones es entrenada la máquina de aprendizaje para la futura detección de alteraciones. La desventaja de este sistema es que requiere entrenas la máquina de aprendizaje por cada imagen utilizada, así mismo, es necesaria dicha máquina de aprendizaje como información adicional para la detección de alteraciones. No se presentan resultados sobre ataques no intencionales. Gadhya et al. [8] presenta un esquema para la detección de alteraciones en imágenes médicas basado en la Transformada Discreta de Wavelet, este sistema requiere la imagen original la cual es comparada con la imagen sospechosa de alteración. Se extraen características de la imagen original la imagen sospechosa de alteración mediante DWT y el operador canny, se calcula el promedio de cada borde y se sustraen ambas imágenes, la desventaja de este sistema es que la imagen original puede ser comprometida a ataques y dicha autenticación ya no sería efectiva. Thirunavukkarasu et al. [9] propone un método para la detección de alteraciones basado en el análisis de los primeros dígitos cuantizados de los coeficientes DCT, los cuales son mostrados en [10,11], la principal desventaja de este sistema es que la precisión de las regiones alteradas detectadas varían dependiendo del factor de calidad empleado en cada prueba.

La principal desventaja de los métodos propuestos en la

literatura es que en su gran mayoría no son robustos a alteraciones no intencionales, por lo tanto suelen detectar como alteración las modificaciones que no afectan la información visual de la imagen. Por otro lado, estas propuestas están pensadas para trabajar bajo un esquema de procesamiento serial, esto limita el desempeño del sistema ya que no se explotan las características del procesamiento paralelo el cual es ofrecido por la mayoría de los sistemas computacionales y dispositivos móviles.

En el presente trabajo se propone un sistema de marcado de agua semi-frágil basado en la DWT y en cómputo paralelo, empleado para detectar alteraciones en imágenes digitales y aprovechando la capacidad de procesamiento paralelo ofrecida por los dispositivos que poseen múltiples núcleos en el procesador. La imagen a color es dividida en sub-imágenes, donde cada sub-imagen es procesada de manera independiente por cada núcleo del procesador. El trabajo está organizado de la siguiente manera: en la sección II se describe el método de inserción, extracción y autenticación; en la sección III se muestran los resultados experimentales, en la sección IV las discusiones generadas a partir de los resultados y en la sección V se exponen las conclusiones generadas.

II. MÉTODO PROPUESTO

El sistema propuesto cuenta con tres etapas: protección de la imagen y detección de alteraciones. La figura 1 muestra el diagrama a bloques del sistema propuesto.

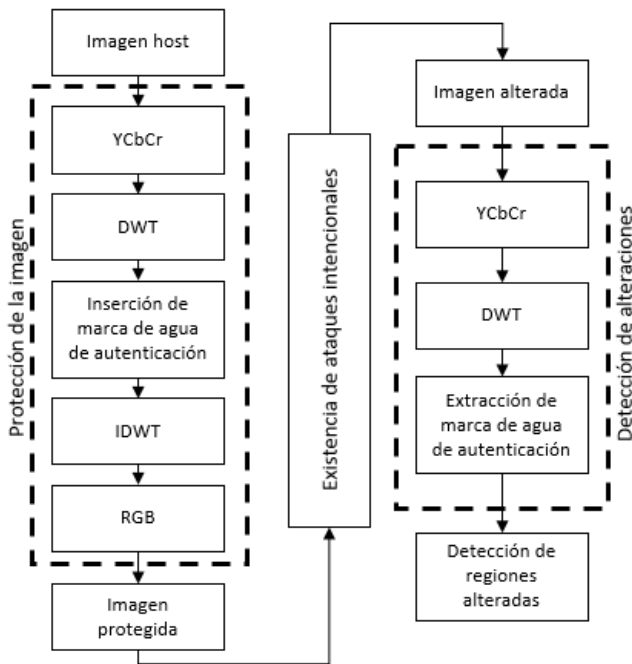


Figura 1. Diagrama a bloques general del sistema propuesto.

A. Protección de la imagen

La etapa de protección de la imagen consiste en insertar información (bits) dentro de la imagen, donde estos bits insertados son utilizados para verificar la autenticidad de la

información contenida dentro de la imagen a proteger. El método de inserción está basado en el dominio de las frecuencias, donde se utiliza la DWT mediante el filtro *Daubechies 4*. Sea I_h la imagen a proteger en el espacio de color RGB, y Δ la fuerza de inserción o paso de cuantificación, los pasos para realizar la protección de la imagen son los siguientes:

1. I_h es transformada al espacio de color YCbCr.
2. Al canal de luminancia se le aplica la transformada DWT, obteniendo las sub-bandas de frecuencia: HH_1 , HL_1 , LH_1 y LL_1 .
3. Se aplica la DWT a la sub-banda LL_1 , obteniendo las sub-bandas: HH_2 , HL_2 , LH_2 y LL_2 .
4. Se aplica Eq. 1 para realizar la inserción de la marca de agua de autenticación, esto se realiza para cada posición (i,j) de LH_2 .

$$LHW_2(i,j) = \frac{|LH_2(i,j)|}{LH_2(i,j)} \left\lfloor \frac{|LH_2(i,j)|}{2\Delta} \right\rfloor 2\Delta + \Delta \quad (1)$$

5. Se aplica la IDWT (*Inverse Discrete Wavelet Transform*) a las sub-bandas de frecuencia HH_2 , HL_2 , LHW_2 y LL_2 , obteniendo LL_{w1} .
6. Finalmente a las sub-bandas de frecuencia HH_1 , HL_1 , LH_1 y LL_{w1} se le aplica la IDWT para obtener el canal de luminancia marcado.
7. Por último, se transforma el canal de luminancia marcado junto con los canales de crominancia (obtenidos en el paso 1) del espacio de color YCbCr al espacio de color RGB.

B. Detección de alteraciones.

La etapa de detección de alteraciones consiste en extraer la información (bits) que fue insertada como marca de agua, estos bits son utilizados para comprobar la autenticidad de la información visual contenida en la imagen. Sea I_{hw} la imagen protegida y sospechosa de haber sido alterada, el proceso de detección de alteraciones se realiza de la siguiente manera:

1. I_{hw} es transformada al espacio de color YCbCr.
2. Al canal de luminancia se le aplica la transformada DWT, obteniendo las sub-bandas de frecuencia: HH_1 , HL_1 , LH_1 y LL_1 .
3. Se aplica la DWT a la sub-banda LL_1 , obteniendo las sub-bandas: HH_2 , HL_2 , LH_2 y LL_2 .
4. Se aplica la Eq. 2 para realizar la extracción de la marca de agua de autenticación, esto se lleva a cabo para cada posición (i,j) de LH_2 .
5. Finalmente, se aplica la Eq. 3 para detectar las regiones alteradas, mismas que son marcadas con cero.

$$wtm = \left\lfloor \left(\frac{LH_2(i,j)}{\Delta} \right) + 0.5 \right\rfloor \quad (2)$$

$$w(i,j) = \begin{cases} 0, & \text{si } \text{mod}(wtm, 2) = 0 \\ 1, & \text{otro} \end{cases} \quad (3)$$

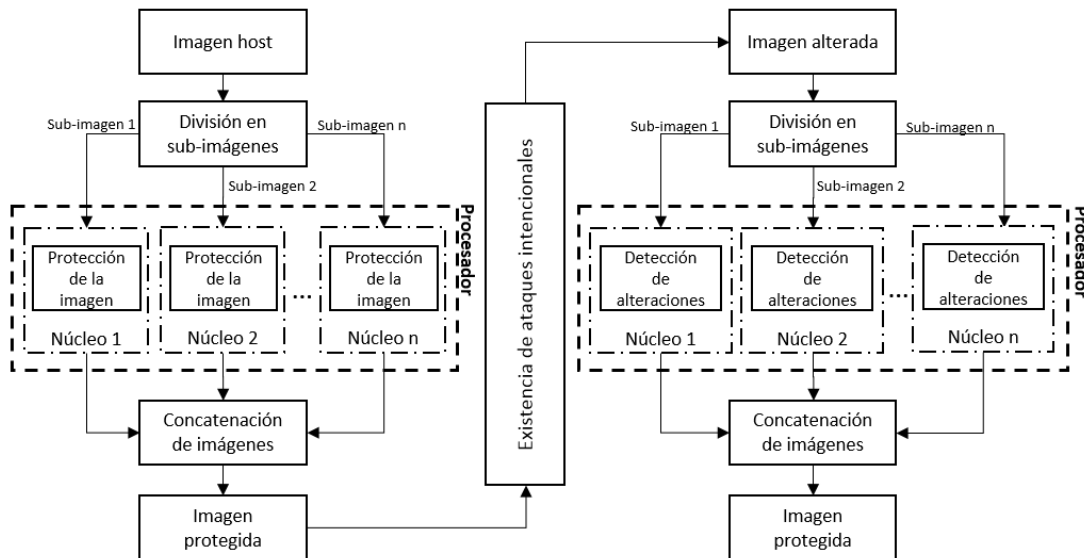


Figura 2. Diagrama a bloques general del sistema propuesto mediante cómputo paralelo.

C. Implementación mediante cómputo paralelo

Como se puede observar, los algoritmos mostrados en la figura 1 (“*Protección de la imagen*” y “*Detección de alteraciones*”) realizan el procesamiento de los datos de manera serial, ya que cada sub-proceso es ejecutado hasta que el sub-proceso anterior termina y envía el resultado a la salida. La implementación mediante cómputo en paralelo implica realizar una tarea de manera paralela en cada núcleo del procesador, a diferencia de la manera serial en la que es difícil saber de manera específica qué tarea realiza cada procesador.

Con el fin de realizar una implementación en paralelo utilizando los distintos núcleos del procesador, la imagen es dividida en sub-imágenes, donde el total de sub-imágenes obtenidas varía de acuerdo al número total de núcleos del procesador. Posteriormente cada imagen fue procesada (relación uno a uno) por cada núcleo del procesador de manera paralela. De esta manera técnicamente el tiempo de procesamiento total disminuye n veces comparándose con el tiempo de procesamiento total con una implementación serial. La figura 2 muestra el diagrama a bloques general de la implementación en paralelo.

El proceso de protección de la imagen mediante cómputo en paralelo se lleva a cabo mediante los siguientes pasos, Sea I_h la imagen a proteger en el espacio de color RGB:

1. I_h es dividida en n sub-imágenes, donde n representa el número total de núcleos del procesador.
2. Se envía a cada núcleo del procesador una sub-imagen generada para su procesamiento.
3. En cada procesador se realiza el algoritmo de *Protección de la imagen*, mostrado en la Figura 1.
4. Una vez procesada cada sub-imagen, concatenar cada una para generar la imagen protegida.

Para llevar a cabo la detección de alteraciones de manera

paralela, se realizan los siguientes pasos, sea I_{hw} la imagen protegida:

1. I_{hw} es dividida en n sub-imágenes, donde n representa el número total de núcleos del procesador.
2. Se envía a cada núcleo del procesador una sub-imagen generada para su procesamiento.
3. En cada procesador se realiza el algoritmo de *Detección de alteraciones*, mostrado en la Figura 1.
4. Una vez procesada cada sub-imagen, se concatena cada salida, obteniendo la imagen de autenticación.

Para que exista una reducción real de tiempo de procesamiento, el tiempo de procesamiento de los procesos “*División en sub-imágenes*” y “*Concatenación de imágenes*” y el tiempo en el traslado de memoria entre cada procesador y la memoria RAM deben ser inferior al tiempo requerido para llevar a cabo el algoritmo “*Protección de la imagen*”. De la misma manera, el tiempo de procesamiento requerido en el proceso “*División en sub-imágenes*” y el tiempo en el traslado de memoria entre cada procesador y la memoria RAM deben ser inferior al tiempo requerido para llevar a cabo el algoritmo “*Detección de alteraciones*”. De acuerdo a lo anterior mencionado, la reducción de tiempo de procesamiento depende del tamaño de las imágenes utilizadas.

III. RESULTADOS EXPERIMENTALES

Para evaluar el sistema propuesto se utilizaron imágenes de tamaño 768x512 píxeles, mismas que fueron interpoladas a distintos tamaños con el fin de evaluar el comportamiento de los resultados para las distintas pruebas realizadas. Estas imágenes fueron extraídas de la base de datos Kodak, son mostradas en la figura 3.



Figura 3. Imágenes utilizadas a)Kodak-3 b)Kodak-14 c) Kodak-21

Para las pruebas realizadas, se tomó en consideración la evaluación de: “Calidad de imágenes marcadas”, “Detección de alteraciones” y “Tiempos de procesamiento”.

A. Calidad de imágenes marcadas

La tabla 1 muestra los valores de PSNR de la imagen marcada contra la imagen original, utilizando un valor $\Delta = 30$. Este valor fue seleccionado debido a que se mantiene un balance entre imperceptibilidad y robustez de la marca de agua insertada.

Tabla 1. PSNR de la imagen protegida vs la imagen original

| Imagen | Tamaño | PSNR (dB) |
|----------|-----------|-----------|
| Kodak-3 | 512x786 | 37.26 |
| | 1024x1536 | 37.19 |
| | 1536x2304 | 36.72 |
| | 2048x3072 | 35.95 |
| | 2560x3840 | 35.77 |
| | 3072x4608 | 35.52 |
| Kodak-14 | 3584x5276 | 35.42 |
| | 512x786 | 38.59 |
| | 1024x1536 | 38.37 |
| | 1536x2304 | 38.01 |
| | 2048x3072 | 37.06 |
| | 2560x3840 | 36.62 |
| Kodak-21 | 3072x4608 | 36.05 |
| | 3584x5276 | 35.78 |
| | 512x786 | 37.81 |
| | 1024x1536 | 37.64 |
| | 1536x2304 | 37.28 |
| | 2048x3072 | 36.46 |
| | 2560x3840 | 36.17 |
| | 3072x4608 | 35.77 |
| | 3584x5276 | 35.58 |

El valor Δ puede ser modificado, pero es necesario tomar en consideración que si se elige un valor menor, la imperceptibilidad aumenta, pero la robustez ante ataques no intencionales disminuye. Finalmente, si se elige un valor Δ mayor, la imperceptibilidad de la marca de agua disminuye, pero la robustez ante los ataques aumenta.

B. Detección de alteraciones

La detección de alteraciones del sistema propuesto es evaluado modificando regiones de la imagen utilizando el software GIMP® y almacenando las imágenes obtenidas en el formato de compresión JPEG con distintos factores de calidad.

La figura 4 muestra las imágenes de prueba utilizadas después de realizar una alteración en el contenido visual de las mismas, se utilizó el software Gimp® y las imágenes fueron almacenadas con el formato de compresión JPEG utilizando factores de calidad 100, 90, 80 y 70.



Figura 4. Imágenes de prueba alteradas mediante el software Gimp®.

Como se puede observar, entre los ataques realizados se encuentran: adición de información adicional (Kodak-3), ataque de collage (Kodak-14) e intercambio de color (Kodak-21).

La Figura 5 muestra las imágenes generadas después del proceso de detección de alteraciones para la imagen Kodak-3 y los distintos factores de calidad utilizados al almacenar la imagen en el formato JPEG.

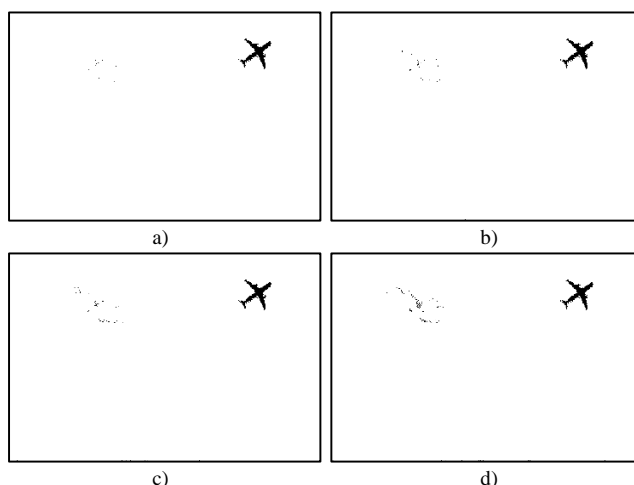


Figura 5. Detección de alteraciones para la imagen Kodak-3 A)Factor de calidad 100, b) Factor de calidad 90, c) Factor de calidad 80 y c) Factor de calidad 70

La Figura 6 muestra las imágenes generadas después del proceso de detección de alteraciones para la imagen Kodak-14 y los distintos factores de calidad utilizados al almacenar la imagen en el formato JPEG.

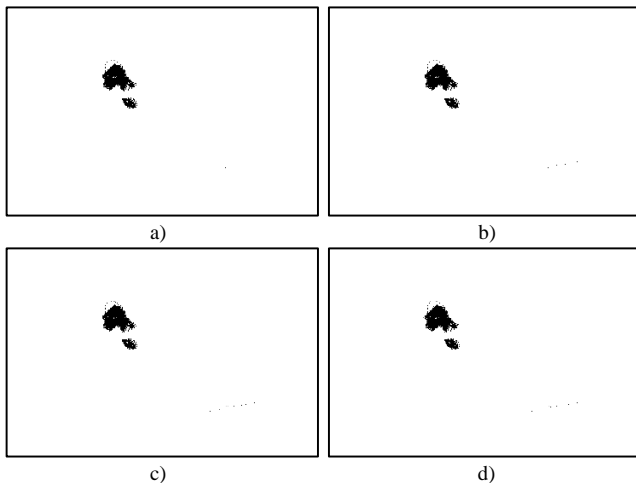


Figura 6. Detección de alteraciones para la imagen Kodak-14 A)Factor de calidad 100, b) Factor de calidad 90, c) Factor de calidad 80 y c) Factor de calidad 70

La Figura 7 muestra las imágenes generadas después del proceso de detección de alteraciones para la imagen Kodak-21 y los distintos factores de calidad utilizados al almacenar la imagen en el formato JPEG.

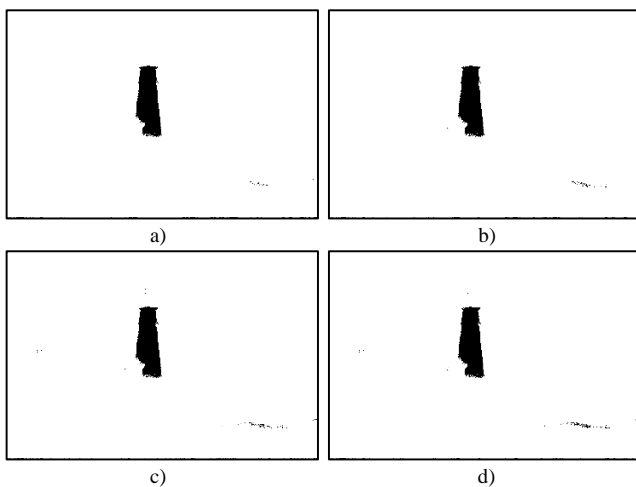


Figura 7. Detección de alteraciones para la imagen Kodak-21 A)Factor de calidad 100, b) Factor de calidad 90, c) Factor de calidad 80 y c) Factor de calidad 70

C. Tiempos de procesamiento

Esta sección presenta la evaluación de los tiempos de procesamiento del sistema propuesto, las características del equipo utilizado son: 8GB de memoria RAM, Procesador Intel® Core™ i5-4200U 1.6GHz, con dos núcleos de procesamiento.

La tabla 2 muestra la evaluación del sistema tomando en consideración el tiempo de procesamiento requerido para ejecutar el proceso serial y paralelo de la fase *protección de la imagen*.

Tabla 2. Tiempos de procesamiento en segundos para el proceso de protección de la imagen

| Imagen | Tamaño | Tiempo de procesamiento (segundos) | |
|----------|-----------|------------------------------------|----------|
| | | Serial | Paralelo |
| Kodak-3 | 512x768 | 1.198 | 2.499 |
| | 1024x1536 | 3.756 | 4.318 |
| | 1536x2304 | 9.366 | 6.796 |
| | 2048x3072 | 17.721 | 10.279 |
| | 2560x3840 | 26.270 | 12.604 |
| | 3072x4608 | 34.331 | 15.379 |
| | 3584x5276 | 49.059 | 21.874 |
| Kodak-14 | 512x768 | 1.255 | 2.873 |
| | 1024x1536 | 4.233 | 4.379 |
| | 1536x2304 | 10.453 | 7.236 |
| | 2048x3072 | 16.389 | 10.901 |
| | 2560x3840 | 24.113 | 11.816 |
| | 3072x4608 | 36.042 | 16.473 |
| | 3584x5276 | 48.692 | 19.937 |
| Kodak-21 | 512x768 | 1.129 | 2.315 |
| | 1024x1536 | 4.070 | 4.247 |
| | 1536x2304 | 9.416 | 8.033 |
| | 2048x3072 | 16.103 | 10.241 |
| | 2560x3840 | 25.179 | 12.936 |
| | 3072x4608 | 35.344 | 15.402 |
| | 3584x5276 | 48.39 | 22.072 |

La tabla 3 muestra la evaluación del sistema tomando en consideración el tiempo de procesamiento requerido para ejecutar el proceso serial y paralelo de la fase *detección de alteraciones*.

Tabla 3. Tiempos de procesamiento en segundos para el proceso de protección de la imagen

| Imagen | Tamaño | Tiempo de procesamiento (segundos) | |
|----------|-----------|------------------------------------|----------|
| | | Serial | Paralelo |
| Kodak-3 | 512x768 | 0.799 | 1.680 |
| | 1024x1536 | 2.352 | 2.575 |
| | 1536x2304 | 5.035 | 4.758 |
| | 2048x3072 | 9.071 | 7.531 |
| | 2560x3840 | 13.942 | 10.533 |
| | 3072x4608 | 19.671 | 12.861 |
| | 3584x5276 | 26.968 | 17.413 |
| Kodak-14 | 512x768 | 0.809 | 2.086 |
| | 1024x1536 | 2.659 | 2.434 |
| | 1536x2304 | 5.125 | 4.296 |
| | 2048x3072 | 8.984 | 6.922 |
| | 2560x3840 | 13.790 | 8.376 |
| | 3072x4608 | 19.723 | 13.850 |
| | 3584x5276 | 27.596 | 15.255 |
| Kodak-21 | 512x768 | 0.817 | 2.117 |
| | 1024x1536 | 2.379 | 2.756 |
| | 1536x2304 | 5.257 | 5.127 |
| | 2048x3072 | 9.103 | 6.971 |
| | 2560x3840 | 13.120 | 10.095 |
| | 3072x4608 | 20.184 | 13.769 |
| | 3584x5276 | 27.531 | 18.207 |

IV. DISCUSIÓN

Los resultados obtenidos en la tabla 1 muestran un PSNR mayor a 35 dB, el cual representa una calidad aceptable para la imagen marcada, es decir, la marca de agua insertada se puede considerar invisible al ojo humano. Se puede observar que al utilizar imágenes inferiores a 1 Megapíxel la calidad obtenida es mayor a diferencia de si se utilizan imágenes de un tamaño superior, pero la calidad no decrece de manera significativa, obteniéndose un valor mayor a 35 dB para imágenes superiores a 17 Megapíxeles.

Durante la etapa de detección de alteraciones, se observó que el sistema propuesto es robusto ante distintos tipos de ataques, entre los que se encuentran: inserción de información adicional, ataque de collage e intercambio de color. Así mismo se observó que el sistema propuesto es capaz de localizar las modificaciones intencionales sin detectar las alteraciones no intencionales. En el caso de compresión JPEG el sistema propuesto es capaz soportar compresión con un factor de calidad menor a 70, a pesar de que se tienen falsas detecciones estas son mínimas y podrían eliminarse con una fase de post-procesamiento a las imágenes de detección obtenidas.

Los resultados mostrados en la tabla 2 muestran la evaluación del tiempo de procesamiento requerido al realizar el proceso de *protección de la imagen* utilizando distintos tamaños de imágenes, se observó que es necesario utilizar imágenes superiores a 2 Megapíxeles para que exista una reducción del tiempo de procesamiento utilizando cómputo en paralelo en un equipo con 8GB de memoria RAM, Procesador Intel® Core™ i5-4200U 1.6GHz, con dos núcleos de procesamiento. Esto significa que al utilizar imágenes superiores a 2 Megapíxeles, el tiempo de procesamiento requerido para procesar las etapas “*División en sub-imágenes*” y “*Concatenación de imágenes*” y el tiempo requerido para el traslado de memoria entre cada procesador y la memoria RAM son inferiores al tiempo requerido para llevar a cabo el algoritmo “*Protección de la imagen*”. Se observó que al utilizar imágenes de superior tamaño, la tasa de reducción de tiempo de procesamiento comparando el tiempo de procesamiento serial y paralelo se incrementa, al utilizar imágenes de 3.5 Megapíxeles, existe una reducción del 5.5%, al utilizar imágenes de tamaño 6 Megapíxeles existe una reducción del 17%, al utilizar imágenes de tamaño 10 Megapíxeles existe una reducción del 24.5%, al utilizar imágenes de tamaño 14 Megapíxeles existe una reducción del 35%, finalmente al utilizar imágenes de tamaño 18 Megapíxeles existe una reducción del 45% del tiempo de procesamiento serial.

Los resultados mostrados en la tabla 3 se muestra la evaluación del tiempo de procesamiento requerido al realizar el proceso de *detección de alteraciones* utilizando distintos tamaños de imágenes, en este caso también es necesario utilizar imágenes superiores a 2 Megapíxeles para que exista una reducción del tiempo de procesamiento utilizando cómputo en paralelo en un equipo con 8GB de memoria RAM, Procesador Intel® Core™ i5-4200U 1.6GHz, con dos núcleos de procesamiento, al utilizar imágenes de 3.5 Megapíxeles, existe una reducción del 5.5%, al utilizar imágenes de tamaño 6

Megapíxeles existe una reducción del 23%, al utilizar imágenes de tamaño 10 Megapíxeles existe una reducción del 25%, al utilizar imágenes de tamaño 14 Megapíxeles existe una reducción del 31%, finalmente al utilizar imágenes de tamaño 18 Megapíxeles existe una reducción del 33% del tiempo de procesamiento serial.

V. CONCLUSIONES

En el presente trabajo se propone un algoritmo para la detección de alteraciones en imágenes digitales, utilizando un sistema de marcado de agua semi-frágil basado en la Transformada Discreta de Wavelet, mediante el filtro *Daubechies 4*. Se realizaron pruebas de calidad para las imágenes después de insertar la marca de agua, para estas pruebas se utilizó la métrica de calidad objetiva PSNR, donde se demostró que la marca de agua insertada permanece invisible al Sistema Visual Humano, obteniéndose valores de PSNR superiores a 35 dB.

Este sistema demostró ser eficiente para detectar alteraciones intencionales y minimizando las falsas detecciones de alteraciones no intencionales como compresión JPEG. Se realizaron distintos ataques intencionales: adición de información visual en la imagen digital, ataque de collage e intercambio de color, las imágenes utilizadas después de realizar los ataques fueron almacenadas en el formato de compresión JPEG, utilizando distintos factores de calidad (100, 90, 80 y 70), donde el sistema propuesto demostró una correcta detección de alteraciones para cada prueba realizada.

Finalmente, este sistema fue implementado mediante cómputo en paralelo, sobre equipos con procesador multi-núcleo. Donde se demostró un buen desempeño para ambas etapas *protección de la imagen* y *detección de alteraciones*. Se obtuvo una reducción de tiempo de procesamiento al ejecutar el sistema mediante cómputo en paralelo a comparación de la ejecución serial. Donde se obtuvo una reducción de hasta el 45% el tiempo total de procesamiento empleando imágenes de 18 Megapíxeles.

Como trabajo a futuro se tiene aplicar una fase de post-procesamiento a la etapa de *detección de alteraciones* para eliminar las falsas detecciones generadas mediante ataques no intencionales.

AGRADECIMIENTOS

Agradecemos al Instituto Politécnico Nacional, al Consejo Nacional de Ciencia y Tecnología de México, a la Comisión de Operación y Fomento de Actividades Académicas (COFAA) del IPN y a la Beca de Estimulo Institucional de Formación de Investigadores (BEIFI) del IPN por el apoyo otorgado para el desarrollo de este trabajo.

REFERENCIAS

- [1] Kiatpapan, S., Kondo, T. (2015) An image tamper detection and recovery method based on self-embedding dual watermarking. Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON). Hua Hin Thailand.

- [2] Yuhang Li, Ling Du. (2014) *Semi-fragile watermarking for image tamper localization and self-recovery*. Security, Pattern Analysis, and Cybernetics (SPAC). pp. 328-333.
- [3] Sajjad Dadkhah, Azizah Abd Manaf, Somayeh Sadeghi. (2014) *An Efficient Image Self-recovery and Tamper Detection Using Fragile Watermarking*. 11th International Conference, ICIAR 2014, Vilamoura, Portugal. pp. 504-513.
- [4] Tom, E., Premanand, B. (2013) *Watermarking for self-recovery of tampered images using compressed sensing*. Control Communication and Computing (ICCC). pp. 479 – 484.
- [5] Y.-F. Chang y W.-L. Tai. (2013) *A block-based watermarking scheme for image tamper detection and self-recovery*. Opto-Electronics Review. Volume 21, Issue 2, pp. 182-190.
- [6] K. R. Chetan y S. Nirmala (2017). *An intelligent fragile watermarking scheme based on contourlets for effective detection, localization and recovery of tampered regions in handwritten document images*. International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 405-410.
- [7] Musaed Alhussein (2016). *Image Tampering Detection Based on Local Texture Descriptor and Extreme Learning Machine*. UKSim-AMSS 18th International Conference on Computer Modelling and Simulation (UKSim) pp- 196-199.
- [8] Tushar D. Gadhiya, Anil K. Roy, Suman K. Mitra y Vinod Mall (2017). *Use of discrete wavelet transform method for detection and localization of tampering in a digital medical image*. IEEE Region 10 Symposium (TENSymp) pp 1-5.
- [9] V. Thirunavukkarasu y J. Satheesh Kumar (2016) *Passive image tamper detection based on Fast Retina Key Point Descriptor*. IEEE International Conference on Advances in Computer Applications (ICACA) pp. 279-285.
- [10] Fu D, Shi Y Q y Su W. A (2007) *generalized Benford's law for JPEG coefficients and its applications in image forensics*. Electronic Imaging 2007. International Society for Optics and Photonics, pp. 65051L-65051L-11.
- [11] Benford F. (1938). *The law of anomalous numbers*. Proceedings of the American Philosophical Society, pp.551-572.

Marca de Agua Visible en Imágenes Digitales a Color utilizando la Distorsión Apenas Perceptible

Franco Saavedra R., Rogelio Reyes R., Clara Cruz R., Volodymyr Ponomaryov, Javier Molina G.

Instituto Politécnico Nacional, ESIME Unidad Culhuacán
Av. Santa Ana No. 1000, Col. San Francisco Culhuacán, CP. 04430, México, Ciudad de México.
Teléfono (55) 5729-6000 Ext. 73264 Fax (55) 56562058
email: fsaavedrar1200@alumno.ipn.mx, javier.molina.21016@gmail.com

Resumen—Actualmente los sistemas de marcado de agua cumplen con los requerimientos de imperceptibilidad y robustez antes diversos ataques, sin embargo, es una tarea difícil lograr la mayor imperceptibilidad y robustez al mismo tiempo. Por lo tanto, es necesario conocer la distorsión mínima que puede aplicarse a una imagen digital, sin que su contenido visual se perciba alterado. La Distorsión Apenas Perceptible (Just Noticeable Distorsion – JND) se refiere al umbral de visibilidad por debajo del cual cualquier cambio en una imagen digital no podrá ser detectado por el Sistema Visual Humano (Human Visual System - HVS). En este artículo se presenta una propuesta de marcado de agua visible en imágenes digitales obteniendo una zona de distorsión mínima por medio del modelo JND. Los resultados obtenidos muestran como la combinación del modelo JND con una técnica de marcado de agua visible ayuda a mejorar el equilibrio entre robustez e imperceptibilidad en la imagen marcada.

Palabras Clave- Sistema Visual Humano, Distorsión Apenas Perceptible, Adaptación de Luminancia, Función de Sensibilidad de Contraste, Clasificación de Texturas, Marca de Agua Visible.

I. INTRODUCCIÓN

El marcado de agua en imágenes digitales emergió como una solución a la problemática de la protección de derechos de autor [1]; ya que hoy en día una gran cantidad de personas alrededor del mundo comparten sus fotografías e imágenes digitales a través de Internet, debido a que éstas pueden ser visualizadas y distribuidas por cualquier persona en todo momento. Se considera de suma importancia la protección intelectual del contenido de estos archivos, el marcado de agua en imágenes digitales proporciona bondades como la protección, autenticación e integridad de los datos añadiendo información adicional a la imagen original teniendo como resultado una imagen protegida. El desarrollo de aplicaciones multimedia para la seguridad y protección de propiedad intelectual se ha convertido en un aspecto importante, propiciando el desarrollo de distintas técnicas de marcado de agua que se pueden aplicar en imágenes digitales, estas técnicas pueden clasificarse en dos categorías: técnicas de marcado de agua en el dominio espacial y técnicas de marcado de agua en el dominio frecuencial [2].

Aunque uno de los principales problemas de las técnicas de marcado de agua ha sido la imperceptibilidad de la misma, ya que degrada la calidad de la imagen original; teniendo en cuenta que de acuerdo al factor de inserción es la perceptibilidad de la marca de agua y la robustez de esta. Se ha considerado la implementación de técnicas de procesado de imagen, buscando una región adecuada de la imagen huésped para que la marca de agua no sea intrusiva. Una de las principales técnicas usadas es el modelo JND, el cual durante los últimos años ha sido ampliamente utilizado para el procesamiento de imágenes digitales. El JND es empleado para detectar los niveles de distorsión máximos y mínimos en ciertas regiones de la imagen, los cuales no pueden ser distinguidos por el ojo humano; su funcionamiento está basado en procesar las características de una imagen, entre las cuales se encuentran la luminancia, las zonas texturizadas, el contraste, los colores o la profundidad de la misma [3].

El modelo JND se ha implementado en múltiples algoritmos computacionales para el procesamiento y manipulación de imágenes con la finalidad de poder emular el comportamiento del HVS, lo cual ha generado varias bondades dentro de este campo entre las cuales se pueden destacar: el mejorar la capacidad de procesamiento en una imagen digital y sus características, obteniendo mejores resultados con respecto a los algoritmos que no consideraban el comportamiento del HVS.

En el presente artículo, se propone un algoritmo adaptivo de marcado de agua visible en las regiones determinadas por el modelo JND. El artículo está organizado de la siguiente manera: en la sección dos se presentan las características de una marca de agua digital, la sección tres describe el sistema propuesto, la sección cuatro presenta las pruebas y resultados obtenidos, por último, se presentan las conclusiones obtenidas en el proceso de desarrollo del proyecto.

II. MARCA DE AGUA DIGITAL

Una marca de agua consiste en la inserción de información adicional de una imagen marca (W) dentro de una imagen huésped (I) con la finalidad de proteger los derechos de autor

y la seguridad del contenido, donde el proceso de inserción de la marca toma la imagen (I) y la marca (W) y genera la imagen marcada (IW). Como se mencionó anteriormente la imagen original puede ser transformada al dominio frecuencial o puede ser utilizada en el dominio espacial.

A. Características de la Marca de Agua

Las principales características que una marca debe de cumplir son [4]:

- Imperceptibilidad: que la calidad de la imagen no se debe ver afectada por la presencia de la imagen marca.
- Robustez: esta característica hace referencia a la resistencia que tiene la marca de agua contra ataques para ser removida de la imagen huésped.
- Capacidad: que es la cantidad de información que puede ser insertada en la imagen huésped.

B. Usos de la Marca de Agua

El proceso de marcado de agua digital se puede realizar en distintos tipos de archivos digitales como: audio, imágenes, documentos, etc. Los principales usos de un marcado de agua de acuerdo a [2] son:

- Seguimiento de la fuente: el contenido marcado se puede enviar a distintos destinatarios.
- Vigilancia de emisiones: las marcas de agua insertadas en imágenes o videos que se transmiten pueden ser detectadas.
- Autenticación de Control de Copia: previene la realización de copias no autorizadas protegidas por derechos de autor.
- Identificación del Propietario: establece la propiedad del contenido protegido.

- Huella Dactilar: el contenido que sea ilegalmente duplicado o distribuido puede ser rastreado.

C. Tipos de Inserción de Marca de Agua

Existen diferentes algoritmos para la inserción de marca digital, como se menciona en [2] los cuales son:

- Inserción Aditiva: la marca de agua es sumada al archivo original.
- Inserción Multiplicativa e inserción por cuantización
- Inserción en diferentes dominios: la imagen original es marcada por medio de transformaciones de dominios, como, por ejemplo, Transformada Discreta de Fourier (DFT), Transformada Discreta de Coseno (DCT), y la Transformada Wavelet (WT).

III. SISTEMA PROPUESTO

Se propone un sistema para la inserción de una marca de agua visible en una imagen digital, analizando la imagen huésped mediante el modelo JND, obteniendo las zonas optimas donde realizar inserción, con la finalidad de que el HVS no note alteración en la imagen marcada con respecto a la imagen original.

En la Fig. 1 se muestra el diagrama a bloques general del sistema propuesto, el cual se compone de dos módulos: el primero es el cálculo y el análisis del JND de la imagen original y el segundo la inserción de la marca de agua obteniendo como resultado la imagen marcada.

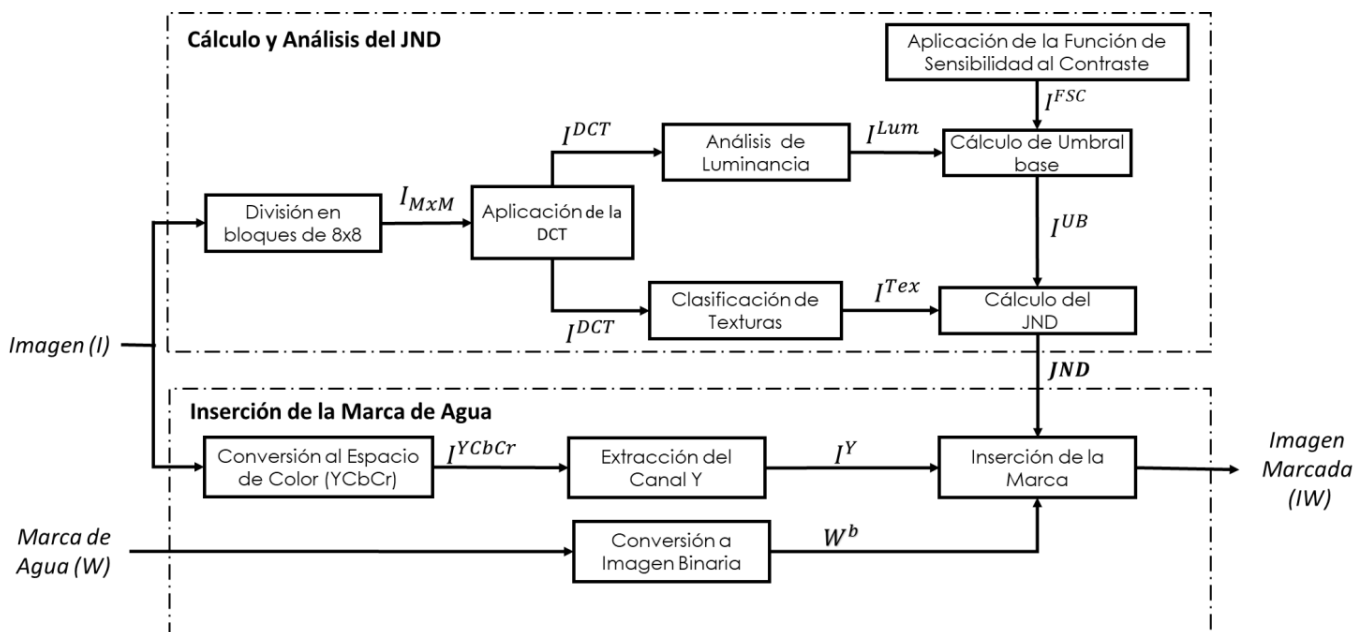


Fig. 1. Diagrama a Bloques del Sistema Propuesto

A. Cálculo y Análisis del JND

En este modelo, la imagen portadora es dividida en bloques no traslapados de 8x8 píxeles a los que posteriormente se aplicara la DCT en cada uno de los bloques de la imagen para la obtención de 64 coeficientes DCT, divididos en un coeficiente DC y 63 coeficientes AC; clasificados en Altas Frecuencias (HF), Medias Frecuencias (MF) y Bajas Frecuencias (LF) como se muestra en la Fig. 2

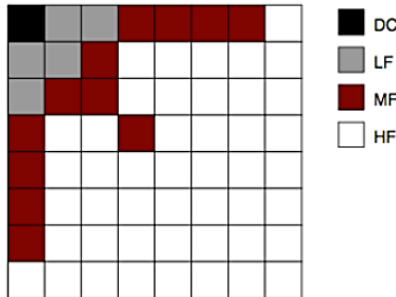


Fig. 2. Clasificación de los coeficientes DCT

La Adaptación de Luminancia permite calcular las zonas de luminosidad perceptibles en una imagen digital mediante el uso de los coeficientes DC de los bloques no traslapados de la imagen portadora, como se muestra en (1).

$$I_m^{Lum} = \begin{cases} k_1 \left(-\frac{2DC_m}{I_m N} \right)^{\lambda_1} + 1, & 2DC_m \leq \frac{I_m N}{2} \\ k_2 \left(\frac{2DC_m}{I_m N} - 1 \right)^{\lambda_2} + 1, & \text{De otra manera,} \end{cases} \quad (1)$$

donde $k_1 = 2$ y $k_2 = 0.8$ son valores definidos en [5]; $\lambda_1 = 3$ y $\lambda_2 = 2$ representan la distancia de visualización, I_m indica el valor de luminosidad del m -ésimo píxel de la imagen, DC_m representa el coeficiente DC del m -ésimo bloque de la imagen y N representa el tamaño del bloque analizado.

La Aplicación de la Función de Sensibilidad al Contraste representa el efecto de la frecuencia espacial en la sensibilidad del HVS, donde el umbral de visibilidad es una función de la luminosancia de fondo L y las frecuencias espaciales f . Este umbral mínimo I_m^{FSC} se describe en (2).

$$I_m^{FSC} = \frac{I_m}{N(I_{max}^L - I_{min}^L)} f(i, j), \quad (2)$$

donde I_{max}^L y I_{min}^L representan el valor de luminosancia máximo y mínimo de los píxeles de la imagen respectivamente; I_m es el valor de luminosancia del m -ésimo píxel de la imagen (0-255), N representa el tamaño del bloque analizado y $f(i, j)$ representa las frecuencias espaciales, establecidas en 6 ciclos por segundo debido a que el HVS percibe mejor los cambios en este número de ciclos por segundo [6].

A partir del Análisis de la Luminancia y la Aplicación de la Función de Sensibilidad al Contraste, se calcula el Umbral Base donde se combinan los valores obtenidos previamente, mediante (3).

$$I_m^{UB} = I_m^{Lum} \cdot I_m^{FSC} \quad (3)$$

Otra característica evaluada en el sistema propuesto es la textura de la imagen digital, a través de la Clasificación de Texturas de la imagen, utilizando (4).

$$I_m^{Tex} \begin{cases} \text{Caso 1: } TexE \leq \mu_1 & \text{Bloque Plano,} \\ \text{Caso 2: } \mu_1 < TexE \leq \mu_2 & \text{Bloque Borde,} \\ \text{Caso 3: } \mu_2 < TexE \leq \mu_3 & \text{Bloque Borde,} \\ \text{Caso 4: } TexE > \mu_3 & \text{Bloque Textura.} \end{cases}$$

$$TexE_m = \sum MF_m + \sum HF_m, \quad (4)$$

donde MF_m y HF_m representan los coeficientes AC de medias y altas frecuencias del m -ésimo bloque de la imagen, $\mu_1 = 125$, $\mu_2 = 290$ y $\mu_3 = 900$ son rangos de frecuencia para determinar el tipo de bloque de la imagen definidos en [7], y $TexE_m$ es la Energía de la Textura del m -ésimo bloque de la imagen.

Una vez obtenidos los valores del Umbral Base y de la Clasificación de Texturas se realiza el cálculo del modelo JND, obteniendo los valores máximos y mínimos de distorsión aplicables a la imagen mediante (5).

$$JND_m = I_m^{UB} * I_m^{Tex} \quad (5)$$

B. Inserción de la Marca de agua

La imagen portadora se convertirá del espacio de color RGB al espacio de color YCbCr, para la utilización del plano Y de la imagen en el cual se realizará la inserción de la marca de agua visible. De igual manera, la imagen marca será sometida a un preprocesamiento para convertirla a imagen binaria.

Los valores JND obtenidos en (5), serán interpolados en el rango de 0.07 a 0.5. Estos nuevos valores del JND determinaran el factor de inserción de la marca de agua. Teniendo el plano Y de la imagen portadora, la imagen marca en binario y los valores JND interpolados se procede a la inserción de la marca de agua, mediante (6).

$$IW = I_m * (1 - JND_m^i) + (W_m * JND_m^i), \quad (6)$$

donde I_m representa el valor de luminosancia del m -ésimo píxel de la imagen portadora, JND_m^i indica el valor interpolado del JND del m -ésimo píxel de la imagen y W_m representa el valor de luminosancia del m -ésimo píxel de la imagen marca.

IV. PRUEBAS Y RESULTADOS

Para la realización de las pruebas del sistema propuesto se utilizaron las imágenes obtenidas de la base de datos USC-SIPI [8] con una resolución de 512 x 512 píxeles, en formato .tiff, las cuales son comúnmente utilizadas en el área de procesamiento digital de imágenes e imágenes capturadas por los autores del presente trabajo. Se utilizaron dos imágenes como marca de agua a color en formato .jpeg con una resolución 169x169 píxeles y 235x175 píxeles respectivamente las cuales se muestran en la Fig. 3.



Fig. 3. Imágenes Marca

Las imágenes mostradas en la Fig. 4, son el resultado del análisis del Umbral de Distorsión Apenas Perceptible en la cual se muestra la imagen original y los valores de JND obtenidos de cada imagen respectivamente.

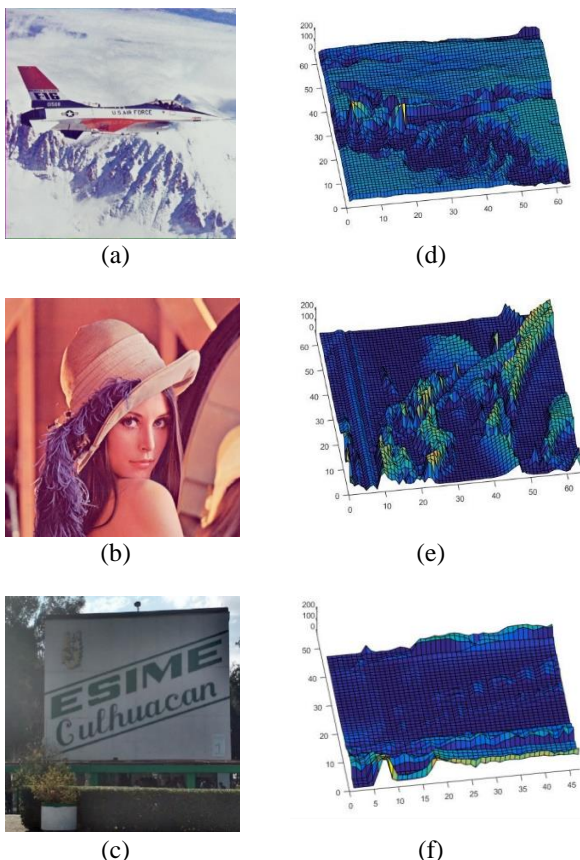


Fig. 4. Resultados de la obtención del modelo JND: (a-c) Imagen Original, (d-f) Umbral de Distorsión Apenas Perceptible (JND)

Como se observa en la Fig. 4(d-f) las regiones del Umbral de Distorsión Apenas Perceptible con tonalidades azules representan las regiones de la imagen en las cuales cualquier cambio o alteración en la intensidad del píxel será altamente perceptible por el HVS, mientras que las regiones de tonalidades amarillas indican las regiones de la imagen en las cuales un cambio en la intensidad del píxel no es perceptible por el HVS, lo que permite realizar cambios sustanciales en la intensidad en estas regiones sin que se degrade la calidad de la imagen. Los valores del umbral de distorsión son determinados con base a las características analizadas de la imagen propuestas en el sistema: luminancia, contraste y textura. En el cual las regiones de la imagen que cuenten con altos niveles de luminosidad o zonas muy texturizadas resultaran en un valor de distorsión mayor permitido, esto debido a que en este tipo de regiones la perceptibilidad el ojo humano es menor ante alguna alteración, en caso contrario en zonas de la imagen donde se puede encontrar zonas planas o de una intensidad de luminosidad media el valor resultante del umbral determina un rango menor de distorsión permitido en el cual la perceptibilidad de cualquier cambio sería más notable ante el HVS.

Las Fig. 5, 6 y 7 muestran la aplicación del algoritmo propuesto, donde se realizaron las pruebas de inserción de la marca de agua visible con las imágenes mostradas en la Fig. 3 tomando en consideración como factor de inserción los valores del umbral de distorsión apenas perceptible interpolados en un rango de 0.07 a 0.5.

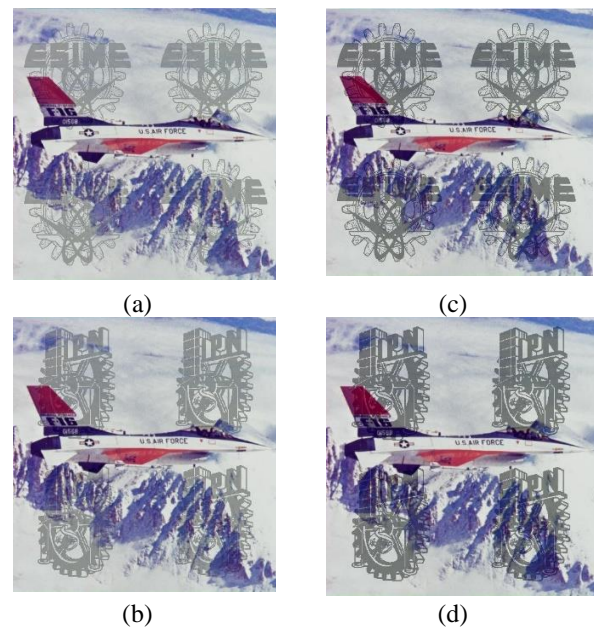


Fig. 5. Resultados de la inserción de la marca de agua visible: (a-b) Imagen marcada con el sistema propuesto, (c-d) Imagen marcada con factor de inserción homogéneo $\alpha = 0.3$



Fig. 6. Resultados de la inserción de la marca de agua visible: (a-b) Imagen marcada con el sistema propuesto, (c-d) Imagen marcada con factor de inserción homogéneo $\alpha=0.3$

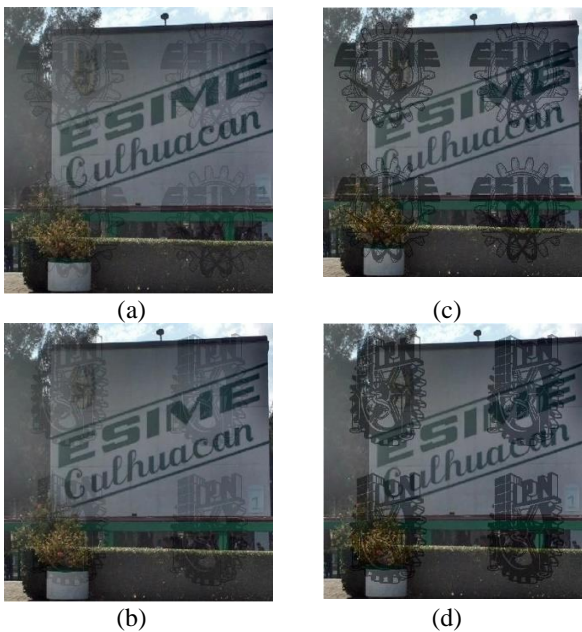


Fig. 7. Resultados de la inserción de la marca de agua visible: (a-b) Imagen marcada con el sistema propuesto, (c-d) Imagen marcada con factor de inserción homogéneo $\alpha=0.3$

En las Fig. 5, 6 y 7 se presenta una comparación entre las imágenes marcadas con el sistema propuesto considerando el umbral de distorsión apenas perceptible y las imágenes marcadas con un factor de inserción homogéneo. Se observa que al realizar la inserción de la marca de agua visible con un factor de inserción homogéneo la calidad de la imagen a proteger se ve afectada por la propia marca de agua visible degradando su calidad, mientras que al insertar la marca de agua visible considerando el umbral de distorsión apenas

perceptible de la imagen a proteger, se puede adaptar el factor de inserción de la imagen marca para que esta sea perceptible ante el HVS pero sin que la calidad de la imagen protegida se vea degradada

V. CONCLUSIONES

El análisis de la textura, la luminancia y el contraste de la imagen digital por medio del modelo JND para obtener este umbral de distorsión, contribuye a que la inserción de una marca de agua visible guarde un equilibrio entre la robustez y la perceptibilidad; de igual manera al tomar en consideración más características de la imagen a proteger tales como el color, mejoraría el cálculo del umbral de distorsión. Se puede concluir que el método propuesto cumple con el objetivo de insertar la marca de agua visible sin que esta degrade la calidad de la imagen a proteger y manteniendo su perceptibilidad ante el ojo humano comparado con un método de inserción aditiva en el cual como se observa la calidad visual de la imagen marcada se ve gravemente deteriorada.

AGRADECIMIENTOS

Al Instituto Politécnico Nacional, al Consejo Nacional de Ciencia y Tecnología y a la Comisión de Operación y Fomento de Actividades Académicas, por su apoyo y patrocinio para la realización de este trabajo.

REFERENCIAS

- [1] Agarwal H., Sen D., Raman B. & Kankanhalli M., "Visible watermarking based on importance and just noticeable distortion of image regions", in Springer Science Business Media New York 2015
- [2] Amit K., Mayank D and Anand M., "A novel Technique for Digital Image Watermarking in Frecuency Domain", IEEE International Conference on Parallel, Distributed and Grid Computing, 2012
- [3] C. A. Bouman, "The visual perception of Images", Digital Image Processing, January 12, 2015
- [4] Urvi H. & Srivastava R., "A Comprehensive Survey on Digital Image Watermarking Techniques," 2015 Fifth International Conference on Communication Systems and Network Technologies, Gwalior, 2015, pp. 591-595.
- [5] Yuthing Jia. "Just Noticeable Distortion Model and Its Application in Image Processing". 2005, M.E. thesis, Beijing, Universidad Nacional de Singapore.
- [6] Ahumada, Albert J., and Heidi A. Peterson. "Luminance-model-based DCT quantization for color image compression." Human vision, visual processing, and digital display III. Vol. 1666. International Society for Optics and Photonics, 1992.
- [7] Zhang X.H. & Lin W.S. & Xue P., Improved estimation for just-noticeable visual distortion, Signal Processing, Volume 85, Issue 4, April 2005
- [8] USC-SIPI. (1981). The USC-SIPI Image Database. Southern California. Recuperado de: <http://www.sipi.usc.edu/database/>

Pseudocoloración y Segmentación de Tomografías Axiales Computarizadas CT Cráneo-Cerebrales Mediante Operador de Gradiente y Morfológico

Aridai Solís S., Clara Cruz R., Rogelio Reyes R., Volodymyr Ponomaryov

Instituto Politécnico Nacional, ESIME Unidad Culhuacán

Av. Santa Ana No. 1000, Col. San Francisco Culhuacán, CP. 04430, México, CDMX.

Tel. (55) 5729-6000 Ext. 73264 Fax (55) 56562058 e-mail: asoliss1401@alumno.ipn.mx, rreyesre@ipn.mx, vponomar@ipn.mx

Resumen- El presente trabajo analiza, describe y detalla el procesamiento utilizado para la pseudocoloración de regiones de interés en tomografías computarizadas (CT) Cráneo-Cerebrales mediante la segmentación, el uso de este tipo de imágenes médicas se ha incrementado drásticamente en las últimas dos décadas, debido a que hoy en día es común que en el sector salud se cuente con equipo especializado para la evaluación diagnóstica. El operador morfológico que se analiza a detalle durante este trabajo es erosión binaria, así como el operador de gradiente de Prewitt; los cuales son aplicados en la CT con el objetivo de encontrar los bordes de la imagen para posteriormente realizar una pseudocoloración, la cual ayudará a distinguir los distintos componentes que existen en una CT y que no son perceptibles para el humano debido a una alta escala de grises. Los resultados obtenidos demuestran la viabilidad y la correcta segmentación de las CT utilizando el método propuesto.

Palabras Clave: Tomografía Computarizada (CT), Segmentación de imágenes (IS), Operadores Morfológicos, Gradiente, Procesamiento digital de imágenes.

I. INTRODUCCIÓN

La tomografía computarizada (CT) cráneo-cerebral es el estudio radiológico de primera elección en el paciente para la evaluación diagnóstica asociada al trauma. La CT permite la detección rápida de lesiones en la bóveda craneal y el encéfalo, además es una herramienta invaluable en la guía del tratamiento [1]. Esto es posible gracias a una eficiente detección de anomalías mediante el procesamiento de las CT basado en técnicas de segmentación de imagen.

La segmentación de imagen es un extenso y activo campo no solamente en el procesamiento digital de imágenes médicas, sino también en áreas como la visión artificial e imagen satelital [2].

El propósito principal de la segmentación es dividir una imagen en regiones las cuales son significativas para una tarea en particular. Varios métodos y enfoques son usados para llevarla a cabo; la elección de un método en particular depende de las características del problema a resolver y su lugar en una estrategia de análisis de imagen más amplia.

Para desarrollar sistemas de interpretación robustos, es importante utilizar tanta información a priori como sea posible durante la segmentación.

II. FUNDAMENTOS

A. Segmentación

Segmentación es la partición de una imagen en regiones significativas para distinguir objetos o regiones de interés ("primer plano") de todo lo demás ("fondo"). En los casos más simples, solo habría estas dos clases (primer plano y fondo) y la imagen segmentada sería una imagen binaria [3].

La segmentación se utiliza como ejemplo: para la detección de órganos, como el cerebro, el corazón, los pulmones o el hígado en imágenes de CT o MR (*resonancia magnética*); para distinguir el tejido patológico, como un tumor, de un tejido normal; y en la planificación del tratamiento. Se puede agregar pseudocolor a la imagen original en función de la extensión de las regiones segmentadas.

B. Atributos básicos en la segmentación

El atributo más básico utilizado para definir las regiones es el nivel de gris de la imagen o el brillo, pero se pueden usar otras propiedades como el color o la textura. La segmentación es la primera etapa en los sistemas de reconocimiento de patrones; una vez que los objetos de interés están aislados del resto de la imagen, se pueden realizar ciertas medidas de caracterización (extracción de características), y esto podría usarse para clasificar los objetos en grupos o clases particulares [2].

C. Técnicas de segmentación

Hay muchos enfoques de segmentación, y se pueden clasificar según la característica y la técnica utilizada. Las características incluyen valores de gris (brillo), textura y magnitudes de gradiente [2].

III. OPERADORES MORFOLÓGICOS

Las tareas de segmentación no suelen dar un resultado exacto de la delimitación de los objetos o regiones de interés. Aparecen píxeles mal clasificados, bordes imprecisos de los objetos o regiones que están solapadas. Por tanto, antes de extraer más características de medio nivel se requiere de una etapa de pre-procesamiento. En esta fase se suele emplear el tratamiento morfológico. Es una técnica de procesamiento no lineal de la señal, caracterizada en realzar la geometría y forma de los objetos [4].

A. Erosión y Dilatación Binaria

La transformación de la erosión es el resultado de comprobar si el elemento estructurante B está completamente incluido dentro del conjunto X. Cuando no ocurre, el resultado de la erosión es el conjunto vacío:

$$\varepsilon_B(X) = X \ominus B = \{x \mid B_x \subseteq X\} \quad (1)$$

Cuando los objetos de la escena sean menores que el elemento estructurante, éstos desaparecerán. Su utilidad consiste en definir una geometría determinada al elemento estructurante y pasarlo sobre la imagen [4]. Por tanto, la erosión supone una degradación de la imagen (Fig. 1). La aplicación iterativa de esta transformación hará que se eliminen todos los objetos existentes en la imagen. La erosión es una transformación antiextensiva:

$$\varepsilon_B(X) \subseteq X \quad (2)$$

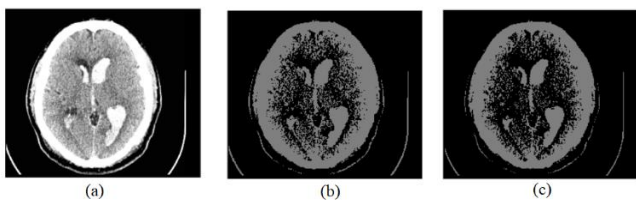


Fig. 1. (a) Imagen original de un corte de CT, (b) imagen umbralizada, (c) imagen aplicando erosión con un elemento estructurante de disco.

B. Procesamiento Morfológico en Niveles de Grises

La operación de erosión es creciente, respeta el orden presente en la estructura del conjunto. Para dos imágenes f y g:

$$\begin{aligned} \text{Si } f \leq g &\Rightarrow \varepsilon(f) \leq \varepsilon(g) \\ \text{Si } f \leq g &\Rightarrow \delta(f) \leq \delta(g) \end{aligned} \quad (5)$$

Gracias a esta propiedad, esta operación morfológica puede extenderse del caso binario a la escala de grises [4], puesto que estas últimas señales pueden expresarse como una suma ponderada de imágenes binarias.

La erosión de una función en un punto será el valor mínimo de la función dentro de la ventana definida por el elemento estructurante B, cuando el origen de B esté situado en dicho punto [5].

Retornará el valor máximo de la función definido por el elemento estructurante (Fig. 2):

$$\begin{aligned} \varepsilon_B(f) &= f \ominus B = \min(f(x-y), y \in B) \\ \delta_B(f) &= f \oplus B = \max(f(x-y), y \in B) \end{aligned} \quad (6)$$

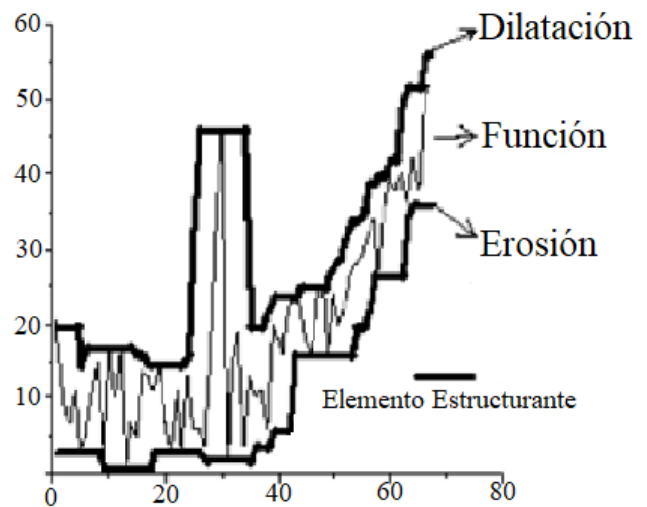


Fig. 2. El resultado de la erosión sobre imágenes en niveles de grises hace que la imagen de salida sea más oscura. La imagen resultante de la dilatación es una imagen más clara [5].

C. Cierre

La dilatación binaria opera engrandando los objetos, cerrando los agujeros y las grietas. El ensanchamiento de los objetos puede ser reducido mediante la aplicación seguida de una erosión. La operación combinada de dilatación y erosión es llamada cierre (*closing*):

$$\varphi_B(X) = X \cdot B = \varepsilon_B(\delta_B(X)) \quad (8)$$

IV. OPERADORES DE GRADIENTE

El gradiente de una imagen $f(x, y)$ en el lugar (x, y) es el vector:

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (9)$$

El vector de gradiente apunta en la dirección de la velocidad máxima de cambio de f en (x, y) [9]. En la detección de bordes, una cantidad importante es la magnitud de este vector:

$$\begin{aligned} \nabla f &= \text{mag}(\nabla f) \\ &= \sqrt{G_x^2 + G_y^2} \\ &= \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \end{aligned} \quad (10)$$

El gradiente toma su tasa máxima de incremento de $f(x, y)$ por unidad de distancia en la dirección de ∇f . La magnitud del gradiente es comúnmente aproximada por:

$$|\nabla f| = |G_x| + |G_y| \quad (11)$$

Esto es más simple de implementar. La dirección del gradiente vector también es importante y está dado por:

$$\alpha(x, y) = \tan^{-1}\left(\frac{G_x}{G_y}\right) \quad (12)$$

Las técnicas comúnmente utilizadas para la detección de bordes y que además contemplan la reducción de ruido, así como el hallazgo de la intensidad del gradiente de la imagen [8] son: Operador de gradiente Prewitt y Operador de gradiente Canny

V. PROCEDIMIENTO DE PSEUDOCOLORACIÓN EN UNA TOMOGRAFÍA COMPUTARIZADA

Pseudocoloración significa "coloración falsa" [6]. En el procesamiento de imágenes la técnica de Pseudocoloración consiste en asignar colores a valores grises basados en un criterio específico. El término "Pseudocolor" enfatiza que los colores fueron asignados oponiéndose artificialmente a los verdaderos colores. El uso principal de la pseudocoloración es para visualización e interpretación humana de detalles de escala de grises en

una imagen o su secuencia. El principal objetivo de la pseudocoloración es obtener un color ergonómico que permita hacer una representación visual más sencilla de los datos, para facilitar su reconocimiento por un humano con vista normal [6].

El procedimiento de pseudocoloración para la imagen médica comienza con una umbralización que nos auxiliará en la segmentación de la imagen; esto nos ayudará a la reducción de ruido en aquellas áreas rugosas que se puedan presentar dentro de la masa intracraneal, se propone utilizar dos niveles de umbral ($T_{\text{mín}}$ y $T_{\text{máx}}$); para poder dar paso a la implementación con operadores morfológicos es necesario tener antes un elemento estructurante para tener un indicio de las formas que obtendremos dicho parámetro estará determinado y dependerá mucho de la región a detectar y/o corregir, los cuerpos que se pretenden detectar en este estudio son hematomas, es decir, sangre acumulada dentro de la masa intracraneal; procedemos a aplicar el operador de erosión para mejorar aún más aquellas regiones que no sean tan claras a simple vista, para la pseudocoloración es necesario calcular los bordes de la imagen procesada para esto se le aplica el gradiente de Prewitt a la imagen ya que este muestra como la imagen cambia abruptamente o suavemente permitiendo así el mejor cálculo de la dirección, con el mejoramiento de estos bordes y el umbralizado se puede realizar finalmente la pseudocoloración. Este procedimiento se puede explicar fácilmente con la ayuda de diagrama de bloques como se muestra en la Fig.3.

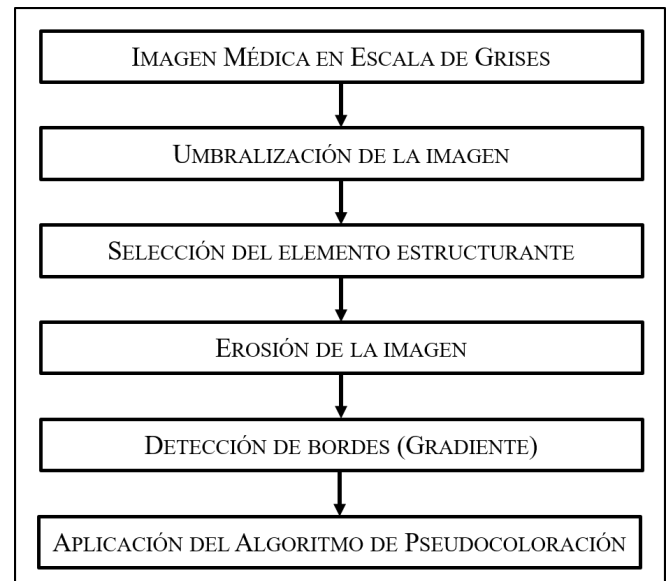


Fig. 3. Procedimiento de pseudocoloración mediante detección de bordes y operadores morfológicos.

VI. RESULTADOS EXPERIMENTALES

El desarrollo, implementación y análisis del algoritmo fue llevado a cabo en el software comercial de Matlab 2015a. En la implementación se pueden leer archivos DICOM que es el formato estándar para el uso de imágenes médicas.

La base de datos contiene 11 estudios axiales CT de cráneo, para las pruebas realizadas se contempló una CT cuyo paciente presenta hematomas parenquimatosos de acuerdo al diagnóstico de un médico radiólogo especializado; las regiones en las que se presenta la patología antes mencionada se pseudocolorea de un color más pronunciado y diferente al resto de la demás masa intracraneal sana, en particular la tomografía consiste en un conjunto de 129 cortes DICOM de 512x512 píxeles, cada corte con un grosor de 0.8mm.

Para la técnica de umbralización se utilizan dos niveles del umbral, un máximo y un mínimo, estos valores dependen de cada tomografía y de los metadatos que contenga, para este ejemplo se utilizaron valores de $T_{\min}=80$ y $T_{\max}=190$ basados en la escala de grises.

Se trabajó con el operador de erosión ya que es el que separa regiones débilmente unidas y permite su distinción, misma que ayuda a la distinción de hematomas (sangre acumulada) en regiones ruidosas, para dicho operador morfológico se utilizó un elemento estructural de disco con radio 3. La detección de bordes se realizó mediante la técnica de Prewitt, por ende, se hizo con una vecindad de 3x3 mismo que permitió definir cambios horizontales y verticales en la imagen, se utilizó la herramienta de image processing toolbox (Matlab) como ayuda para la visualización de la pseudocoloración.

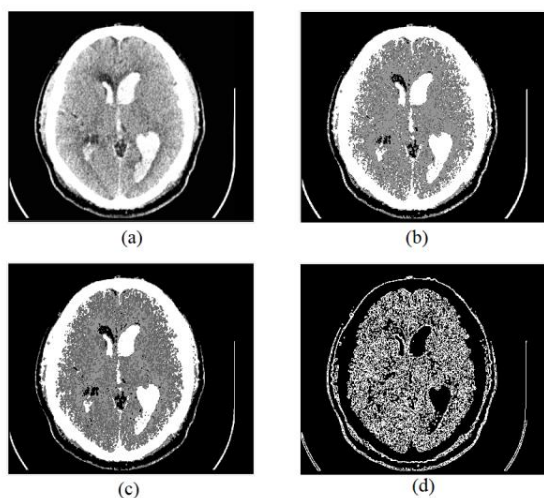


Fig. 4. (a) Imagen CT original, (b) Umbralización con T_{\min} y T_{\max} , (c) Erosión con elemento estructurante de disco y radio 1, (d) Detección de bordes mediante método Prewitt.

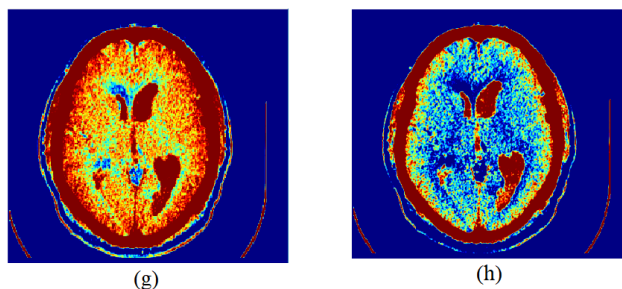


Fig. 4. (g) Pseudocoloración de corte de CT antes de la aplicación del operador morfológico, (h) Pseudocoloración de corte de CT después de la aplicación del operador morfológico (erosión con elemento estructurante de disco con radio 1).

VII. CONCLUSIONES

Para la detección de áreas de interés en este tipo de imágenes hace falta más que una simple binarización o bien, una umbralización, es importante mezclar siempre distintas técnicas que permitan mejorar dichas áreas de interés; para el caso de los operadores morfológicos la forma y la extensión viene definida por el elemento estructurador, dependiendo de la elección del elemento estructurador la operación morfológica afectará las estructuras de la imagen DICOM, en cuanto a la detección de bordes el método de Prewitt es adecuado cuando lo que se quiere detectar implica cambios bruscos en la imagen como lo fue en este caso con regiones muy ruidosas. La pseudocoloración es una herramienta que está creciendo gradualmente gracias a su eficacia en el sector salud, esto conlleva a estar en constante investigación y mejoramiento de las técnicas existentes.

AGRADECIMIENTOS

Agradecemos al médico radiólogo especializado Álvaro Morales S. por su apoyo en la realización del proyecto, al Instituto Politécnico Nacional, a la Beca de Estimulo Institucional de Formación de Investigadores (BEIFI) del IPN y a la COFAA por el apoyo otorgado.

REFERENCIAS

- [1] Gamal Hamdan Suleiman M.D., "Severe Cranioencephalic Trauma 1st Part" MEDICRIT Revista de Medicina Interna y Medicina Crítica, pp.132, Venezuela, 2005
- [2] Geoff Dougherty, Digital Image Processing for Medical Applications, Cambridge University Press, New York 2009, pages 309-335.
- [3] J. W. Bai, P. A. Li and K. H. Wang, "Automatic whole heart segmentation based on watershed and active contour model in CT images," 2016 5th International Conference on Computer Science and Network Technology (ICCSNT), Changchun, China, 2016, pp. 741-744.

- [4] B. Chanda and D.D. Majumder, *Digital Image Processing and Analysis*, New Delhi: PH India, p54, 2002.
- [5] T. M. Lehmann, A. Kaser, and R. Repges, “A simple parametric equation for pseudocoloring grey scale images keeping their original brightness progression,” *Image Vis. Comput.*, vol. 15, no. 3, pp. 251–257, 1997.
- [6] Dai Jianbiao and Zhou Shaoxiang, “Computer-aided pseudocolor coding of gray images: complementary color-coding technique,” *Proc. SPIE. Electronic Imaging and Multimedia Systems*. vol. 2889, pp. 181–191, September 1996.
- [7] Li Jinxiu , Li Junli and Wei Ping, “Pseudocolor coding of medical images based on gradient” *ICBBE Bioinformatics and Biomedical Engineering* , pp. 932-935, 6-8 July 2007.
- [8] Canny, J., *A Computational Approach To Edge Detection*, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
- [9] *Digital Image Processing*”, Rafael C. Gonzalez, Richard E. Woods. Prentice Hall, second edition., pp.134.

Cálculo Inicial de Profundidad para un Sistema de Visión Monocular

Saúl Martínez Díaz

División de Estudios de Posgrado
TNM Instituto Tecnológico de La Paz
La Paz BCS, México
saulmd@itlp.edu.mx

Maarouf Saad

Département de Génie Électrique
École de Technologie Supérieure
Montréal, Canada
Maarouf.Saad@etsmtl.ca

Jawhar Ghommam

Department of Electrical and Computer Engineering
College of Engineering, Sultan Qaboos University
Al Khoudh, Omán
jawhar.ghommam@gmail.com

Abstract— En años recientes, debido a la gran cantidad de posibles aplicaciones, ha crecido el interés de la comunidad científica por la odometría visual. En muchos casos es conveniente utilizar una sola cámara para este propósito. Desafortunadamente, un sistema monocular permite estimar la ubicación de la cámara con respecto a algún objeto del mundo real únicamente hasta un factor de escala. Sin embargo, si se provee al sistema con la ubicación inicial, es posible estimar ubicaciones sucesivas al mover la cámara. En este artículo se presenta un método para calcular la profundidad inicial de puntos en el espacio tridimensional con respecto a la cámara. El método utiliza tres puntos de referencia, de los cuales únicamente se requiere conocer la distancia real entre cada par de puntos, sin importar la posición en la que se encuentren.

Keywords—visión monocular; odometría visual; estimación de pose; SLAM; estructura desde el movimiento

I. INTRODUCCIÓN

En los últimos años la capacidad de procesamiento de las computadoras se ha incrementado considerablemente. Esto ha hecho posible procesar imágenes digitales en tiempo real. Con ello ha crecido también el interés por el desarrollo de sistemas de medición basados en visión artificial (odometría visual). Esto se debe a que existen muchas aplicaciones en las que se puede utilizar esta tecnología. Entre ellas se encuentran la navegación de robots autónomos [1,2], la localización y mapeo simultáneo (SLAM por sus siglas en inglés) de lugares desconocidos [3,4] y la inspección en sistemas industriales

[5,6]. Muchas de estas aplicaciones requieren información de la localización de los objetos en coordenadas tridimensionales (3D) del mundo real. Los sistemas de GPS utilizados normalmente para este propósito son susceptibles a interferencias, sobre todo en lugares rodeados de edificios, bajo puentes o en interiores. También, presentan márgenes de error grandes, de hasta varios decímetros. Por otro lado, las cámaras RGBD, además de su alto costo, utilizan sensores infrarrojos para determinar la distancia a la que se encuentran los objetos (profundidad). Esto dificulta e incluso impide su aplicación en algunos lugares iluminados con luz natural.

Los sistemas de visión son de bajo costo y, en algunos casos, presentan ventajas con respecto a las tecnologías dominantes que se utilizan en la actualidad. Un sistema de visión puede utilizar una sola cámara (monocular), un par de cámaras (binocular) o más de 2 cámaras. En los sistemas binoculares, por lo general, es necesario realizar una calibración estereoscópica para conocer la rotación y traslación de una cámara con respecto a la otra, la cual sirve como referencia, y mantener fijos esos parámetros. Para calcular la posición de un punto en el espacio tridimensional, cada cámara debe capturar una imagen que contenga dicho punto, después debe identificar las coordenadas del punto dentro de las dos imágenes y triangular para obtener sus coordenadas 3D. Sin embargo, existen algunos inconvenientes prácticos en este tipo de sistemas basados en dos (o más) cámaras como son la diferencia en la respuesta de cada cámara al color y luminancia de la señal de entrada (lo cual dificulta el proceso de identificación), el uso de mayor espacio físico, mayor consumo

de energía, un mayor costo computacional al procesar dos imágenes en cada ocasión y la posibilidad de que las cámaras pierdan la calibración debido a movimientos o vibraciones. Además, cuando se observan puntos lejanos con dos cámaras, el sistema se degenera y tiende a comportarse como monocular. Todo esto ha puesto a los sistemas monoculares como una buena alternativa.

Uno de los principales problemas en un sistema monocular es que la ubicación de algún punto en el espacio 3D se puede conocer únicamente hasta cierto factor de escala, el cual puede obtenerse a partir de la profundidad de dicho punto. En la literatura se han propuesto varias aproximaciones para estimar este factor de escala: En [7] utilizan una cámara como sensor principal y una unidad de medición inercial (IMU por sus siglas en inglés) para determinar la escala. En [8] estiman la profundidad utilizando una red neuronal convolucional, esta estimación se refina entrenando la red con las imágenes consecutivas. En [9] utilizan un patrón de tres círculos concéntricos de diámetro conocido en un plano con la cámara perpendicular a dicho plano para calcular la profundidad inicial. En [10] suponen que el campo de visión de la cámara montada en un dirigible es siempre perpendicular a la tierra, se coloca la cámara a una distancia conocida, se detecta el rostro de una persona centrado en la imagen, se cuentan los píxeles y con estos datos se establece una relación para calcular la profundidad detectando el mismo rostro en futuras imágenes.

En este trabajo proponemos un método para calcular la profundidad de un grupo de tres puntos en el espacio 3D utilizando una cámara calibrada. Para ello se debe conocer únicamente la distancia entre cada par de puntos. Los puntos pueden estar en cualquier posición y no se requiere que la cámara esté perpendicular al plano formado por los tres puntos. Utilizando esta técnica en al menos dos imágenes, tomadas con la misma cámara en diferente pose y las cuales contengan los mismos tres puntos, se puede calcular el cambio de pose (rotación y traslación) de la cámara. Con esta información inicial y el uso de algún detector de puntos de interés adecuado es posible estimar los cambios de pose en futuras imágenes. El resto del artículo se organiza de la siguiente manera: la sección II presenta los conceptos básicos que soportan el trabajo, en la sección III se presenta el método propuesto, en la sección IV se presentan algunos resultados experimentales y en la sección V se resumen nuestras conclusiones.

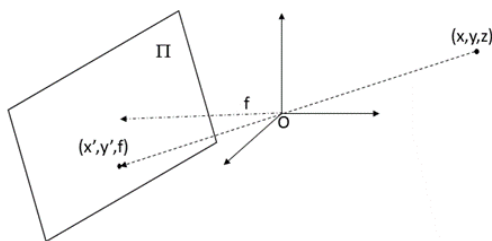


Fig. 1. Modelo ojo de aguja

II. CONCEPTOS BÁSICOS

A. Modelo Ojo de Aguja

Debido a su simpleza, el modelo Ojo de Aguja es el más utilizado para representar la formación de imágenes en una cámara. En la figura 1 se muestra el principio en el que se basa este modelo. Como se puede apreciar, cada punto P con coordenadas (x, y, z) en el espacio 3D es proyectado a través del ojo de aguja (el cual se toma como origen del sistema de coordenadas) al punto P' con coordenadas (x', y', f) del plano Π de la cámara, donde f es la distancia focal de la cámara. De esta figura, por semejanza de triángulos, se puede establecer que [11]:

$$\frac{x'}{x} = \frac{y'}{y} = \frac{f}{z} = \lambda \quad (1)$$

Donde λ es un factor de escala el cual, al ser conocido, permite calcular las coordenadas 3D a partir de las coordenadas del píxel que representa al punto deseado dentro de la imagen y la distancia focal, la cual se obtiene en la calibración.

B. Calibración de Cámaras

Para la configuración del sistema estereoscópico de cámaras se utiliza el modelo de Ojo de Aguja. La calibración de las cámaras se realizó con la técnica propuesta por Zhang [12]. Con esta técnica de calibración solamente se requiere que la cámara observe un patrón plano (ver figura 2) tomado desde diferentes orientaciones. El patrón o la cámara pueden ser movidos libremente y no es necesario conocer el movimiento realizado. Esta calibración permite obtener los parámetros intrínsecos y extrínsecos de la cámara, los cuales se relacionan de la siguiente forma [13]:

$$\lambda \mathbf{P}' = \mathbf{A}[\mathbf{R} \ \mathbf{t}]\mathbf{P} \quad (2)$$

Donde \mathbf{P} es un punto del objeto en 3D, en coordenadas homogéneas; \mathbf{P}' es el mismo punto del objeto en coordenadas 2D homogéneas; $[\mathbf{R} \ \mathbf{t}]$ es una matriz de parámetros extrínsecos (rotación y traslación); λ es un factor de escala arbitrario y \mathbf{A} es una matriz de parámetros intrínsecos. En particular, en este trabajo se utiliza la información de la matriz de parámetros intrínsecos, definida por:

$$\mathbf{A} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

Donde f_x, f_y proporcionan información (dependiendo del tamaño del píxel) de la distancia focal en la dirección de x e y , respectivamente; c_x, c_y son las coordenadas del punto principal de la imagen.

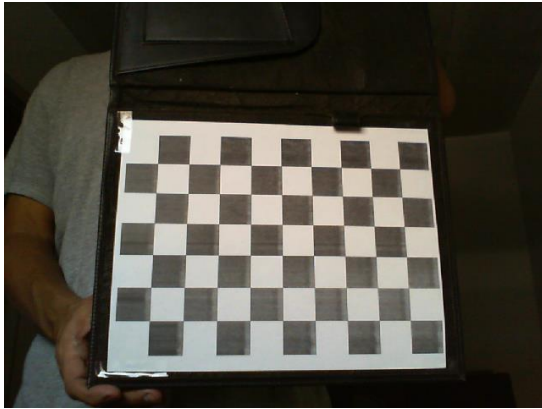


Fig. 2. Patrón de calibración

III. MÉTODO PROPUESTO

De la ecuación (1) se pueden obtener las siguientes relaciones:

$$\begin{aligned} x &= \frac{x'}{f} z \\ y &= \frac{y'}{f} z \end{aligned} \quad (4)$$

Supongamos que se tienen al menos tres puntos P1, P2, P3 en el espacio 3D con coordenadas (x_1, y_1, z_1) , (x_2, y_2, z_2) , y (x_3, y_3, z_3) , respectivamente. Supongamos también que las distancias d_1 , d_2 y d_3 entre cada par de puntos son conocidas. Utilizando las relaciones en (4), la distancia euclidiana cuadrática entre cada par de puntos se puede calcular por:

$$\begin{aligned} d_1 &= \left(\frac{x'_1}{f} z_1 - \frac{x'_2}{f} z_2 \right)^2 + \left(\frac{y'_1}{f} z_1 - \frac{y'_2}{f} z_2 \right)^2 + (z_1 - z_2)^2 \\ d_2 &= \left(\frac{x'_1}{f} z_1 - \frac{x'_3}{f} z_3 \right)^2 + \left(\frac{y'_1}{f} z_1 - \frac{y'_3}{f} z_3 \right)^2 + (z_1 - z_3)^2 \\ d_3 &= \left(\frac{x'_3}{f} z_3 - \frac{x'_2}{f} z_2 \right)^2 + \left(\frac{y'_3}{f} z_3 - \frac{y'_2}{f} z_2 \right)^2 + (z_3 - z_2)^2 \end{aligned} \quad (5)$$

Este sistema de ecuaciones no lineales puede resolverse numéricamente para z_i , las cuales representan la profundidad a la que se encuentra cada punto con respecto a la cámara. Sustituyendo estos resultados en (3) podemos obtener también las coordenadas (x_i, y_i) de los tres puntos. Si se toman dos imágenes con la misma cámara en diferente pose y se aplica el método propuesto es posible obtener la posición relativa entre ambas tomas. Con esta información inicial es posible determinar la pose en tomas sucesivas. Una manera de hacer esto último es utilizar algún detector de puntos adecuado para

obtener otros puntos de interés en las dos imágenes. Con esta información se pueden buscar algunos de esos puntos de interés en una tercera imagen y triangular para obtener el desplazamiento relativo con el cual se puede calcular la nueva pose relativa. Este proceso puede repetirse para cada nueva imagen, utilizando las dos imágenes previas para triangular.

IV. RESULTADOS EXPERIMENTALES

Para probar el método propuesto se utilizó el patrón de tablero de ajedrez mostrado en la figura 2. Cada cuadro es de 27x27 milímetros. Se utilizó una cámara USB Microsoft configurada para entregar imágenes en RGB con una resolución de 640x480x3 pixeles. La cámara se calibró utilizando el método de Zhang. Después se aplicó el siguiente algoritmo:

- Capturar una imagen del patrón
- Convertir la imagen a escala de grises
- Corregir la distorsión radial [14]
- Detectar esquinas del patrón utilizando el algoritmo de Harris [15]
- Seleccionar tres puntos del patrón
- Resolver el conjunto de ecuaciones (5) con el algoritmo de descenso de gradiente propuesto en [16]

La cámara fue colocada a diferentes distancias del patrón y se aplicó el algoritmo. En la práctica, aun cuando se tomen dos imágenes desde la misma distancia, el resultado no es idéntico debido a factores como ruido aleatorio en los sensores y variaciones en la fuente de iluminación. Por esta razón, para garantizar resultados estadísticamente correctos se efectuaron 30 pruebas en cada posición. Los resultados del promedio m y la desviación estándar σ se muestran en la tabla I, todas las medidas están en milímetros.

Las diferencias en la profundidad entre los tres puntos se deben a que el ángulo de visión de la cámara no es perpendicular al tablero, por lo cual no todos los puntos se encuentran a la misma distancia. Por otro lado, la desviación estándar, la cual se relaciona con la incertidumbre de la medición, es menor a 0.25% del promedio, en todos los casos. Esto indica que el error esperado es bajo, del orden de un par de milímetros.

TABLA I PROFUNDIDAD MEDIDA VS. CALCULADA

| Distancia medida del tablero | P1 calculado | | P2 calculado | | P3 calculado | |
|------------------------------|--------------|----------|--------------|----------|--------------|----------|
| | m | σ | m | σ | m | σ |
| 400 | 416.35 | 0.40 | 403.98 | 0.59 | 391.54 | 0.79 |
| 450 | 449.51 | 0.49 | 438.73 | 0.79 | 427.85 | 1.12 |
| 500 | 502.31 | 0.49 | 491.62 | 0.71 | 480.89 | 0.93 |
| 550 | 542.63 | 0.66 | 530.80 | 0.96 | 518.97 | 1.05 |
| 600 | 600.96 | 0.61 | 589.95 | 0.90 | 578.89 | 1.19 |

V. CONCLUSIONES

En este artículo se presentó un método para calcular la profundidad de puntos en el espacio tridimensional utilizando un sistema de visión monocular. Para el cálculo solamente es necesario conocer la distancia entre cada par de puntos. Esta información puede servir como inicialización para sistemas de control industrial, SLAM o navegación de robots autónomos, que utilicen una sola cámara. Los resultados experimentales muestran un buen desempeño del algoritmo en imágenes tomadas con una cámara de bajo costo en condiciones no controladas.

RECONOCIMIENTOS

Los autores agradecen por el apoyo y las facilidades brindadas para la realización de este trabajo al Consejo Nacional de Ciencia y Tecnología, el Tecnológico Nacional de México - Instituto Tecnológico de La Paz y al École de Technologie Supérieure de Montreal Canada.

REFERENCIAS

- [1] K. Wang, Y. Liu, and L. Li, "Vision-based tracking control of underactuated water surface robots without direct position measurement," *IEEE Transactions on Control Systems Technology*, vol. 23(6), pp. 2391-2399, November 2015.
- [2] M. Knudson and K. Tumer, "Adaptive navigation for autonomous robots," *Robotics and Autonomous Systems*, vol. 59, pp. 410-420, February 2011.
- [3] R. Mur-Artal and J. D. Tardos, "ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras," *IEEE Transactions on Robotics*, vol. 33(5), pp. 1255-1262, June 2017.
- [4] R. Mur-Artal and J. D. Tardos, "Visual-inertial monocular slam with map reuse," *IEEE Robotics and Automation Letters*, vol. 2(2), pp. 796-803, April 2017.
- [5] L. Xu, X. He, X. Li and M. Pan, "A machine-vision inspection system for conveying attitudes of columnar objects in packing processes," *Measurement*, vol. 87, pp. 255-273, March 2016.
- [6] R. Shanmugamani, M. Sadique and B. Ramamoorthy, "Detection and classification of surface defects of gun barrels using computer vision and machine learning," *Measurement*, vol. 60, pp. 222-230, October 2014.
- [7] J. J. Tarrio, and S. Pedre, "Realtime edge based visual inertial odometry for MAV teleoperation in indoor environments," *J. Intell. Robot Syst.*, pp. 1-18, October 2017.
- [8] X. Yin, X. Wang, X. Du and Q. Chen, "Scale recovery for monocular visual odometry using depth estimated with deep convolutional neural fields," in *2017 IEEE International Conference on Computer Vision*, pp. 5871-5879, 2017.
- [9] Z. Said, K. Sundaraj and M. N. A. Wahab, "Depth estimation for a mobile platform using monocular vision," in *International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012)*, vol. 41, pp. 945-950, 2012.
- [10] N. Yao, E. Anaya, Q. Tao, S. Cho, H. Zheng and F. Zhang, "Monocular vision-based human following on miniature robotic blimp," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3244-3249, 2017.
- [11] A. I. Barranco-Gutiérrez., S. Martínez-Díaz and J. L. Gómez-Torres, *Visión Estereoscópica por Computadora con MATLAB y OPENCV*, 1st ed., Lulu, 2014, pp. 17-19.
- [12] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22(11), pp. 1330-1334, 2000.
- [13] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge University Press, 2004, pp. 154-156.
- [14] G. Bradski and A. Kaehler, *Learning OpenCV*, 1st ed., O'Reilly, 2008, pp. 375-377.
- [15] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Conference* pp. 147-151, 1988.
- [16] J.J. Moré, and D.C. Sorensen, "Computing a trust region step," *SIAM Journal on Scientific and Statistical Computing*, vol. 3, pp 553-572, 1983.

Algoritmo genético para la generación de trayectorias de un robot industrial con evasión de colisiones para una interacción segura humano-robot.

César Eduardo Cea Montufar,
Bárbara Mayela Gutiérrez Mejía.

Departamento de Formación Básica Disciplinaria,
Unidad Profesional Interdisciplinaria de Ingeniería
Campus Hidalgo del Instituto Politécnico Nacional
San Agustín Tlaxiaca, Hidalgo, México.
ccea@ipn.mx, bargutm@gmail.com

Emmanuel Alejandro Merchán Cruz,
Javier Ramírez Gordillo,
Miguel Ángel Cerro Ramirez,

Sección de Estudios de Posgrado e Investigación.
Escuela Superior de Ingeniería Mecánica y Eléctrica
Unidad Azcapotzalco del Instituto Politécnico Nacional.
Ciudad de México, México.
eamerchan@ipn.mx, ramirezgordillo@gmail.com,
migue.a.cerro86@gmail.com

Resumen—Se presenta el desarrollo de un método evolutivo que combina el problema de planeación de trayectorias, incorporación de obstáculos y campos de potencial artificial para posicionar y orientar la herramienta de un robot industrial ABB® IRB1600 en función de la generación de trayectorias libres de colisión con obstáculos presentes en su área de trabajo, garantizando la integridad del robot, estaciones y trabajadores.

Palabras Clave—Algoritmo genético, función multiobjetivo, planeación de trayectorias, robot industrial, campo de potencial artificial, cuaternión dual.

I. INTRODUCCIÓN

Garantizar la seguridad de los trabajadores así como la integridad de la maquinaria y equipo es una de las principales consideraciones en la industria moderna y dentro del campo de la manufactura el uso de robots industriales para realizar distintas tareas es cada vez mayor dado que proporciona una producción más alta, eficiente y de mejor calidad, esta condición genera mayores situaciones en las que humanos y robots operan en el mismo espacio de trabajo por lo que es necesario prevenir y evitar cualquier tipo de colisión considerando todas las formas posibles en que un daño puede afectar a una persona, desde el contacto físico hasta los efectos psicológicos adversos resultantes de una interacción desagradable o peligrosa, este trabajo busca desarrollar un algoritmo capaz de generar de manera automática la trayectoria de un robot industrial de la marca ABB modelo IRB1600 utilizando un algoritmo genético para que el robot pueda realizar diversas tareas evitando colisionar con cualquier obstáculo o persona. [1]

El planteamiento del problema consiste en que el robot manipulador pueda desplazarse desde una configuración inicial hasta un punto final sin colisionar, el problema contempla la existencia de mínimos locales entre ciertas configuraciones de los obstáculos y la meta, por lo que la búsqueda de trayectorias libres de colisiones implica alcanzar

el mínimo global evadiendo los obstáculos en una serie de configuraciones coordinadas mediante un algoritmo planeador capaz de escapar de zonas donde se localicen mínimos locales o puntos singulares y para lograrlo un enfoque de optimización facilita el problema y la búsqueda de soluciones factibles. [2]

Esta estrategia considera a la meta como un dipolo de atracción y a los obstáculos como superficies con una carga de repulsión respecto a la punta del manipulador que es atraída por la meta la cual es un mínimo global de la función de potencial y cerca de los obstáculos la función de potencial crece indefinidamente haciendo que el manipulador se aleje de ellos, así la idea consiste en buscar el camino más corto entre el manipulador y la meta entre configuraciones libres de colisión a través del gradiente descendiente como una función de potencial caracterizado en los obstáculos por un campo de repulsión.

II. MODELO CINEMÁTICO DEL ROBOT INDUSTRIAL DE 6 GDL

El manipulador robótico IRB1600 de ABB® mostrado en la figura 1 es un brazo robótico de uso industrial empleado por empresas del sector industrial especializadas en la robótica y automatización, en diversas aplicaciones como pintura, soldadura, ensamble, manejo y manipulación de materiales entre otras es esta misma versatilidad del robot la que lleva a las empresas a la búsqueda de métodos que faciliten la programación del robot y aseguren una interacción segura con su entorno. [3]



Fig. 1. Manipulador IRB1600.

El robot manipulador cuenta con 6 grados de libertad los cuales representan el movimiento de cada uno de sus eslabones, representados en la figura 2, obtener el modelo cinemático del manipulador permite establecer una relación entre las configuraciones de cada una de sus articulaciones con la posición y orientación de la herramienta en el espacio de trabajo.

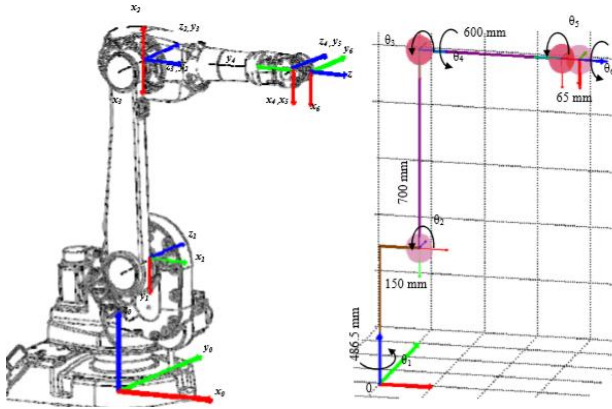


Fig. 2. Movimiento y sistemas coordenados por articulación.

La función objetivo a solucionar depende de ecuaciones de enlace cinemático resuelto con cuaterniones duales para facilitar el algoritmo, puesto que el álgebra de cuaternión es simple y su procedimiento representa menor cantidad de operaciones en su cálculo [4], además de poder describir las rotaciones y traslaciones sobre un cuerpo rígido con base en los sistemas de coordenadas relacionados en la representación paramétrica de Denavit y Hartenberg manteniendo compatibilidad con los métodos convencionales. [5], la tabla I muestra los parámetros D-H del manipulador IRB1600 en su configuración inicial.

TABLA I. PARÁMETROS DENAVIT Y HARTENBERG DEL ROBOT

| i | θ | α | a | d |
|-----|-------------------|----------|-------|--------|
| 1 | $\theta_1=0$ | $-\pi/2$ | 0.150 | 0.4865 |
| 2 | $\theta_2=-\pi/2$ | 0 | 0.700 | 0 |
| 3 | $\theta_3=\pi$ | $\pi/2$ | 0 | 0 |
| 4 | $\theta_4=0$ | $-\pi/2$ | 0 | 0.600 |
| 5 | $\theta_5=0$ | $\pi/2$ | 0 | 0 |
| 6 | $\theta_6=0$ | 0 | 0 | 0.065 |

Las ecuaciones de enlace cinemático basadas en cuaterniones duales y parámetros Denavit-Hartenberg se determinan por las ecuaciones cinemáticas desarrolladas por Ramírez-Gordillo en 2011 [6]. Los parámetros de la tabla I evalúan las ecuaciones que se muestran de (1) a (6) para obtener posición y orientación en cuaterniones duales de cada articulación mostrados en la tabla II.

$${}^0DQ_1^{DH} = \begin{bmatrix} \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_1}{2}\right), -\frac{\sqrt{2}}{2} \cos\left(\frac{\theta_1}{2}\right), i, -\frac{\sqrt{2}}{2} \sin\left(\frac{\theta_1}{2}\right), j, \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_1}{2}\right), k \\ 0.0375 \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_1}{2}\right) - 0.1216 \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_1}{2}\right), \epsilon, \left(0.0375 \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_1}{2}\right) + 0.1216 \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_1}{2}\right)\right) i\epsilon \\ \left(0.0375 \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_1}{2}\right) - 0.1216 \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_1}{2}\right)\right) j\epsilon, \left(0.0375 \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_1}{2}\right) + 0.1216 \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_1}{2}\right)\right) k\epsilon \end{bmatrix} \quad (1)$$

$${}^1DQ_2^{DH} = \begin{bmatrix} \cos\left(\frac{\theta_2}{2} - \frac{\pi}{4}\right), 0i, 0j, \sin\left(\frac{\theta_2}{2} - \frac{\pi}{4}\right)k \\ 0\epsilon, 0.350 \sin\left(\frac{\theta_2}{2} - \frac{\pi}{4}\right) i\epsilon, 0.350 \cos\left(\frac{\theta_2}{2} - \frac{\pi}{4}\right) j\epsilon, 0k\epsilon \end{bmatrix} \quad (2)$$

$${}^2DQ_3^{DH} = \begin{bmatrix} \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_3}{2} + \frac{\pi}{2}\right), \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_3}{2} + \frac{\pi}{2}\right), i, \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_3}{2} + \frac{\pi}{2}\right), j, \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_3}{2} + \frac{\pi}{2}\right), k, 0\epsilon, 0i\epsilon, 0j\epsilon, 0k\epsilon \end{bmatrix} \quad (3)$$

$${}^3DQ_4^{DH} = \begin{bmatrix} \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_4}{2}\right), -\frac{\sqrt{2}}{2} \cos\left(\frac{\theta_4}{2}\right), i, -\frac{\sqrt{2}}{2} \sin\left(\frac{\theta_4}{2}\right), j, \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_4}{2}\right), k \left(0.150 \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_4}{2}\right)\right) \epsilon \\ \left(0.150 \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_4}{2}\right)\right) i\epsilon, \left(0.150 \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_4}{2}\right)\right) j\epsilon, \left(0.150 \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_4}{2}\right)\right) k\epsilon \end{bmatrix} \quad (4)$$

$${}^4DQ_5^{DH} = \begin{bmatrix} \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_5}{2}\right), \frac{\sqrt{2}}{2} \cos\left(\frac{\theta_5}{2}\right), i, \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_5}{2}\right), j, \frac{\sqrt{2}}{2} \sin\left(\frac{\theta_5}{2}\right), k, 0\epsilon, 0i\epsilon, 0j\epsilon, 0k\epsilon \end{bmatrix} \quad (5)$$

$${}^5DQ_6^{DH} = \begin{bmatrix} \cos\left(\frac{\theta_6}{2}\right), 0i, 0j, \sin\left(\frac{\theta_6}{2}\right)k, 0.0325 \sin\left(\frac{\theta_6}{2}\right)\epsilon, 0i\epsilon, 0j\epsilon, 0.0325 \cos\left(\frac{\theta_6}{2}\right)k\epsilon \end{bmatrix} \quad (6)$$

TABLA II. POSICIÓN Y ORIENTACIÓN POR CADA ARTICULACIÓN

| ${}^{i-1}DQ_i$ | w | i | j | k | ϵ | $i\epsilon$ | $j\epsilon$ | $k\epsilon$ |
|----------------|------|-------|------|-------|------------|-------------|-------------|-------------|
| 0DQ_1 | .707 | -.707 | 0 | 0 | .053 | .053 | -.172 | .172 |
| 1DQ_2 | .707 | 0 | 0 | -.707 | 0 | .247 | -.247 | 0 |
| 2DQ_3 | 0 | 0 | .707 | -.707 | 0 | 0 | 0 | 0 |
| 3DQ_4 | .707 | -.707 | 0 | 0 | 0 | 0 | 0 | .212 |
| 4DQ_5 | .707 | .707 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5DQ_6 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | .032 |

Mediante (7) se puede conocer la posición y orientación del efector final respecto al sistema de coordenadas de la base del robot considerando las configuraciones angulares de todas sus articulaciones, en (8) se puede observar esta relación considerando las condiciones iniciales establecidas en la asignación de los parámetros D-H.

$${}^0DQ_6^{DH} = \prod_{i=1}^n {}^{i-1}DQ_i^{DH} = {}^0DQ_1^{DH} {}^1DQ_2^{DH} {}^2DQ_3^{DH} {}^3DQ_4^{DH} {}^4DQ_5^{DH} {}^5DQ_6^{DH} \quad (7)$$

$$T_f = \begin{bmatrix} 0 & 0 & 1 & 0.815 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1.186 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

III. MODELO DINÁMICO DEL ROBOT INDUSTRIAL DE 6 GDL

El análisis de movimiento de los cuerpos bajo la acción de las fuerzas que lo producen, se describe en términos de ecuaciones que relacionan el tiempo con el cambio de configuración del robot, obedeciendo a la segunda ley de Newton, donde la razón de cambio en el momento de una partícula es igual a la fuerza aplicada a esta.

En los sistemas dinámicos, los manipuladores robóticos de n-grados de libertad son considerados como sistemas no lineales multivariable, asociados a las posiciones, velocidades, aceleraciones y torques. La formulación de Lagrange (9) considera la energía total de un sistema de n-grados de libertad como la diferencia entre energía cinética (K) y la energía potencial (P).

$$L(q, \dot{q}) = K(q, \dot{q}) - P(q) \quad (9)$$

El Lagrangiano (L) determina la energía contenida en el sistema expresado en coordenadas generalizadas del brazo (q_i), considerando perturbaciones externas, pares de torsión que se aplican al sistema en cada articulación τ_i y disipación de energía R_{Di} mostrado en (10) y su representación matricial expresada en (11) por la matriz simétrica de inercia M , matriz de fuerza centrífuga y de Coriolis C , matriz de carga gravitatoria G y matriz de rozamiento viscoso F_D . [7]

$$\frac{d}{dt} \left(\frac{\partial L(q, \dot{q})}{\partial \dot{q}_i} \right) - \frac{\partial L(q, \dot{q})}{\partial q_i} + \frac{\partial R_{Di}}{\partial \dot{q}_i} = \tau_i \quad (10)$$

$$\tau(q) = M(q) \ddot{q} + C(q, \dot{q}) + G(q) + F_D(\dot{q}) \quad (11)$$

Algunos parámetros del robot se muestran en la tabla III, estos aspectos determinan la complejidad del problema basado en la simplicidad del algoritmo genético, así como el diseño de la función aptitud enfocado en una función de optimización en búsqueda de trayectorias para satisfacer condiciones de posición y orientación.

TABLA III. PARÁMETROS DINÁMICOS DEL ROBOT DE 6 GDL

| i | Masa [Kg] | Tensor de inercia [Kgm ²] | | | | | | Centro de masa [m] | | | Fricción Coulomb y Viscosa | |
|---|-----------|---------------------------------------|----------|----------|----------|----------|----------|--------------------|-------|-------|----------------------------|-------|
| | | I_{xx} | I_{yy} | I_{zz} | I_{xy} | I_{yz} | I_{xz} | r_x | r_y | r_z | f_c | f_v |
| 1 | 13 | 1.1 | 1.1 | .12 | 0 | 0 | 0 | 0 | 0 | .3 | 29 | 6.2 |
| 2 | 22 | .4 | .96 | .96 | 0 | 0 | 0 | .1 | 0 | .1 | 14 | 8.8 |
| 3 | 5 | .07 | .01 | .07 | 0 | 0 | 0 | .02 | 0 | 0 | 7.3 | 5.3 |
| 4 | 1.2 | .01 | .01 | .01 | 0 | 0 | 0 | 0 | 0 | .1 | 4.7 | 2.5 |
| 5 | .62 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1.7 | 1.2 |
| 6 | .2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | .06 | .6 |

IV. IMPLEMENTACIÓN DEL ALGORITMO GENÉTICO

El posicionar la herramienta del robot industrial en un punto específico del espacio cartesiano sin recurrir a la solución de su cinemática inversa requiere del diseño adecuado de la función multiobjetivo del algoritmo genético (12) a (18) la cual depende de las ecuaciones de enlace cinemático resueltas con cuaterniones duales, la función de potencial de atracción, el potencial de repulsión y la función de mínimo desplazamiento de los eslabones para generar trayectorias con distancias seguras entre y el robot y los obstáculos presentes en su espacio de trabajo.

$$f_i = C_1 e^{-(w_1 Dq_i + w_2 P a_i + w_3 P r_i + w_4 O_i)} \quad (12)$$

$$Dq_i = \sqrt{\sum_{j=1}^{j=8} (e_i^{Og} - e_i^{Qa})^2} \quad (13)$$

$$P a_i = \beta \sqrt{(x_g - (\Delta x + x)_i)^2 + (y_g - (\Delta y + y)_i)^2 + (z_g - (\Delta z + z)_i)^2} \quad (14)$$

$$P r_i = \rho \sum_{k=1}^l \sqrt{(x_{obs} - (\Delta x + x)_i)^2 + (y_{obs} - (\Delta y + y)_i)^2 + (z_{obs} - (\Delta z + z)_i)^2} \quad (15)$$

$$O_i = \sum_{j=1, k=1}^{j=m, k=n} (\theta_k^{j+1} - \theta_k^j)^2 \quad (16)$$

$$P r < (s + r) \{ P r \quad (17)$$

$$r \leq P r \leq (s + r) \{ s + r - P r \quad (18)$$

f_i = Función aptitud.

C_i = Constante de penalización [0, 1].

Dq = Distancia entre los elementos de dos cuaterniones duales

Pa = Distancia entre el punto deseado y el inicial.

Pr = Distancia entre el punto deseado y el de los obstáculos.

i, j = ésimas iteración.

k = Número de soluciones.

n = Tamaño de las soluciones.

β = Factor de escala positivo para el potencial de atracción.

ρ = Factor de escala positivo para el potencial de repulsión.

r = Distancia de contacto inminente

s = Distancia de zona de influencia del obstáculo.

O = Mínimo movimiento entre las articulaciones.

θ = Desplazamiento angular de los eslabones.

Qg = Cuaternión dual deseado del manipulador.

Qa = Cuaternión dual actual del manipulador. n^o

l = Número total de obstáculos.

x, y = Coordenadas del efector final.

$\Delta x, \Delta y$ = Conjunto de soluciones encontradas.

A. Caso 1

En este caso se propone que el robot obtenga la trayectoria que satisfaga condiciones de posición y orientación sin colisionar con un obstáculo en su espacio de trabajo, la tabla IV proporciona los parámetros del algoritmo genético.

TABLA IV. PARÁMETROS DEL ALGORITMO GENÉTICO.

| | |
|--------------------------------------|-------------------------------|
| $f(\Delta \theta^s)$ | $\in [-0.5^\circ, 0.5^\circ]$ |
| Número de individuos en la población | 110 |
| Precisión inicial | 5 |
| Probabilidad de cruce | 0.85 |
| Probabilidad de mutación | 0.11 |
| Regeneración | 0.13 |
| Factor de perturbación | 0.2 |
| Máximo de generaciones | 1000 |

El robot manipulador tiene una configuración inicial de $(0^\circ, 0^\circ, 0^\circ, 0^\circ, 0^\circ, 180^\circ)$ y se encuentra desplazado en $(0.2m, 0.2m, -0.2m)$ del sistema de coordenadas mundo; mientras que la orientación y posición es de $(0^\circ, 90^\circ, 0^\circ, 1.1m, 0m, -0.4m)$ además de añadirse un conjunto de obstáculos cuya configuración forma el cuerpo que tiene que evadir el robot hasta llegar a su meta.

Por lo tanto, la orientación y posición de la punta del manipulador se encuentra representada por (19) respecto al sistema de coordenadas global wD_6 , la orientación y posición deseada está determinada por (20).

$${}^wDQ_6 = 0.5 + 0.5i + 0.5j + 0.5k - 0.55\varepsilon + 0.057i\varepsilon + 0.042j\varepsilon + 0.45k\varepsilon \quad (19)$$

$$DQ = 0.707 + 0.707j - 0.0707\varepsilon + 0.6717i\varepsilon + 0.0707j\varepsilon + 0.2474k\varepsilon \quad (20)$$

Se contemplan restricciones en el movimiento de los eslabones, estos tienen un límite de movimiento debido a la configuración física del manipulador robótico, donde los eslabones pueden colisionar con ellos mismos, así los límites de cada articulación son descritos en la tabla V, es importante señalar que cada vez que se violan estos límites de movimiento angular el algoritmo descarta esa configuración penalizando la función aptitud por medio de la constante de penalización (12) anulando su valor alcanzado.

TABLA V. LÍMITES DE MOVIMIENTO PARA CADA ARTICULACIÓN.

| $\theta's$ | $+\theta's$ | $-\theta's$ |
|------------|-------------|-------------|
| θ_1 | 180° | -180° |
| θ_2 | 150° | -90° |
| θ_3 | 65° | -245° |
| θ_4 | 200° | -200° |
| θ_5 | 115° | -115° |
| θ_6 | 400° | -400° |

Los obstáculos representan dos superficies con los vértices en las coordenadas de inicio para $Ob_1 (0.75, -0.2, 0.55)$ y $Ob_2 (1.3, -0.2, 0.55)$, los cuales terminan en $Ob_1 (1.3, 0.2, 0.55)$ y $Ob_2 (1.3, 0.2, 0.95)$ respectivamente. La trayectoria generada por el algoritmo genético se muestra en la figura 3, donde cada punto que forma la superficie tiene un radio $r = 0.1$ y una zona de influencia alrededor del obstáculo de $rs=0.05$ sin colisionar con los obstáculos que forman la superficie y aunque pasa cerca de la zona de influencia, depende del factor de escala positiva del campo de repulsión para aproximar o alejar los eslabones del manipulador de los obstáculos tanto como el problema lo requiera.

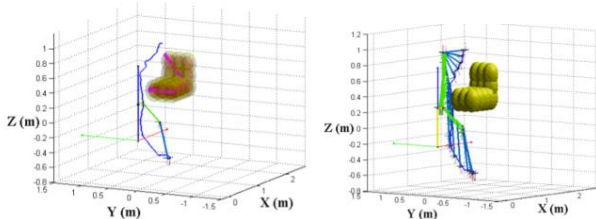


Fig. 3. Trayectoria y configuraciones de las articulaciones.

La variación angular por cada una de las articulaciones del robot se puede observar en la figura 4, con esta variación angular y el tiempo en que se realice la trayectoria se pueden determinar las variaciones de velocidades, aceleraciones y torque por cada articulación, los movimientos de la trayectoria planeada por el algoritmo se muestran en la figura 5.

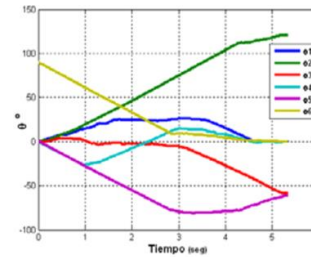


Fig. 4. Posición angular de cada articulación en la trayectoria.

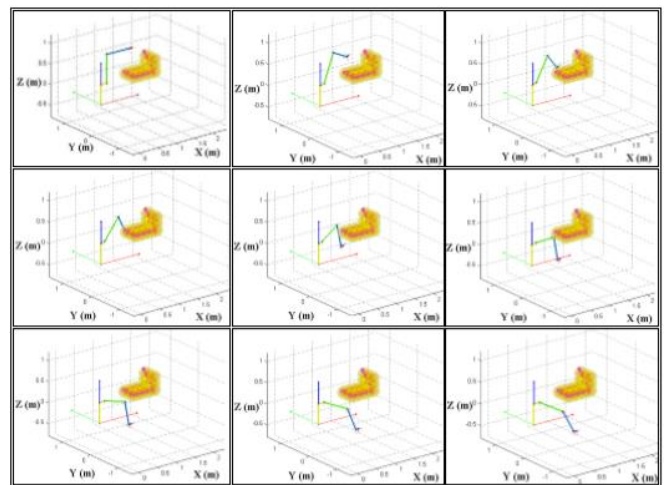


Fig. 5. Perfil de configuraciones del manipulador.

El desempeño del algoritmo y sus resultados se pueden observar en la tabla VI, mientras que en la figura 6 se representa la evolución en la disminución del error por segmentos evaluados así como la aptitud alcanzada por cada segmento.

TABLA VI. RESULTADOS DEL ALGORITMO GENÉTICO.

| gen | segmentos | error | aptitud | aptitud promedio | tiempo |
|------|-----------|-------|---------|------------------|--------|
| 1000 | 385 | .103 | 0.01618 | 0.02443 | 5,338 |

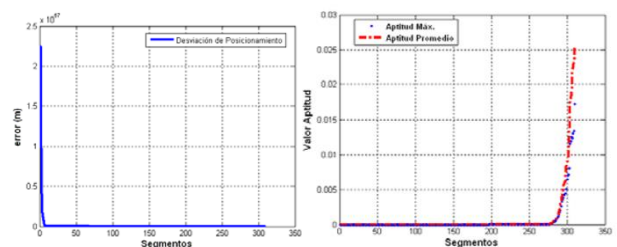


Fig. 6. Evolución de la aptitud y el error por cada segmento.

B. Caso 2

El caso propuesto consiste en que el robot industrial ABB® IRB1600 tiene que satisfacer condiciones de posición y orientación sin colisionar con los obstáculos que forman dos estaciones de trabajo, donde el robot tiene que realizar la tarea sin colisionar con la primera estación, planear a través del algoritmo genético y auxiliarse del campo de potencial artificial para evadir la segunda estación de trabajo a la cual tiene que entrar desplazarse de su posición inicial además el robot se encuentra en una posición conocida como empotramiento en el pórtico (Gantry), esta posición es de gran utilidad a nivel industrial puesto que el robot tiene más volumen libre de trabajo dado que sus eslabones y el empotramiento no interfieren en su trabajo como en los casos donde el robot se encuentra anclado al suelo, la tabla VII indica los parámetros del algoritmo genético.

TABLA VII. PARÁMETROS DEL ALGORITMO GENÉTICO.

| $f(\Delta\theta's)$ | $\in[-0.5^\circ, 0.5^\circ]$ |
|--------------------------------------|------------------------------|
| Número de individuos en la población | 50 |
| Precisión inicial | 4 |
| Probabilidad de cruce | 1 |
| Probabilidad de mutación | 0.37 |
| Regeneración | 0.2 |
| Factor de perturbación | 0.2 |
| Máximo de generaciones | 1000 |

El robot manipulador tiene una configuración inicial de $(0^\circ, 0^\circ, 0^\circ, 0^\circ, 0^\circ, 0^\circ)$ y se encuentra orientado y desplazado en $(180^\circ, 0^\circ, 0^\circ, 0.2m, 0.2m, 2.2m)$ respecto al sistema de coordenadas mundo; mientras que la orientación y posición deseada es de $(-90^\circ, 0^\circ, 90^\circ, 0.2m, 1m, 1m)$ además de añadirse un conjunto de obstáculos cuya configuración forma 2 cuerpos y meta se encuentra dentro de uno de los objetos.

Por lo tanto, la orientación y posición de la punta del manipulador se encuentra representada por (21) respecto al sistema de coordenadas global wD_6 , la orientación y posición deseada está determinada por (22).

$${}^wDQ_6 = -0.707i - 0.707k + 0.707\varepsilon - 0.707i\varepsilon + 0.0005j\varepsilon + 0.0707k\varepsilon \quad (19)$$

$$DQ = 0.5 - 0.5i + 0.5j + 0.5k - 0.45\varepsilon + 0.05i\varepsilon - 0.05j\varepsilon + 0.55k\varepsilon \quad (20)$$

Los obstáculos se encuentran localizados de tal forma que representan dos estaciones en forma de cubos, donde cada punto que forman los cubos tienen un radio $r = 0.15$ y una zona de influencia alrededor del obstáculo de $rs=1$, al igual que en caso 1, los límites de movimiento en cada articulación se encuentran en la tabla V y son considerados para determinar la constante de penalización en la función aptitud (12).

La figura 7 muestra la trayectoria resultante por el algoritmo donde se llega a la meta sin colisionar con las superficies presentes en la trayectoria realizando ambas tareas.

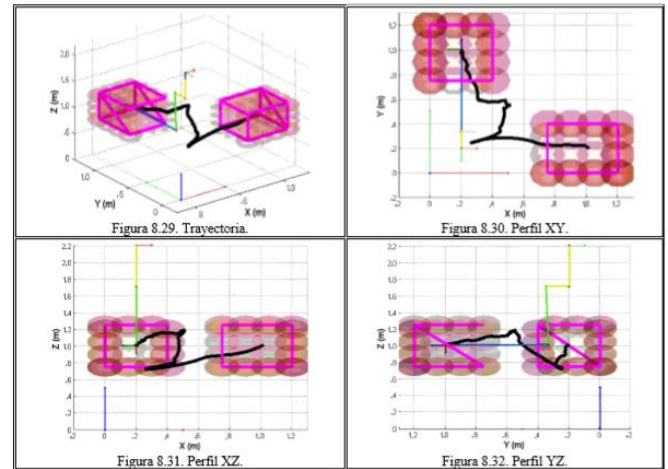


Fig. 7. Trayectoria generada.

La variación angular por cada una de las articulaciones del robot se observa en la figura 8, con esta variación angular y el tiempo en que se realice la trayectoria se pueden determinar las variaciones de velocidades, aceleraciones y torque por cada articulación, los movimientos de la trayectoria planeada por el algoritmo se observan en la figura 9.

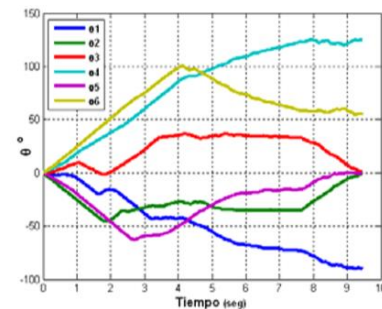


Fig. 8. Posición angular de cada articulación en la trayectoria.

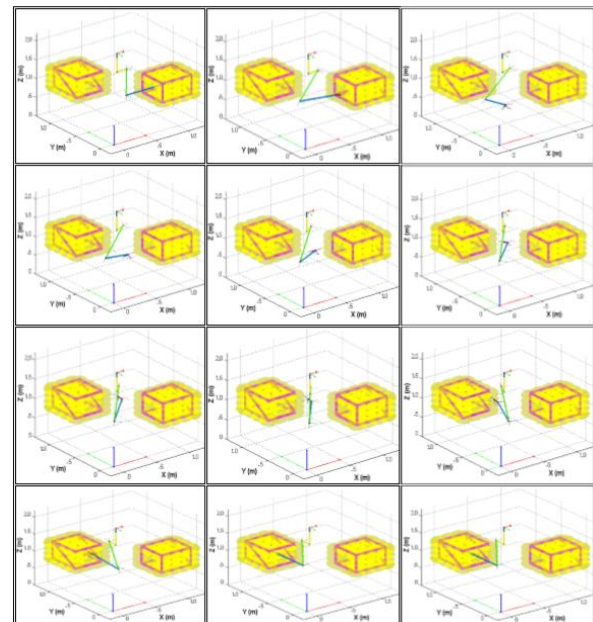


Fig. 9. Perfil de configuraciones del manipulador.

El desempeño del algoritmo y sus resultados se pueden observar en la tabla VIII, mientras que la figura 10 muestra la evolución en la disminución del error por segmentos evaluados así como la aptitud alcanzada por cada segmento.

TABLA VIII. RESULTADOS DEL ALGORITMO GENÉTICO.

| gen | segmentos | error | aptitud | aptitud promedio | tiempo |
|-----|-----------|--------|---------|------------------|--------|
| 761 | 506 | 1.0623 | 0.9119 | 0.9413 | 9.4593 |

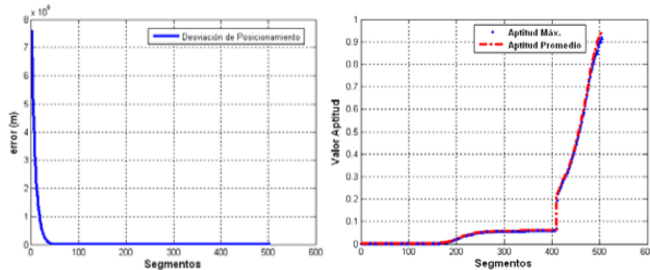


Fig. 10. Evolución de la aptitud y el error por cada segmento.

V. CONCLUSIONES

Dirigir manipuladores para alcanzar sus objetivos, implica buscar conjuntos de desplazamientos óptimos, que se limitan a explorar una gama limitada de soluciones que puede ser menor o igual a un desplazamiento máximo por unidad de tiempo para determinar no solo posiciones angulares, también velocidades, aceleraciones y torques necesarios para que los manipuladores puedan describir trayectorias deseadas. Esto evita la necesidad de un proceso de programación sincronizado para trazar rutas seguras y evitar alguna colisión.

El algoritmo planeador evoluciona a partir de un enfoque puramente genético y las trayectorias se obtuvieron en base al rendimiento del algoritmo dado por la función aptitud, el proceso de planeación de trayectorias explota el hecho de que los manipuladores tienden a moverse de manera uniforme en algunas etapas de su trayectoria.

Durante este movimiento uniforme, el esquema del cromosoma $\Delta\theta$'s es idéntico o muy próximo al que se acaba de determinar, por lo tanto mediante la hipotética mejor solución colocada en la población inicial, el siguiente conjunto mueve al manipulador de manera uniforme en ese instante y el número necesario de las generaciones para producir una nueva actualización por parte del planeador disminuye, reduciendo el

tiempo de ejecución global, si el manipulador tiene que cambiar de dirección, la inserción de $\Delta\theta$'s en el paso anterior no genera ningún problema, ya que la aptitud para este cromosoma se evalúa como para cualquier conjunto de la población por lo que se reduce el tiempo de ejecución del proceso general.

Debido a que los manipuladores industriales se mueven de manera estable, perturbaciones como fricción viscosa o de Coulomb disminuyen por las acciones de control presentes en el controlador del robot que reducen las perturbaciones naturales del sistema dinámico, en este sentido, la planeación de trayectorias para manipuladores robóticos es un problema que manifiesta un gran número de estrategias para resolverla, las herramientas matemáticas que se emplean para su aplicación y solución en forma sencilla resultan de gran ventaja.

AGRADECIMIENTOS

Los autores de este trabajo agradecen el apoyo al Instituto Politécnico Nacional en conjunto con la Secretaría de Investigación y Posgrado (proyecto SIP-20181228) por los medios otorgados para realizar esta investigación.

REFERENCIAS

- [1] Montesdeoca, J., Toibeo, J., Carelli, R. "Human-Robot Interaction, Evolution, Advances and New Challenges", *IEEE 2017 XVII Workshop on Information Processing and Control (IPIC)*, 1-4, 2017.
- [2] Khatib, O. "Real-time obstacle avoidance for manipulators and mobile robots" *International Journal of Robotics Research*, 90-98, 1986
- [3] Barrientos, A., et al. *Fundamentos de Robótica*. Segunda edición, Mc Graw Hill, Madrid, España, 2007.
- [4] Sahul, S., Biswall, B., "A Novel Method for Representing Robot Kinematics using Quaternion Theory", *IEEE Sponsored Conference on Computational Intelligence, Control and Computer Vision in Robotics and Automation*, 75-82, 2008
- [5] Cea, C. Ramírez-Gordillo, J., "Solución Cinemática Compacta Aplicada a Manipulador Robótico de 5 GDL", *Nuevos Avances en Robótica y Computación*, 109-114, 2015.
- [6] Ramírez-Gordillo, J., Merchán-Cruz E., "Desarrollo de una Nueva Solución Compacta a la Cinemática de Manipuladores Robóticos basada en Cuaterniones Duales", *Revista iberoamericana de automática e informática industrial*, 334-344, 2011.
- [7] Renan, A. C. "Obtención de la ecuación de Euler Lagrange utilizando vectores base y vectores recíprocos", *Ingeniería Revista Académica*, Universidad Autónoma de Yucatán, 17-22, 2004.

Desarrollo de un algoritmo genético para análisis espacial de conglomerados con formas irregulares.

Luis Fdo. Ruiz M., Israel M. Santillán M., Jorge E. Luna T., Luis A. Cárdenas F.
Tecnológico Nacional de México

Instituto Tecnológico de La Paz, La Paz, B.C.S.

Email: luisfe_9322@hotmail.com, israel.santillan@itlp.edu.mx, jelt0024@gmail.com, armando.cardenas@itlp.edu.mx

Abstract—En este trabajo se presenta un algoritmo genético para análisis espacial de conglomerados con formas irregulares. El algoritmo aplica técnicas estadísticas de exploración espacial para identificar áreas geográficas consideradas como conglomerados. Con el objetivo de mejorar la eficiencia del algoritmo se paralelizaron sobre GPGPU con tecnología CUDA los módulos más relevantes del mismo. Como resultados de los experimentos realizados se lograron identificar conglomerados estadísticamente significativos y con un alto nivel de confianza al utilizarse datos de cáncer de mama de la ciudad de La Paz, B.C.S.

I. INTRODUCCIÓN

El análisis espacial de conglomerados permite identificar aquellas áreas geográficas que presentan un mayor número de ocurrencias de un evento que los esperados estadísticamente. Al identificar patrones en una región de la ocurrencia de un evento en específico, ya sea, una enfermedad, una plaga, la demanda de un producto o cualquier otro; se puede ligar tal incidencia del evento con los posibles factores que lo propician.

Un conglomerado es la ocurrencia de más casos que los esperados para una ubicación geográfica, período de tiempo o ambos. La diferencia entre una alarma y un verdadero conglomerado, es que la primera es una noción empírica de que están ocurriendo demasiados casos, y el conglomerado propiamente dicho es la ocurrencia real y significativa de más casos que los esperados, basados en un sustento científico[1].

Existen diversos métodos que permiten realizar un análisis espacial de conglomerados. La mayoría de estos, son eficientes para la identificación de conglomerados con formas circulares o elípticas, pero existen muy pocos métodos capaces de detectar conglomerados con formas irregulares, y los que lo hacen generalmente requieren una gran cantidad de tiempo de ejecución[2]. Uno de los métodos más utilizados, es el propuesto por Kulldorff[3], que permite realizar análisis estadístico-espacial para formas circulares o elípticas. Dicho método ha sido aplicado en una gran variedad de estudios epidemiológicos para la detección de conglomerados asociados a las zonas de riesgo. Este método está basado en la generación de una serie de ventanas circulares sobre el área de estudio para detectar los conglomerados más significativos, y después aplica el simulación tipo Monte Carlo para evaluar la significancia estadística de los conglomerados localizados. Una desventaja importante de este método es que no es capaz de detectar conglomerados con formas irregulares, con lo cual no se delimita correctamente el área geográfica del problema.

El software FleXScan[4] es capaz de detectar conglomerados con formas irregulares que se encuentren dentro de, relativamente, pequeños vecindarios de cada región. Dicho método trabaja bien para conglomerados de tamaño pequeño a moderado, considerando treinta regiones como máximo. Este método no es realizable para conglomerados de mayor tamaño y se requiere de un algoritmo más eficiente[4].

La mayoría de las áreas geográficas como ríos, ciudades, montañas, entre muchos otros; tienen formas irregulares, y es conveniente que al momento de aplicar un análisis espacial los conglomerados que se obtengan tengan formas irregulares. De esta manera las regiones señaladas por el conglomerado estarán mejor delimitadas en caso de que se pretendan tomar acciones respecto al evento de estudio.

II. ESTADÍSTICA DE EXPLORACIÓN ESPACIAL

La estadística de exploración espacial es una técnica que utiliza la ubicación geográfica de los casos, y los analiza mediante métodos estadísticos[5]. La estadística de exploración espacial permite identificar la existencia y ubicación geográfica de conglomerados de acuerdo a un cierto nivel de significancia estadística elegida[6]. El método consiste en hacer una exploración en el área geográfica de interés trazando espacios o ventanas. Para cada ventana se analiza el riesgo de que ocurra el evento de interés dentro de ésta, comparándolo con el resto de la entidad.

III. ALGORITMO GENÉTICO

Es una técnica de optimización global que pertenece al área de la computación evolutiva y que se basa en la teoría de la evolución de las especies. Inicialmente estos algoritmos generan un conjunto de individuos aleatoriamente que conforman a la población. Dichos individuos están esparcidos dentro del espacio de búsqueda. Entonces se aplican tres operadores genéticos: selección, cruce y mutación, que permiten hacer que la población evolucione a lo largo de las generaciones del algoritmo y de esta manera, converja hacia el óptimo global. El operador genético de selección funciona como un filtro de manera similar a lo que es la selección natural que vemos en los sistemas biológicos. Aquellos individuos menos aptos se eliminan, y los individuos con una mejor aptitud tienen más probabilidad de mantenerse y pasar a la siguiente generación. El cruce permite que los individuos de la población intercambien información de manera similar a lo que es la

reproducción sexual en los organismos naturales. Se forman pares de individuos que pasaron por la fase de selección y se forman nuevos individuos que integran información de los dos individuos que los formaron. La mutación es usada para cambiar aleatoriamente la información o características de algún individuo de la población. Después de que estos operadores genéticos son aplicados a la población inicial se obtiene una nueva población y el contador de la generación incrementa en uno. Este mismo proceso se repite hasta que se alcanza un número de generaciones o se cumple un criterio de convergencia del algoritmo que inicialmente haya sido indicado[7].

IV. SIMULACIÓN MONTE CARLO

Los métodos de simulación Monte Carlo son una clase de técnicas computacionales basadas en generar sintéticamente variables aleatorias para deducir el alcance de la distribución de probabilidad. Estos métodos se aplican en un amplio rango de problemas de ingeniería y tienen ventajas comparados a métodos de enfoque determinístico como lo es su facilidad de implementación[8].

V. CUARTILES

Son los tres valores que dividen un conjunto de datos ordenados en cuatro partes porcentualmente iguales. La diferencia entre el tercer cuartil y el primer cuartil se conoce como rango intercuartil[9]. Con la aplicación del rango intercuartil podemos ubicar valores atípicos, ya sean atípicos leves o atípicos extremos.

VI. DATOS ATÍPICOS

Un dato atípico es una observación que se aleja tanto del resto de las observaciones como para despertar sospechas de que fue generada por un factor diferente. Cuando en un proceso del cual se extraen datos, hay un comportamiento inusual, resulta en la creación de datos atípicos[10]. El principal problema es que son elementos que pueden no ser representativos de la población pudiendo distorsionar seriamente el comportamiento de los contrastes estadísticos. Pero también pueden ser indicativos de las características de una parte de la población y probablemente, una señal de alerta. Entre los diferentes tipos de datos atípicos, se encuentran los datos extraordinarios para los que el investigador no tiene explicación. En estos casos lo mejor que se puede hacer es replicar el análisis con y sin dichas observaciones con el fin de analizar su influencia sobre los resultados. Si dichas observaciones tienen efecto en el análisis muy probablemente haya un factor que esté propiciando el fenómeno.

VII. MÉTODO

El método propuesto funciona principalmente en torno al algoritmo genético, que permite llevar a cabo la exploración en el espacio para detectar aquellas áreas geográficas en las cuales, muy probablemente, la ocurrencia de los casos se haya dado por algún factor que la propicia y no solo por casualidad. Dicha búsqueda hace uso de métodos de estadística

de exploración espacial para seleccionar o descartar regiones geográficas, con el objetivo de mantener zonas, a través de las generaciones del algoritmo, que sean candidatas a ser consideradas un conglomerado. Para delimitar el área del conglomerado obtenido se aplica un algoritmo de trazado para formas irregulares obtenido del proyecto "Desarrollo de un algoritmo genético para análisis espacial de conglomerados epidemiológicos con formas irregulares"[11].

A. Paralelización de función de cálculo de aptitud y operadores del algoritmo genético

El método de cálculo de aptitud y los operadores de cruce y mutación del algoritmo genético están paralelizados basados en la técnica de paralelismo de grano grueso. Estos algoritmos paralelizados son llamados por los individuos de la población del algoritmo genético cada vez que lo requieren y solo hay comunicación entre ellos cada generación. El tiempo de ejecución del algoritmo genético es afectado considerablemente por estos módulos ya que, al estar paralelizados, pueden ser llamados simultáneamente por distintos individuos de la población.

B. Funcionamiento del algoritmo genético

Inicialmente el algoritmo genético genera un conjunto de posibles soluciones del fenómeno en estudio de manera aleatoria. Estas posibles soluciones, también llamadas ventanas, son áreas de estudio que se conforman de varias regiones involucradas en el fenómeno de estudio, y que a su vez, estas regiones tienen población y pueden o no tener incidencia de casos. Como se muestra en la Figura 1, cada punto color azul representa una ubicación geográfica de la cual se conocen las coordenadas, su número de habitantes y el número de casos u ocurrencias del problema de estudio en dicha área. Se representa con un punto de color rojo las regiones en las cuales se tiene un número de población pero no hay incidencia de casos.

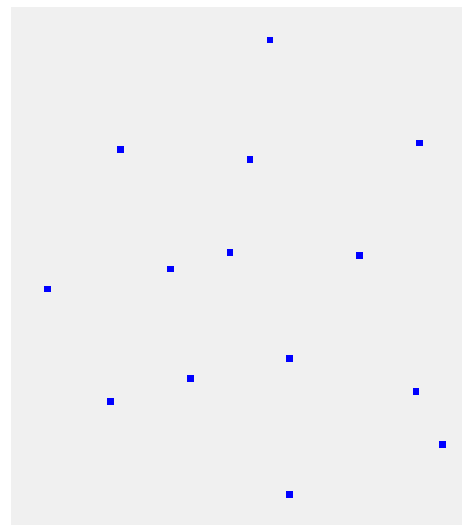


Fig. 1. Ventana del algoritmo genético.

Después el algoritmo genético aplica el cálculo de aptitud a cada una de las ventanas. Para éste cálculo se optó por usar la fórmula de verosimilitud, que se muestra en (1).

$$\left(\frac{c}{E[c]}\right)^c \left(\frac{C-c}{C-E[c]}\right)^{C-c} \quad (1)$$

Donde C es el número total de casos, c es el número de casos observado dentro de la ventana y $E[c]$ es el número de casos esperados dentro de la ventana bajo la hipótesis nula. El algoritmo procede con la etapa de selección, en la cual se mantienen una cierta cantidad de ventanas mientras que otras se eliminan, todo esto en base al valor de aptitud previamente calculado. Luego el algoritmo pasa a la etapa de cruza, en la cual se generan nuevas ventanas (hijos) con características extraídas de pares de ventanas (padres). Finalmente estas ventanas seleccionadas pueden o no, dependiendo de una variable de azar, entrar en la fase de mutación. Esto es, las ventanas sufrirán un pequeño cambio en el cual pueden crecer o decrecer geográficamente (se le agregan o quitan regiones). En la Figura 2 se muestra el aspecto que tendría una ventana obtenida al azar. En la Figura 3 se ve la ventana obtenida al aplicar una mutación a la ventana de la Figura 2.

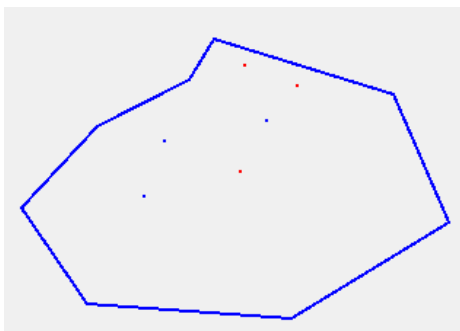


Fig. 2. Ventana formada aleatoriamente.

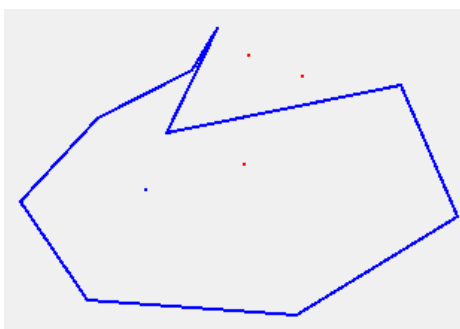


Fig. 3. Ventana de la figura 2 tras sufrir una mutación.

Se vuelven a calcular las aptitudes de las ventanas nuevas y se repite el proceso para que las ventanas evolucionen a través de cada generación del algoritmo. Hasta este punto se obtienen uno o más conglomerados que pueden tener formas irregulares.

VIII. PRUEBAS Y RESULTADOS

El algoritmo genético se probó con una base de datos de cáncer de mama de la ciudad de La Paz, B.C.S. Los parámetros de entrada para el experimento se muestran en la Tabla 1.

TABLE I
PARÁMETROS DEL ALGORITMO GENÉTICO PARA EL EXPERIMENTO.

| | |
|--------------------------|----|
| Número total de ventanas | 50 |
| Generaciones sin mejorar | 30 |

La Figura 4 muestra el conglomerado resultante de la ejecución del algoritmo genético. Se puede apreciar que el área abarcada por el conglomerado se encuentra en la parte centro de la ciudad de La Paz.

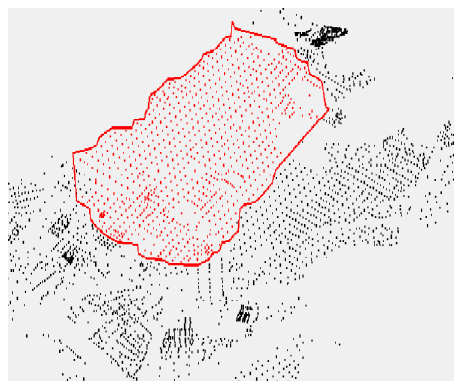


Fig. 4. Conglomerado resultante del algoritmo genético probado con datos de cáncer de mama de la ciudad de La Paz, B.C.S.

En la Tabla 2 se muestran los datos del conglomerado que arrojó el experimento.

TABLE II
DATOS DEL CONGLOMERADO RESULTANTE.

| | |
|-----------------|---------|
| Casos | 84 |
| Población | 14156 |
| Casos esperados | 55.7197 |
| Verosimilitud | 4625.31 |
| Riesgo relativo | 1.77401 |

Se aplicó el algoritmo de trazado para formas irregulares[11] en conjunto con un algoritmo de recocido simulado con la intención de delimitar de la mejor manera posible el área del conglomerado. El algoritmo de recocido simulado no tuvo los resultados que se esperaban, puesto que la mayoría de las veces no mejora la precisión con la que se delimita el área abarcada por el conglomerado. Más bien que elimina áreas que son relevantes para el problema de estudio, es decir, excluye zonas en las cuales se tiene incidencia de casos. Se pretendía aplicar esta propuesta a los conglomerados arrojados por el algoritmo genético pero debido a los malos resultados se optó por trabajar con una nueva propuesta en la cual se aplica un degradado de color a los conglomerados.

IX. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha presentado un algoritmo genético para análisis espacial de conglomerados con formas irregulares. Se pretende agregarle a dicho algoritmo, un método de degradado de color para los conglomerados y realizar pruebas con una base de datos de cáncer de mama de la ciudad de La Paz B.C.S. y una base de datos de problemas de corazón de la prefectura de Saitama, Japón. Aunque el algoritmo de trazado de formas irregulares[11] cumple con su función de delimitar el área de un conjunto de coordenadas, se pudiera buscar una alternativa para un método más preciso.

REFERENCES

- [1] CDC (Centers for Disease Control). 2001. Guidelines for investigating clusters of health events.
Sitio Web: <http://www.cdc.gov/mmwr/preview/mmwrhtml/00001797.htm>
- [2] Tai-Chi Wang, Ching-Syang Jack Yue: A binary-based approach for detecting irregularly shaped clusters. Wang and Yue International Journal of Health Geographics 2013.
- [3] Kulldorff M, Nagarwalla N: Spatial disease clusters: detection and inference. StatMed 1995.
- [4] Tango T, Takahashi K: A flexibly shaped spatial scan statistic for detecting clusters. IntJHealthGeogr 2005.
- [5] Santamaña-Ulloa, C. 2002. Análisis para la determinación de conglomerados espacio temporales en la incidencia de cáncer. Costa rica, 1990-1997. Master's thesis, Universidad de Costa Rica.
- [6] Kulldorff, M., Feuer, E. J., Miller, B. A., and Freedman, L. S. 1997. Breast cancer clusters in the northeast united states: A geographic analysis. American Journal of Epidemiology.
- [7] David A. Coley: An Introduction to Genetic Algorithms for Scientists and Engineers. World Scientific Publishing 1999.
- [8] Masayuki Yano, James Douglass Penn, George Konidaris, Anthony T. Patera: Math, Numerics and Programming (for Mechanical Engineers) 2012.
- [9] Jackie Nicholas: Introduction to Descriptive Statistics. 1999.
- [10] Charu C. Aggarwal: Outlier Analysis. 2016.
- [11] Luis Ruiz, Isaim Murillo, Enrique Luna: Desarrollo de un algoritmo genético para análisis espacial de conglomerados epidemiológicos con formas irregulares. Instituto Tecnológico de La Paz. 2016.

Implementación de un Cluster Heterogéneo de GPGPU para Algoritmo Genético

Mauricio Rojas Salazar

División de Estudios de Posgrado e Investigación
Instituto Tecnológico de La Paz
La Paz B.C.S., México
Email: mr.salazar00@gmail.com

Marco Antonio Castro Liera

División de Estudios de Posgrado e Investigación
Instituto Tecnológico de La Paz
La Paz B.C.S., México
Email: mcastro@itlp.edu.mx

Jesús Antonio Castro

División de Estudios de Posgrado e Investigación
Instituto Tecnológico de La Paz
La Paz B.C.S., México
Email: jcastro@itlp.edu.mx

Iliana Castro Liera

División de Estudios de Posgrado e Investigación
Instituto Tecnológico de La Paz
La Paz B.C.S., México
Email: icastro@itlp.edu.mx

Resumen—En el presente artículo se describe la implementación de un cluster heterogéneo de Unidades de Procesamiento Gráfico de Propósito General (GPGPU, por sus siglas en inglés) en el cual se ejecutó un algoritmo genético para pruebas de eficacia y eficiencia.

Para el cluster se implementó OpenMPI y para la programación en la GPU se utilizó OpenCL.

Index Terms—OpenCL, OpenMPI, GPGPU, clúster, algoritmo genético, kernel, host.

I. INTRODUCCIÓN

En el computo paralelo existe la tendencia del uso de las Unidades de Procesamiento Gráfico de Propósito General (GPGPU, por sus siglas en inglés).

Mientras los procesadores cuentan con una cantidad muy reducida de ALUs, las tarjetas gráficas poseen una cantidad mucho mayor, permitiendo hacer en una fracción del tiempo lo que a una CPU le tomaría hacer en horas.

Otra práctica que existe es la creación de clusters utilizando equipos con tarjetas graficas del mismo fabricante, arquitectura comúnmente llamada cluster homogéneo que combina las ventajas de un cluster con las ventajas de las GPGPUs.

En el Instituto Tecnológico de La Paz se han realizado varios proyectos de investigación con procesamiento paralelo en GPGPU. Los más relevantes con relación a este trabajo son:

1. La implementación de un AG y un PSO en una GPGPU, utilizando CUDA y una tarjeta Nvidia[1].
2. La implementación de un AG y un PSO en un cluster homogéneo, utilizando CUDA y tarjetas gráficas Nvidia[2].

Actualmente no existe mucho trabajo sobre clusters heterogéneos, de aquí la iniciativa propuesta en el presente artículo.

Para el desarrollo del cluster heterogéneo se implementó OpenMPI para el enlace y la comunicación de los equipos.

En el caso de la programación para la GPGPU se empleó OpenCL. Esta herramienta consta de una interfaz y un lenguaje de programación, brindando un estándar en el desarrollo del código. Las aplicaciones fueron desarrolladas en lenguaje C y, al momento de compilación, se utilizan las librerías de cada fabricante, lo que permite usar el mismo código para tarjetas de diferentes marcas.

II. MARCO TEÓRICO

A. OpenCL

Open Computing Language (OpenCL) es un framework de programación heterogénea que es manejado por el consorcio de tecnología sin fines de lucro Khronos, con la participación de compañías como Intel, ARM, AMD, NVIDIA, QUALCOMM, Apple y muchos otros[3]. Con OpenCL es posible crear aplicaciones que pueden ser ejecutadas en una gran variedad de dispositivos de diferentes fabricantes. Es compatible con una amplia gama de niveles de paralelismo de manera eficiente a sistemas homogéneos o heterogéneos, de uno o varios dispositivos que consten de CPU, GPU y otros tipos de dispositivos. La definición de OpenCL ofrece un lenguaje del lado del dispositivo y una capa de administración del host para los dispositivos en un sistema.

Dentro de las arquitecturas admitidas se incluyen: CPU multinúcleo, procesadores vectoriales (tales como la GPU) y dispositivos paralelos de grano fino (como FPGA). Lo más importante es que la compatibilidad multiplataforma de OpenCL lo convierte en un excelente modelo de programación para que los desarrolladores lo aprendan y utilicen, con la confianza de que continuará estando ampliamente disponible en los próximos años, con alcance y aplicabilidad cada vez mayores[4].

B. OpenMPI

OpenMPI es una implementación de la interfaz de paso de mensajes de código abierto, desarrollada y mantenida por un

consorcio de socios académicos, de investigación y de la industria. Por lo tanto, Open MPI puede combinar la experiencia, las tecnologías y los recursos de toda la comunidad informática de alto rendimiento, para crear la mejor biblioteca de MPI disponible. OpenMPI ofrece ventajas para los proveedores de sistemas y software, desarrolladores de aplicaciones e investigadores de informática[5].

1) *MPI*: MPI (Message-Passing Interface) es una especificación de la interfaz de biblioteca para el paso de mensajes. Todas las partes de esta definición son significativas. MPI aborda principalmente el modelo de programación paralela de transmisión de mensajes, en el que los datos se trasladan del espacio de direcciones de un proceso a otro mediante operaciones cooperativas en cada proceso. Las extensiones del modelo clásico de paso de mensajes se proporcionan en operaciones colectivas, operaciones de acceso a memoria remota, creación de procesos dinámicos y E/S paralelas. MPI es una especificación, no una implementación. Esta especificación es para una interfaz de biblioteca; MPI no es un lenguaje, y todas las operaciones MPI se expresan como funciones, subrutinas o métodos. El estándar ha sido definido a través de un proceso abierto por una comunidad de vendedores de computo paralelo, informáticos y desarrolladores de aplicaciones[6].

C. Algoritmo Genético

Los algoritmos genéticos (AG) fueron introducidos por Holland (1975) e imitan los principios básicos de la naturaleza formulados por Darwin (1859) y Mendel (1866). Dichos principios básicos son:

1. Existe una población de soluciones. Las propiedades de una solución se evalúan según el fenotipo y los operadores de variación se aplican al genotipo. Algunas de las soluciones se eliminan de la población si el tamaño de la población supera un límite superior.
2. Los operadores de variación crean nuevas soluciones con propiedades similares a las existentes. El operador principal de búsqueda es la recombinación y la mutación sirve como operador de fondo.
3. Los individuos de mejor calidad se seleccionan con mayor frecuencia para la reproducción, mediante un proceso de selección.[7]

III. METODOLOGÍA

A. Desarrollo de un Framework para OpenCL

En OpenCL la mayoría de las funciones tienen varios parámetros, haciéndolas altamente configurables. Esto es debido a que esta herramienta no es única y exclusivamente para las GPUs, sino para muchos otros dispositivos de procesamiento. Por estas dos características del lenguaje es que en este trabajo se desarrolló un framework para facilitar la tarea de programación, disminuyendo en gran medida el cuerpo resultante del programa y el número de parámetros con los cuales se trabaja. Esto se llevó a cabo después de observar que el código suele volverse repetitivo y que muchos parámetros

de los métodos suelen permanecer iguales, puesto que son para propósitos específicos.

B. Desarrollo de un RNG para OpenCL

Hasta el momento en que se elaboró este trabajo, OpenCL no cuenta con soporte para la generación de números pseudoaleatorios, por lo cual fue necesario desarrollar un RNG (Random Number Generator), que se logró extrayendo el código fuente referente a la clase RNG de la librería GSL de C (concretamente, el generador que se tomó para trabajar es el basado en mt19937). Se utilizó el código de este generador como base y se ajustó para implementarlo con OpenCL y que pudiera ejecutarse en el lado del kernel.

C. Algoritmo Genético a Implementar

En el algoritmo 1 se puede apreciar, en pseudocódigo, el flujo del programa, indicando al final de cada línea en cual equipo se ejecuta la mayor parte (si en la CPU o en la GPU).

Algoritmo 1 Pseudocódigo general del AG

```

1: inicio AG
2: se genera la población inicial (GPU)
3: se ordenan los individuos de mejor a peor aptitud (GPU)
4: para toda i (Generaciones/Periodo migratorio entre nodos) (CPU) hacer
5:   para toda i en (Generaciones/Periodo migratorio entre bloques) (CPU) hacer
6:     LOGICA_AG (GPU)
7:     se ordenan los individuos de mejor a peor aptitud (GPU)
8:     migración copia a los mejores individuos del bloque siguiente en los peores del bloque actual (GPU)
9:   fin para
10:  búsqueda del mejor de cada bloque por reducción binaria (GPU)
11:  búsqueda del mejor individuo de todos los bloques (CPU)
12:  comparte el mejor resultado obtenido con los demás nodos (CPU)
13:  recibe el mejor resultado obtenido en otro nodo y valida si es mejor que el mejor local (CPU)
14: fin para
15: fin AG
    
```

En el algoritmo 2 se puede apreciar el flujo de la lógica del AG.

Algoritmo 2 Pseudocódigo de la lógica del AG

```

1: inicio LOGICA_AG
2: selección por torneo
3: cruce
4: mutación
5: búsqueda y actualización del mejor del bloque
6: fin LOGICA_AG
    
```

IV. RESULTADOS

Las funciones de prueba que se utilizaron en la implementación del AG son las propuestas por Suganthan et al [8] para la "Competencia de Computación Evolutiva 2005".

De la batería de 5 funciones unimodales y 7 funciones básicas propuestas, se seleccionaron las siguientes:

- Rastrigin:

$$f(x) = \sum_{i=1}^D (x_i^2 - 10 \cos(2\pi x_i) + 10) \quad (1)$$

D: dimensiones

Propiedades:

- Multimodal, Desplazable, Separable, Escalable.
- Gran cantidad de óptimos locales.
- $X \in [-5, 5]^D$

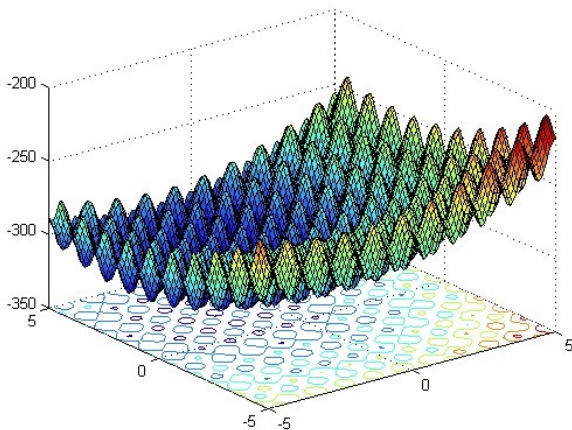


Fig. 1: Función de Rastrigin.

- Ackley:

$$f(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i)\right) + 20 + e \quad (2)$$

D: dimensiones

Propiedades:

- Multimodal, Rotable, Desplazable, No-Separable, Escalable.
- Óptimo global en frontera.
- Si se inicializa la población cerca de la frontera, el problema se resuelve fácilmente.
- $X \in [-32, 32]^D$

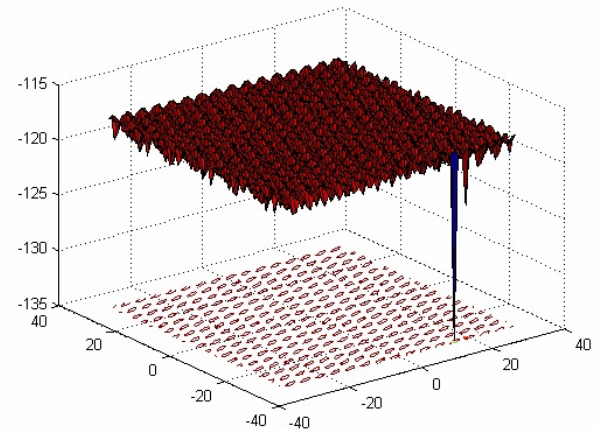


Fig. 2: Función de Ackley.

| Equipo 1 | |
|------------|--|
| GPU | AMD Radeon R9 270X con 4GB de memoria global, 1280 núcleos y una frecuencia de reloj de 1050 MHz |
| Procesador | AMD FX(tm)-8320 Eight-Core Processor @ 3.50GHz x 4 |
| RAM | 8 GB a 1600 MHz |
| Librería | OpenCL 1.2 |
| SO | Linux Mint 17.3 Cinnamon 64-bit |

Tabla 1. Especificaciones del equipo 1

| Equipo 2 | |
|------------|---|
| GPU | GeForce GT 730 con 2GB de memoria global, 96 núcleos y una frecuencia de reloj de 700 MHz |
| Procesador | Intel core i7-4790 CPU @ 3.60 GHz x 4 |
| RAM | 8 GB a 1600 MHz |
| Librería | OpenCL 1.1 |
| SO | Linux Mint 17.3 Cinnamon 64-bit |

Tabla 2. Especificaciones del equipo 2

En la optimización de las funciones propuestas, para espacios de búsqueda de 30 dimensiones, se utilizaron 14 bloques con 64 individuos, un periodo migratorio del 10% del total de generaciones, probabilidad de cruce de 0.16, probabilidad de mutación de 0.7, tamaño del torneo de 2 y se obtuvieron los siguientes resultados para 10 muestras:

| Generaciones | Aciertos | Tiempo(s) |
|--------------|----------|-----------|
| 400 | 9 | 8.165438 |

Tabla 3. Resultados función Rastrigin

| Generaciones | Aciertos | Tiempo(s) |
|--------------|----------|-----------|
| 400 | 10 | 7.794402 |

Tabla 4. Resultados función Ackley

V. CONCLUSIONES

Es posible implementar aplicaciones de cómputo paralelo por medio de un cluster heterogéneo integrado por tarjetas de gráficos de diferentes fabricantes, utilizando OpenCL como lenguaje de programación y OpenMPI como herramienta de comunicación entre los nodos del cluster. El punto clave reside en que el software propietario se enlaza en tiempo de ejecución, por medio de las librerías proporcionadas por el fabricante de cada tarjeta de gráficos (GPU). Estos resultados reviven la idea que dio origen a los clusters de computadoras de bajo costo: utilizar equipos en desuso, pero ahora incluyendo en el proceso las tarjetas de gráficos que estos equipos contienen.

REFERENCIAS

- [1] I. Castro Liera, "Paralelización de Algoritmos de Optimización Basados en Poblaciones Usando GPGPU," p. 112, 2011.
- [2] J. I. Mercado Bareño, "Implementación de un Cluster de GPGPU para Algoritmos de Optimización," no. 612, 2015.
- [3] R. Tay, *OpenCL Parallel Programming Development Cookbook*. 2013.
- [4] B. Gaster, L. Howes, D. R. Kaeli, P. Mistry, and D. Schaa, *Heterogeneous computing with OpenCL, 2nd Edition*. 2011.
- [5] M. P. I. Open, "Open source high performance computing," 2012.
- [6] E. Lusk, S. Huss, B. Saphir, and M. Snir, "MPI: A message-passing interface standard," 2009.
- [7] F. Rothlauf, *Design of Modern Heuristics - Principles and Application*. 2011.
- [8] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y. P. Chen, A. Auger, and S. Tiwari, "Problem Definitions and Evaluation Criteria for the CEC 2006 Special Session on Constrained Real-Parameter Optimization," *KanGAL*, no. May, pp. 251–256, 2005.

Sensor Linearization using Linear Genetic Programming

Humberto Velasco Arellano*, Martín Montes Rivera* and José Eder Guzmán Mendoza*

*Universidad Politécnica de Aguascalientes

Maestría en Ciencias en Ingeniería

Email: mc160006@alumnos.upa.edu.mx

Email: martin.montes@upa.edu.mx

Email: jose.guzman@upa.edu.mx

Abstract—There are different sensors used in several industrial applications either for controlling, measuring or warning in many kinds of processes, but a recurrent problem is its linearization. Classical methods for this propose allow to interpret a non-linear behavior or linearize it in specific ranges, on the other hand genetic programming allows the obtaining of unknown equations in other branches of engineering linearizing non-linear functions. In this paper the use of linear genetic programming is proposed for obtaining equations that linearize sensors in their entire measuring range and is tested with two different sensors, the proposed method has better results than other explored techniques.

Index Terms— *Genetic programming, Sensor linearization, Synthesizing Tables.*

I. INTRODUCTION

Measuring is a comparison of two different magnitudes to classify and identify, quantities or qualities, in order to interpret the world as the human does through his senses, but standardizing the results to get similar measures whit similar conditions. [1].

In engineering, measuring is made with electronic sensors when developing automatic systems that interpret and display information in human-machine interfaces, allowing to ingress data, read a variable and then taking control actions [2].

Majority of sensors have no linear responses producing outputs that are difficult to interpret, this demands an increasing of the system complexity for providing a reliable measure, but when sensor linearization is performed then measuring is allowed with a certain level of reliability and without increasing the system complexity. [3].

There are several applications where linearization is performed like in circuit linearization in [4], which is also used in microcontrollers [5] [6], neuronal networks [7] [8] [9], FPGA's as system emulators [10] [11] and fuzzy logic systems [12].

The contribution in automation problems has produced different patents [13] [14] around the linearization problem.

In this work, sensor linearization is performed by using Genetic Programming (GP in relative) as an alternative for classical methods.

The selection of GP as optimization algorithm is supported on the capabilities found in related works, like linearizing transformation, where the GP is able to get a regression for predicting the behavior of a symbolic model [15]. Another related work where identification of non-linear dynamic systems is made to apply control compensation in [16]. Finally is considered the algorithm in [17] for obtaining equations when synthesizing water steam tables.

II. THEORETICAL FRAMEWORK

This section defines the concepts and background methodologies considered throughout this work.

A. Transducer and Sensor

A transducer is a device that transforms any physical signal to other kind with different physical properties, in other words, an element that transforms an input energy [3].

B. Sensor Linearization and Linearity

The linearity is a grade of coincidence between the reaction curve of a sensor and the expected straight line for representing a proportional input-output relation, There are different kinds of linearity, depending on the comparison line used:

- **Independent Linearity:** Line is obtained by recursive least square reduction.
- **Zero Adjusted Linearity:** Line is obtained by recursive least square reduction, but forcing a cross on zero.
- **Terminal Linearity:** Line is obtained considering the output without input and the maximum theoretical output.
- **Linearity Across Extremes:** Line is obtained using the real minimal and maximum values get by experiment.
- **Theoretical Linearity:** Line is proposed by the sensor manufacturer.

All of them work as comparison approach with the sensor response [3].

Sensor linearization is a signal conversion from a non-linear to linear behavior [2].

C. Genetic Programming

Evolutionary algorithms are population-based meta-heuristic optimization algorithms that use biological inspired mechanisms like mutation, crossover, natural selection, and survival of the fittest for iterative improve of solution candidates. The cycle of evolutionary algorithms follows the Darwin natural selection principles that are described in Fig. 1 [17], [18].

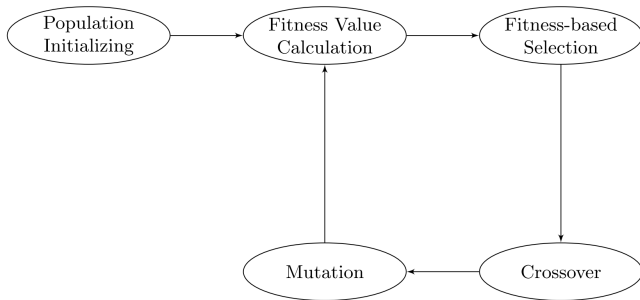


Fig. 1. Evolutionary algorithms behavior with courtesy of [17]

GP capabilities include finding the program structure that better satisfies the objective function minimization [18].

Individuals of GP are commonly represented by a tree structure, but linear GP work with structures that demand less computational power than tree structures and equation structures are linear GP individuals based on equations, which are used in this paper. [19].

III. SIMULATION TEST

The followed methodology for testing the proposed algorithm in the linearization of sensors with linear programing was performed in MATLAB 2016b 64 bits, which executed in a Windows 10 PC.

Two different no linear sensors were used during the test, first one is the Sharp IR GP2Y0A21YK sensor like the one displayed on Fig. 2 [20].



Fig. 2. Sharp IR GP2Y0A21YK sensor [20]

The other one was a thermocouple type J. This kind of sensor is electrically binded by two wires of different

materials producing resistance variation between the sensor therminals with temperature changes Fig. 3 shows the block diagram of this sensor[19].

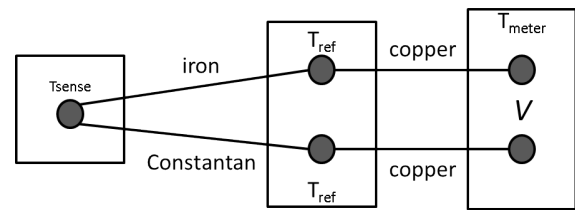


Fig. 3. Block diagram of thermocouple J[19]

The logarithmic response of Sharp IR GP2Y0A21YK is the shown in Fig. 4.

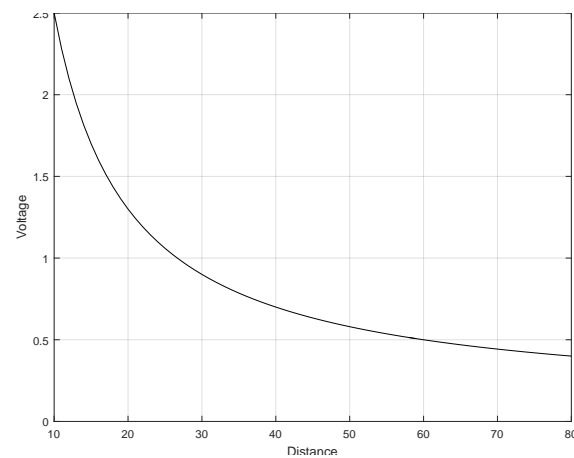


Fig. 4. Sharp sensor behavior

Thermocouple type J has a similar response, where x axis is the temperature and y axis is voltage as shown the Fig. 5 [21].

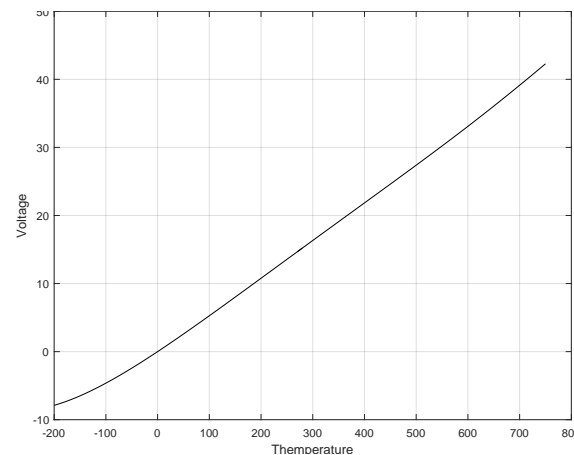


Fig. 5. Thermocouple sensor behavior [21]

IV. METHODOLOGY

The GP methodology implemented for linearizing the sensors is the following:

- GP is based on a population of 1001 base equation elements with 3 genes each one.
- A certain percentage of the population is submitted into a tournament, the tournament process is selected in random way as is showing in the figure 6.

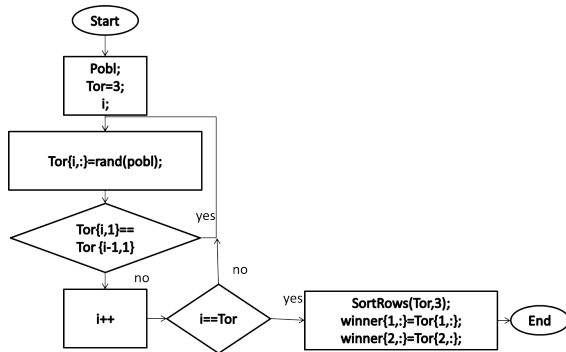


Fig. 6. flow diagram, tournament

- Once fitness is calculated, then those with the smallest fitness (winners) are crossed. This operation is performed by randomly selecting a crossover point where the winners will be split and mixed to generate a new individual, the process is shown in figure 7.

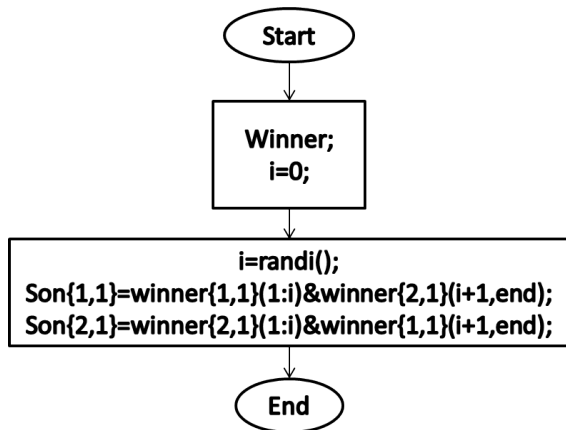


Fig. 7. flow diagram, corssover

- If the mutation rate is surpassed the offspring is mutated to get genetic diversity, here is selected one position randomly and the allele with this position will be toggled as the inverse of the selected allele, the flow diagram in the figure 8 shows this process.

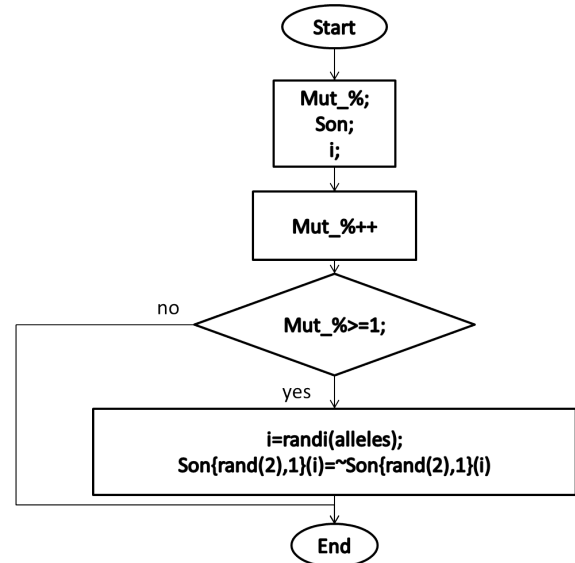


Fig. 8. flow diagram, mutation

- The entire process is repeated until 10000 generations are reached, deleting the worst individuals (with higher values of fitness) in order to maintain ecological stability, since cost function is the error measured with a desired straight line.

- The best candidate solution is the equation that better linearizes the reaction curve of the sensor.

Reaction curve used during fitness evaluation was taken from the sensor datasheet in both cases, described in [20] and [19]. In the thermocouple datasheet the resistance-temperature table is shown. The values of 0 offset is used in this study, but in the case that there exists any offset due to the connectors, the model must be trained again [21].

The related data for the sharp sensor is shown in Table I and for thermocouple in the figure 5 or in its datasheet.

V. SIMULATION RESULTS

The behavior of the sensors shown in figure 4 and 5 was taken into account to identify the distance and temperature as dependent variables for the SHARP and thermocouple sensors respectively as well as voltage and resistance as independent variables.

The parameters used in GP algorithm with the best result for linearizing SHARP sensor are shown in Table II.

TABLE I
SENSOR SHARP TABLE OF VALUES

| sharp 2y0a2l | | | | | |
|----------------|-------------|----------------|-------------|----------------|-------------|
| distancia (cm) | voltaje (V) | distancia (cm) | voltaje (V) | distancia (cm) | voltaje (V) |
| 10 | 2.5 | 34 | 0.80588235 | 58 | 0.5137931 |
| 11 | 2.28181818 | 35 | 0.78571429 | 59 | 0.50677966 |
| 12 | 2.1 | 36 | 0.76666667 | 60 | 0.5 |
| 13 | 1.94615385 | 37 | 0.74864865 | 61 | 0.49344262 |
| 14 | 1.81428571 | 38 | 0.73157895 | 62 | 0.48709677 |
| 15 | 1.7 | 39 | 0.71538462 | 63 | 0.48095238 |
| 16 | 1.6 | 40 | 0.7 | 64 | 0.475 |
| 17 | 1.51176471 | 41 | 0.68536585 | 65 | 0.46923077 |
| 18 | 1.43333333 | 42 | 0.67142857 | 66 | 0.46363636 |
| 19 | 1.36315789 | 43 | 0.65813953 | 67 | 0.45820896 |
| 20 | 1.3 | 44 | 0.64545455 | 68 | 0.45294118 |
| 21 | 1.24285714 | 45 | 0.63333333 | 69 | 0.44782609 |
| 22 | 1.19090909 | 46 | 0.62173913 | 70 | 0.44285714 |
| 23 | 1.14347826 | 47 | 0.6106383 | 71 | 0.43802817 |
| 24 | 1.1 | 48 | 0.6 | 72 | 0.43333333 |
| 25 | 1.06 | 49 | 0.58979592 | 73 | 0.42876712 |
| 26 | 1.02307692 | 50 | 0.58 | 74 | 0.42432432 |
| 27 | 0.98888889 | 51 | 0.57058824 | 75 | 0.42 |
| 28 | 0.95714286 | 52 | 0.56153846 | 76 | 0.41578947 |
| 29 | 0.92758621 | 53 | 0.55283019 | 77 | 0.41168831 |
| 30 | 0.9 | 54 | 0.54444444 | 78 | 0.40769231 |
| 31 | 0.87419355 | 55 | 0.53636364 | 79 | 0.40379747 |
| 32 | 0.85 | 56 | 0.52857143 | 80 | 0.4 |
| 33 | 0.82727273 | 57 | 0.52105263 | | |

TABLE II
INISIALIZTION VALUES FOR GP ALGORITHM IN SHARP SENSOR CASE

| Paramether | value |
|--------------------|-------|
| Population size | 1001 |
| Operation quantity | 10 |
| Deepness | 3 |
| Generations | 10000 |
| Mutations | 35 |
| Seed | 1 |
| Tournament | 10 |
| Elitism | 1000 |

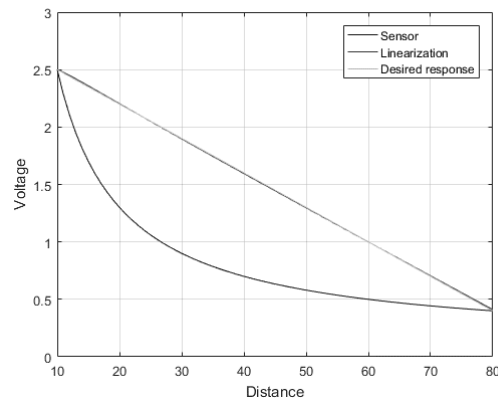


Fig. 9. GP solution behavior for sharp sensor linearization

The resulting equation for linearizing SHARP sensor is shown in equation 1, where the variable D is the distance and V is the voltage referred to the Table I.

$$V = \frac{(7.3679 D - 73.598) (0.5 D - 40.204)}{126.48 D - 63.0} \quad (1)$$

This solution gives an efficiency of 0.004625038334 volts of average error, which is 0.3093% of relative error, the behavior of this solution is shown in Fig. 9

Fitness evolution across generations is shown in Fig. 10, where cost function is minimized as expected by getting better results across generations, to finally get a stable behavior caused by a local minimum.

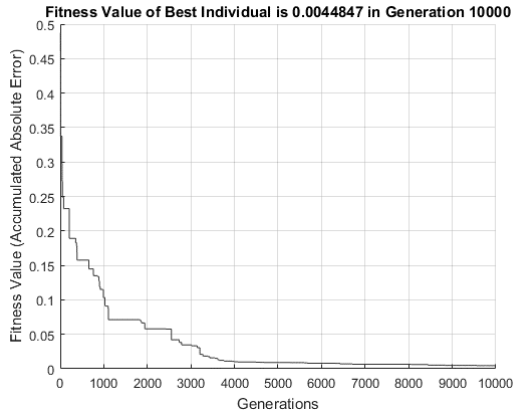


Fig. 10. GP fitness for sharp sensor linearization

The parameters used in GP algorithm for linearizing the thermocouple are shown in Table III, the gotten result is obtained after a adjusting this values.

TABLE III
INISIALIZATION VALUES FOR GP ALGORITHM IN THERMOCOUPLE CASE

| Paramether | value |
|--------------------|-------|
| Population size | 1001 |
| Operation cuantiti | 14 |
| Deepness | 4 |
| Generations | 10000 |
| Mutations | 35 |
| Seed | 1 |
| Tournament | 10 |
| Elitism | 1000 |

In equation 2 is represented the solution found for linearizing the thermocouple sensor, in this case the variable T is the dependent variable.

$$V + \frac{3.0 T - 1147.4}{T + \cos(T) - 0.0023148 T^2 - 533.0} - \frac{\left(T + \frac{9.0}{\cos(T)} + 0.05556 T^2 - 76.0\right)}{9.0 \sin(T) + 1201.5} \quad (2)$$

The Fig. 11 shows the behavior of the thermocouple linearization, where the obtained solution is almost linear so new response of the sensor is almost the expected line, with 0.04517668791 error as fitness, which is 0.6571% of relative error.

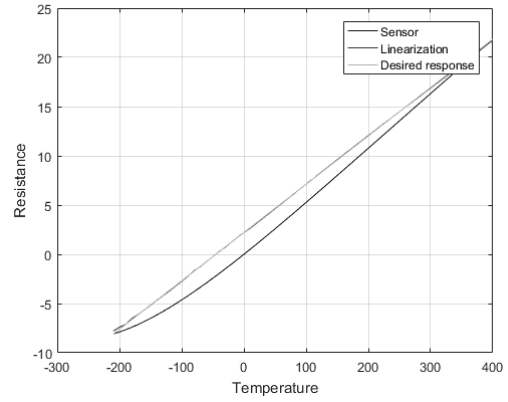


Fig. 11. GP solution behavior for thermocouple sensor linearization

Finally the Fig. 12 shows the evolution of the GP algorithm in the thermocouple sensor, as was expected in a GP algorithm the fitness is improved fast at the beginning and in the right side of the figure is exhibit how the population remains stable in a minimum local.

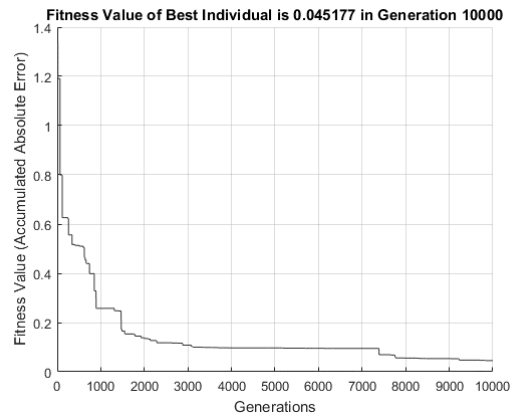


Fig. 12. GP fitness for thermocouple sensor linearization

VI. CONCLUSION AND FUTURE WORK

The paper presents the linearization of two different sensors, a distance sharp sensor, and a thermocouple type J, with a new proposed method using linear genetic programming. As was expected, the linearization in the sensors was reachable resulting with better approaches during a simulation test, in comparison with neural networks, because previous researches report a relative error of 0.5% [6], while the fuzzy logic mehtod have reached a 0.534% of relative error and the results. Nevertheless the proposed algorithm obtain a lower relative error of 0.3093%, however is important to validate as future work the linearization with real sensors and compare the results with their simulations in order to determine the efficiency of the algorithm. This test was run with equation-based individuals for the GP algorithm , a future work will be

the comparison of base tree individuals with the structure used in this paper.

REFERENCES

- [1] C. González and R. Zeleny, “Metrología ed,” 1988.
- [2] L. Corona, G. Abarca, and J. Mares, “Sensores y actuadores: Aplicaciones con arduino,” *Grupo Editorial Patria*, 2015.
- [3] R. P. Areny, *Sensores y acondicionadores de señal*. Marcombo, 2004.
- [4] A. J. López-Martín, M. Zuza, and A. Carloseña, “A cmos a/d converter with piecewise linear characteristic and its application to sensor linearization,” *Analog Integrated Circuits and Signal Processing*, vol. 36, no. 1-2, pp. 39–46, 2003.
- [5] H. Erdem, “Implementation of software-based sensor linearization algorithms on low-cost microcontrollers,” *ISA transactions*, vol. 49, no. 4, pp. 552–558, 2010.
- [6] G. Dempsey, N. Alt, B. Olson, and J. Alig, “Control sensor linearization using a microcontroller-based neural network,” in *IEEE International Conference on Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation.*, vol. 4. IEEE, 1997, pp. 3078–3083.
- [7] N. Medrano-Marques *et al.*, “A general method for sensor linearization based on neural networks,” in *International Symposium on Circuits and Systems, 2000. Proceedings. ISCAS 2000 Geneva.*, vol. 2. IEEE, 2000, pp. 497–500.
- [8] N. J. Medrano-Marques and B. Martín-del Brio, “Sensor linearization with neural networks,” *IEEE Transactions on Industrial Electronics*, vol. 48, no. 6, pp. 1288–1290, 2001.
- [9] M. O. Kayed, A. A. Balbola, and W. A. Moussa, “A smart high accuracy calibration algorithm for 3-d piezoresistive stress sensor,” *IEEE Sensors J.*, vol. 17, no. 5, pp. 1255–1263, 2017.
- [10] K. Rana, V. Kumar, and A. Batra, “Fpga resource requirements for thermistor linearization: A comparative study,” *Experimental Techniques*, vol. 41, no. 2, pp. 153–169, 2017.
- [11] K. P. S. Rana, V. Kumar, A. K. Dagar, A. Chandel, and A. Kataria, “Fpga implementation of steinhart-hart equation for accurate thermistor linearization,” *IEEE Sensors Journal*, 2018.
- [12] H.-N. L. Teodorescu, “Fuzzy logic system linearization for sensors,” in *International Symposium on Signals, Circuits and Systems (ISSCS), 2017*. IEEE, 2017, pp. 1–4.
- [13] P. R. Holloway, R. Subrahmayan, and G. S. Sheehan, “Linearized temperature sensor,” Feb. 6 2001, uS Patent 6,183,131.
- [14] M. V. Ivanov, “Bridge sensor linearization circuit and method,” Mar. 6 2001, uS Patent 6,198,296.
- [15] F. A. Castillo, K. A. Marshall, J. L. Green, and A. K. Kordon, “Symbolic regression in design of experiments: A case study with linearizing transformations,” in *Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation*. Morgan Kaufmann Publishers Inc., 2002, pp. 1043–1047.
- [16] M. Witczak, A. Obuchowicz, and J. Korbicz, “Genetic programming based approaches to identification and fault diagnosis of non-linear dynamic systems,” *International Journal of Control*, vol. 75, no. 13, pp. 1012–1031, 2002.
- [17] M. Montes-Rivera, M. O. Aguilar-Justo, and A. Ochoa-Zezzatti, “Equations for describing behavior tables in thermodynamics using genetic programming: Synthesizing the saturated water and steam table,” *Research in computer scienc*, 2016.
- [18] T. Weise, “Global optimization algorithms-theory and application,” *Self-published*, vol. 2, 2009.
- [19] M. M. Rivera, M. A. P. Ramos, J. L. O. Mora, and C. A. O. Zezzatti, “An alternative structure to replace the tree-based individuals in a genetic programming algorithm for decoupling mimo systems.”
- [20] *GP2Y0A21YK DATA-SHEET*, SHARP Corporation.
- [21] *Tablas de termocuplas y Pt100*, Arian Control & instrumentación, rev. b.

Propuesta de una arquitectura distribuida orientada a servicios para la adopción de la industria 4.0 en la manufactura de celdas solares

David Matias-Dominguez, Nohe R. Cazarez-Castro, Armando Martinez-Graciliano, Selene L. Cardenas-Maciell, Jorge Alonso-Perez.

Tecnológico Nacional de México - Instituto Tecnológico de Tijuana,
Calz. Tecnológico, S/N, 22414, Tijuana, Baja California, México.

david.matias17@tectijuana.edu.mx, nohe@ieee.org, aamg_a@tectijuana.edu.mx, lilettecardenas@ieee.org,
jalonso@tectijuana.edu.mx.
<http://tectijuana.edu.mx/>

Resumen—Para las industrias de manufactura la información es un recurso de gran importancia, esta información permite generar conocimiento para fines como la innovación, predicción de fallas o defectos; normalmente estos datos son transmitidos vía Internet. En este trabajo se propone una arquitectura distribuida orientada a servicios para la adopción de la industria 4.0 en la manufactura de celdas solares. Entre sus principales características están: la escalabilidad, integración de nuevos recursos, tolerancia a fallas, además permite integrar el procesamiento en tiempo real y procesamiento por lotes, según se requiera.

I. INTRODUCCIÓN

Así como la primera revolución industrial fue una revolución energética, ahora estamos al borde de la revolución industrial más grande de la historia, la cuarta revolución industrial [1], la cual está cambiando la manera de vivir, trabajar y relacionarnos con los demás. Dicha revolución industrial “proporciona la capacidad de conectar sistemas digitales, físicos y biológicos” [2]; según un informe presentado por Cisco en [3], para el 2020 estarán conectados alrededor de 50,000 millones de dispositivos, lo cual representa que cada vez más objetos se están conectando a la red, es decir, que el Internet de las Cosas (IoT) [4] se encuentra en pleno proceso de desarrollo, generando nuevos retos como la implementación de nuevas tecnologías de comunicación y protocolos de Internet, que garanticen la velocidad de transmisión de grandes volúmenes de datos de los dispositivos de IoT conectados, incluyendo las redes con bajo ancho de banda. Entre los protocolos [5] más conocidos que son utilizados en el IoT se encuentran: OPC UA, HTTP(REST/JSON), MQTT, CoAP, DDS, AMQP, XMPP, etc; cada uno con sus respectivas características. Por otra parte trabajar con dichos dispositivos de IoT, requiere la gestión y manipulación de grandes volúmenes, variedad, velocidad, veracidad y valor de los datos, sin embargo, las herramientas tradicionales de bases de datos no satisfacen las necesidades, dando origen al ecosistema del Big Data [6], como la ingesta de datos, almacenamiento, gestión de recursos, motores de procesamiento, etc.

En el presente trabajo se propone una arquitectura para la capa de cómputo de la industria 4.0 en la producción de celdas solares, en la misma son integrados algunos de los pilares de la cuarta revolución industrial [2], tales como: el Big Data y análisis de los datos, Internet de las Cosas, Computación en la nube.

Dicho trabajo se encuentra estructurado por cinco secciones. En las 3 primeras se hace una descripción detallada de la problemática y se propone una arquitectura distribuida orientada a servicios; mientras que en las secciones finales se ilustra la implementación de la arquitectura propuesta, integrando las tecnologías emergentes y se muestran los principales resultados y conclusiones.

II. DESCRIPCIÓN DEL PROBLEMA

En la manufactura de celdas solares, varios grupos de investigadores han explorado diversas técnicas de revestimiento tal como el método de bisturí (doctor *blade*) [7], serigrafía [8], impresión por inyección [9], [10] y deposición por rocío [11]. Actualmente se encuentra en la fase de implementación un prototipo de deposición por rocío y otro de *screen printing* [12], los cuales fueron desarrollados en las instalaciones del Tecnológico Nacional de México/Instituto Tecnológico de Tijuana (ITT), en el laboratorio de energías renovables. Estos prototipos han sido utilizados para la deposición de películas compactas. La técnica de *screen printing* se basa en la técnica de serigrafía, sin embargo, ninguno de estos dos prototipos cuenta con un módulo de recolección de datos, con base a esto, las celdas solares contarán con propiedades de producción cualitativas y cuantitativas diferentes entre unas y otras, como la velocidad de barrido, el tiempo de producción, la fuerza aplicada, etc. Además, considerando que cada técnico de celdas solares puede variar por completo los parámetros de producción.

Ante la problemática de producir celdas solares con distintas propiedades y el hecho de no contar con un registro de todas las variables y parámetros generados durante la fabricación

de la celda solar, y considerando el error humano durante la fabricación, es propuesta una arquitectura distribuida orientada a servicios como soporte a los trabajos previos realizados en [12] durante la automatización en la deposición de películas delgadas de nanomateriales para la posible utilización en celdas solares.

Este trabajo se centra en la capa de cómputo, aplicando los métodos de mensajería más eficientes [13], [14] para la obtención y recolección de datos, generados por los mecanismos de deposición por rocío y de *screen printing* durante la operación. Estos datos serán procesados tomando en consideración que las celdas solares pueden ser fabricadas adoptando todas las técnicas de impresión en sustratos rígidos y flexibles, lo cual permite un fácil ajuste de las características de las celdas solares a la aplicación deseada, por ejemplo, la transparencia, el color, la forma, el diseño y otras características pueden variarse fácilmente cambiando los parámetros de impresión y las fórmulas de pasta o tinta utilizadas en [15].

III. PROPUESTA DE ARQUITECTURA DISTRIBUIDA ORIENTADA A SERVICIOS

La arquitectura propuesta se compone en 5 capas, las cuales se muestran en la Fig. 1, entre sus principales características se encuentran la compartición de recursos, concurrencia, apertura para añadir nuevos recursos, escalabilidad y la tolerancia a fallas o prevención de defectos.

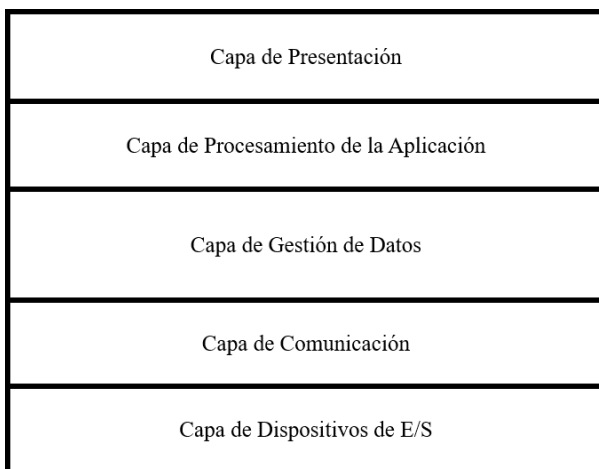


Fig. 1. Estructura en capas de la arquitectura propuesta.

A continuación, son presentadas las funciones de cada capa presentada en la Fig. 1.

A. La Capa de Dispositivos de E/S

Representa el sistema de obtención de datos, esta capa enlaza directamente a los sensores y actuadores instalados en los mecanismos que nos estarán proveyendo información; en nuestro caso a partir de los mecanismos definidos en [9], [10], estarán compuestos en su mayoría por sensores y actuadores que realizarán distintas tareas. Sobre los instrumentos de medición serán recopilados los datos en unidades de ingeniería correspondiente a la magnitud medida, sin importar el volumen

y la variedad de los datos que estos generen. Estos dispositivos de E/S estarán conectados a la capa de comunicación, por medio de algunos de los protocolos de comunicación [4].

Los sensores y partes mecánicas del prototipo de *screen printing* diseñado en las instalaciones del ITT se muestran en la Fig. 2, el prototipo corresponde a una de las etapas del proceso de manufactura de celdas solares. Este mecanismo consta de 3 motores de corriente directa, 2 sensores de posición digitales, un sensor de distancia ultrasónico y una cámara de vídeo.

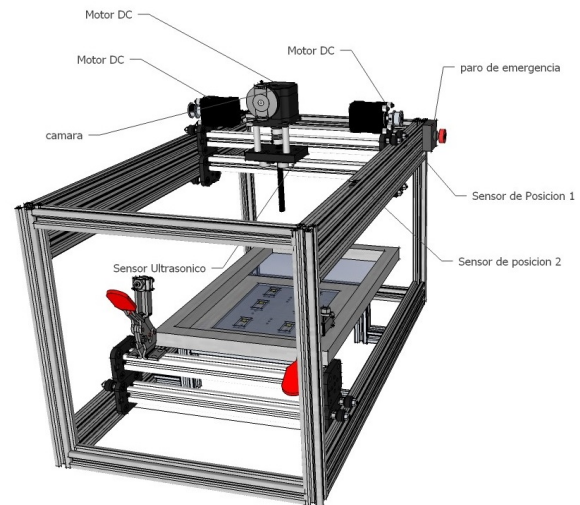


Fig. 2. Mecanismo de *screen printing*.

B. La Capa de Comunicación

Se centra en los medios o plataformas para transportar grandes volúmenes de datos, así como una gran variedad de éstos, manteniendo una velocidad de transferencia previamente definida, y utilizando los principales formatos de intercambio de datos [6].

C. La Capa de Gestión de Datos

Almacena los datos recolectados en centro de datos, y se les asigna un periodo de vida. Sobre esta capa se apoyará la capa de procesamiento de la aplicación.

D. La Capa de Procesamiento de la Aplicación

Involucra máquinas de aprendizaje, procesamientos en tiempo real, y procesamiento por lotes, como parte de una arquitectura tipo Lambda [16], para predecir fallas, tiempos de vida de los componentes en los mecanismos, así como ajustes en las características de producción de las celdas solares.

E. La Capa de Presentación

Se visualizan los datos y la creación de proyectos por medio de una interfaz gráfica, ya sean aplicaciones móviles, de escritorio, o web.

La estructura en forma de árbol para la creación de un proyecto se muestra en la Fig. 3, la cual involucra la capa

de presentación, la capa de gestión de datos y la capa de procesamiento de la aplicación, la misma estructura de árbol se componen de la siguiente manera:

Definir los datos de la organización, posteriormente es definido el nombre del proyecto; se considera que una organización puede tener múltiples proyectos. Dentro de esta capa son descritas las propiedades del proyecto, posteriormente son especificadas las fases o etapas que lo componen. Finalmente, por cada fase especificada son seleccionados los dispositivos necesarios, así como especificar si las señales son de entrada o salida y si son analógicas o digitales: números, valores booleanos, cadena de caracteres, imágenes, audio o vídeo, si en dado caso que un dispositivo no se encuentre disponible dentro del banco de datos, se le brinda al usuario la opción de someter un nuevo elemento, y especificar cada una de sus características.

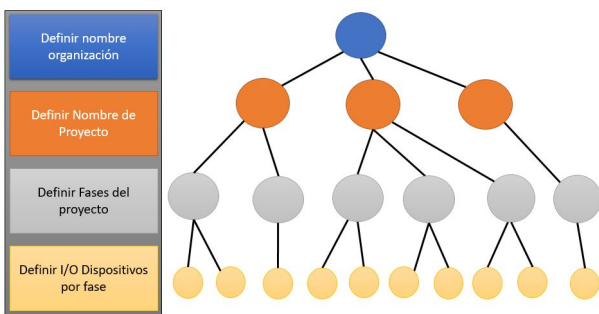


Fig. 3. Esquema general de despliegue en forma de árbol.

El flujo de datos de la arquitectura propuesta se visualiza en la Fig. 4 integrando cada una de las capas descritas anteriormente. La idea principal de dicha arquitectura es que el flujo de la información entrante desde la capa de comunicación sea replicada en la subcapa de procesamiento por lotes y la subcapa de procesamiento en tiempo real, de este manera se facilita la visualización de la información en tiempo real hacia la capa de presentación, además se puede realizar consultas desde la capa de gestión de datos para resolver algunos tipos de problemas según sea el caso de uso. Por su parte, desde la capa de comunicación son construidas los canales de transmisión para el intercambio de datos entre dispositivos, procesos, servidores, sistemas o aplicaciones. La arquitectura garantiza la ejecución de grandes cargas de trabajo y permite implementaciones de diversos casos de uso manteniendo siempre presente sus principales características como: la escalabilidad, tolerancia a fallos y robustez.

IV. IMPLEMENTACIÓN DE LA ARQUITECTURA DISTRIBUIDA ORIENTADA A SERVICIOS SOBRE ALGUNAS FASES DE LA MANUFACTURA DE CELDAS SOLARES

En este apartado son integradas las tecnologías emergentes afines al presente proyecto, con el fin de garantizar la escalabilidad, velocidad, y tolerancia a fallos. La integración de estas herramientas en cada capa de la arquitectura propuesta se muestra en la Fig. 5.

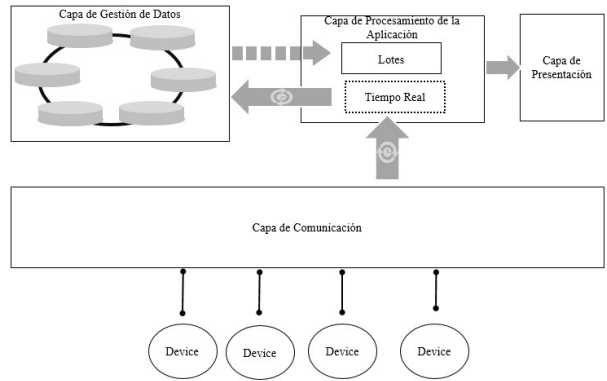


Fig. 4. Propuesta de arquitectura orientada a servicios.

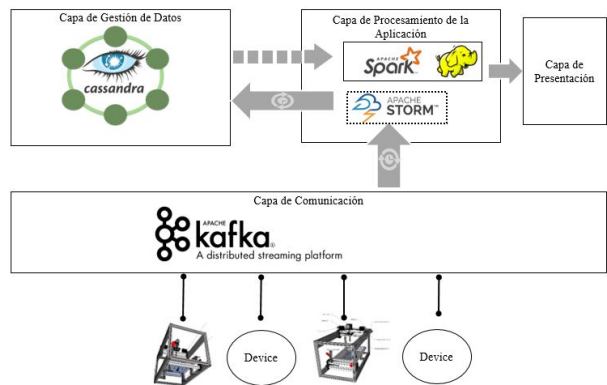


Fig. 5. Implementación de la arquitectura distribuida orientada a servicios sobre la manufactura de celdas solares.

Para transportar mensajes en el modelo publicador/suscriptor entre la capa de comunicación y la capa de dispositivos de E/S durante las diferentes fases del proceso de manufactura (véase Fig. 6), se establece el uso del protocolo de conectividad MQTT [17], este protocolo fue seleccionado debido a que consume pocos recursos en dispositivos, y es útil para conexiones en redes con alta latencia. Por su parte Apache Kafka [18] es una plataforma de transmisión distribuida diseñada para transferir en tiempo real grandes volúmenes de datos como colas de mensajerías y permite evitar cuellos de botella, asimismo cuenta con múltiples conectores para la integración de almacenamiento y procesamiento de datos dentro de la transmisión.

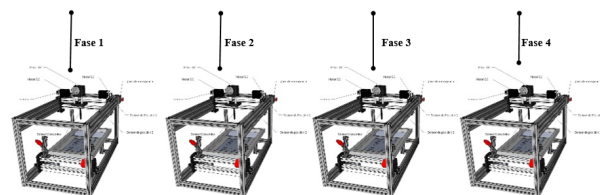


Fig. 6. Capa de Dispositivos de E/S, de las fase de producción de celdas solares.

Con base en los resultados obtenidos en las pruebas repor-

tadas en [13], en la capa de procesamiento de la aplicación se implementan las tecnologías Apache Spark para la manipulación de datos por lotes y Apache Storm para la manipulación de datos en tiempo real.

En la capa de gestión de datos, fue seleccionado el sistema gestor de base de datos Apache Cassandra debido a su rendimiento sobre otros sistemas gestores NoSQL. Esta decisión se hizo tomando como referencia el análisis hecho a seis tiendas de datos en [14] por la universidad de Toronto, donde Apache Cassandra muestra mejores resultados en cuanto a los criterios: rendimiento por carga de trabajo, el proceso de carga, analítica por carga de trabajo y operativa mixta.

Finalmente, la estructura para las primeras dos fases del proyecto de producción de celdas solares es mostrada en la Fig. 7.

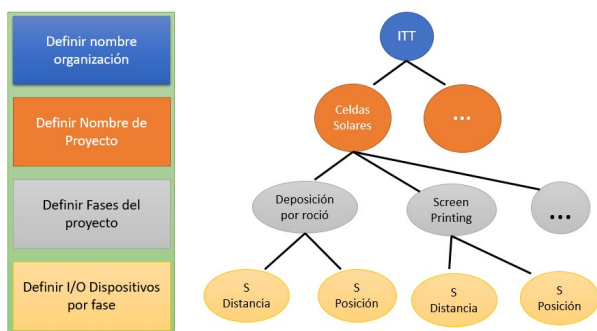


Fig. 7. Despliegue en forma de árbol para la producción de celdas solares.

V. CONCLUSIÓN

La arquitectura propuesta en el presente trabajo atenderá las necesidades de la capa de cómputo para la manufactura de celdas solares, se prevee que podrá ser utilizada en otros procesos de manufactura debido a la robustez de las tecnologías de software propuestas en aspectos tales como el soporte a una amplia gama de datos estructurados, semiestructurados y no estructurados.

Considerando las ventajas de la obtención y recopilación de gran volumen y variedad de información que proporciona la arquitectura, puede construirse una base de conocimiento que permita adaptar los procesos productivos para atender con mayor rapidez y flexibilidad a los requerimientos que exige la industria actual.

AGRADECIMIENTOS

Este proyecto fue parcialmente financiado por el Tecnológico Nacional de México con el proyecto 6351.17-P.

REFERENCIAS

- [1] Lasi, H., Fettke, P., Kemper, H. G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. *Business & Information Systems Engineering*, 6(4), 239-242.
- [2] Schwab, K. (2016). La cuarta revolución industrial. Debate.
- [3] Evans, D. (2011). Internet de las cosas. Cómo la próxima evolución de Internet lo cambia todo. Cisco Internet Business Solutions Group-IBSG, 11(1), 4-11.
- [4] Xia, F., Yang, L. T., Wang, L., & Vinel, A. (2012). Internet of things. *International Journal of Communication Systems*, 25(9), 1101.
- [5] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4), 2347-2376.
- [6] Aguilar, L. J. (2016). Big Data, Análisis de grandes volúmenes de datos en organizaciones. Alfaomega Grupo Editor.
- [7] Schilinsky, P., Waldauf, C., & Brabec, C. J. (2006). Performance analysis of printed bulk heterojunction solar cells. *Advanced Functional Materials*, 16(13), 1669-1672.
- [8] Krebs, F. C., Jørgensen, M., Norrman, K., Hagemann, O., Alstrup, J., Nielsen, T. D., ... & Kristensen, J. (2009). A complete process for production of flexible large area polymer solar cells entirely using screen printing-first public demonstration. *Solar Energy Materials and Solar Cells*, 93(4), 422-441.
- [9] Hoth, C. N., Choulis, S. A., Schilinsky, P., & Brabec, C. J. (2007). High photovoltaic performance of inkjet printed polymer: fullerene blends. *Advanced Materials*, 19(22), 3973-3978.
- [10] Aernouts, T., Aleksandrov, T., Giroto, C., Genoe, J., & Poortmans, J. (2008). Polymer based organic solar cells using ink-jet printed active layers. *Applied Physics Letters*, 92(3), 22.
- [11] Vak, D., Kim, S. S., Jo, J., Oh, S. H., Na, S. I., Kim, J., & Kim, D. Y. (2007). Fabrication of organic bulk heterojunction solar cells by a spray deposition method for low-cost power generation. *Applied Physics Letters*, 91(8), 081102.
- [12] ALONSO-PEREZ, J., REYNOSO-SOTO, E., TRUJILLO-NAVARRETE, B., & Nohe, C. C. (2017). Fabricación automatizada de películas delgadas compactas en el desarrollo de celdas solares tipo Grätzel. *Revista de Tecnología*, 4(11), 36-42.
- [13] Chintapalli, S., Dagit, D., Evans, B., Farivar, R., Graves, T., Holderbaugh, M., ... & Poulosky, P. (2016, May). Benchmarking streaming computation engines: storm, flink and spark streaming. In *Parallel and Distributed Processing Symposium Workshops*, 2016 IEEE International (pp. 1789-1792). IEEE.
- [14] Rabl, T., Gmez-Villamor, S., Sadoghi, M., Munts-Mulero, V., Jacobsen, H. A., & Mankovskii, S. (2012). Solving big data challenges for enterprise application performance management. *Proceedings of the VLDB Endowment*, 5(12), 1724-1735.
- [15] Mariani, P., Vesce, L., & Di Carlo, A. (2015). The role of printing techniques for large-area dye sensitized solar cells. *Semiconductor Science and Technology*, 30(10), 104003.
- [16] Lin, J. (2017). The Lambda and the Kappa. *IEEE Internet Computing*, 21(5), 60-66.
- [17] Hwang, H. C., Park, J., & Shon, J. G. (2016). Design and implementation of a reliable message transmission system based on MQTT protocol in IoT. *Wireless Personal Communications*, 91(4), 1765-1777.
- [18] Kumar, M., & Singh, C. (2017). Building Data Streaming Applications with Apache Kafka: Design, develop and streamline applications using Apache Kafka, Storm, Heron and Spark.

Sistemas de Gestión de Seguridad de la Información, un reto para las universidades de México

Alejandra Guadalupe Silva Trujillo*^{||}, Josué Sánchez Olvera*[†], José Eduardo Corpus Sánchez*[‡],
Pablo Andrés Castañeda Estrada*[§], Jesús Gerardo Heredia Guerrero*[¶]

*Universidad Autónoma de San Luis Potosí, Facultad de Ingeniería, Dr. Manuel Nava No. 8,
Col. Zona Universitaria Poniente, C.P. 78290, San Luis Potosí, S. L. P., México

^{||}Email: asilva@uaslp.mx

[†]Email: josue.sanchezo@outlook.com

[‡]Email: eddcsanchez@gmail.com

[§]Email: andres.cas.estr@gmail.com

[¶]Email: garhard28@hotmail.com

Abstract—La seguridad informática en México es un tema del que poco se habla en nuestro país, en un estudio reciente encontramos que 4 de cada 10 instituciones de educación superior no cuentan con políticas de seguridad informática. Aunado a que casi el 80% no indican los mecanismos de tratamiento de datos personales en su institución. Como en toda organización, en las universidades la información es una parte muy valiosa. Las ventajas de tener un Sistema de gestión de seguridad de la información son claras, ya que al implementarlo, además de garantizar los tres aspectos fundamentales de seguridad en un sistema: integridad, confidencialidad y disponibilidad de los datos, permite aumentar la confianza entre la comunidad universitaria y la mejora de la imagen institucional.

I. INTRODUCCIÓN

La certificación ISO 27001:2013 es un estándar para los requisitos de seguridad de la información, y ha sido aprobada por la International Organization for Standardization (ISO) y por la International Electrotechnical Commission (IEC) [1]. Desarrollar un Sistema de Gestión de Seguridad de la Información en esta norma muestra a la sociedad que las instituciones y universidades, identifican, analizan y tratan todos los aspectos y riesgos asociados a la seguridad de la información que se manejan.

En México, son pocas las instituciones educativas que cuentan con un Sistema de Gestión de Seguridad de la Información (SGSI), que se podría traducir como un reflejo del compromiso de la organización en el cumplimiento de los estándares de seguridad de la información relativa a las formaciones y títulos que imparte. La importancia de establecer un marco regulatorio es vital ya que se gestionan grandes volúmenes de información, gran parte de ella sensible. Con la creciente adopción en el uso de las tecnologías de la información en nuestras actividades cotidianas, es necesario evaluar los riesgos y aplicar salvaguardas y políticas que nos ayuden en

la protección de activos, que va desde la información hasta los recursos técnicos y humanos.

Los beneficios de establecer un SGSI se pueden traducir en los siguientes aspectos: i) Establecimiento de la mejora continua en la gestión de la seguridad; ii) reducción de costos relacionados a incidentes; iii) aumento del nivel de confianza de la población universitaria; iv) mejora de la imagen institucional; v) mejores índices de continuidad y disponibilidad de servicios.

En un estudio realizado por el Instituto Nacional de Estadística y Geografía (INEGI) se muestra que la penetración de Internet en los hogares ha aumentado considerablemente, al cuadruplicar en diez años el número de hogares con acceso a la red, de 10.1% en 2006 a 47% en 2016 [2]. En el ámbito de la industria el fenómeno sigue la misma tendencia. De acuerdo con las cifras del Foro Económico Mundial, México ocupa actualmente el lugar número 71 en lo que se refiere al desarrollo tecnológico [3]. Aunque el país ha crecido considerablemente, el resto del mundo crece más rápido. La mayor área de oportunidad se encuentra en el sector público. Si bien existe una Estrategia Digital Nacional por parte de las autoridades del gobierno mexicano, cuyos mayores retos se traducen en: la inclusión en el acceso a Internet; la implementación de un gobierno digital; el combatir con eficacia los ciberdelitos y avanzar en el uso de la inteligencia artificial, se han convertido en esfuerzos relevantes pero insuficientes [4]. Cuando se planteó la Estrategia Digital Nacional en 2013, una de sus metas era el escalar peldaños en el índice de digitalización de la Organización para la Cooperación y el Desarrollo Económicos (OCDE). Sin embargo, el último reporte del informe Perspectivas de la Economía Digital 2017, una evaluación bianual de la OCDE, México está en el penúltimo sitio en cuanto a la generación de valor agregado en el sector de las Tecnologías de Información y Comunicación (TICs)

[5]. Sin duda existe un número considerable de deficiencias las cuales menguan los esfuerzos que incentiven el desarrollo de la red tecnológica en México, que impide al país explotar plenamente el potencial de las TICs.

En materia de Seguridad de la Información, los esfuerzos realizados en el entorno legal surgen con el establecimiento de la Ley Federal de Protección de Datos en Posesión de los Particulares (LFPDPPP) [6], siendo ésta de carácter obligatoria para todas aquellas personas físicas y morales que traten datos personales, debiendo entonces entre otras cosas, implementar medidas de seguridad para proteger los datos personales de los Titulares que se recaben. Y recientemente, en 2017 fue aprobada la Ley General de Protección de Datos Personales en Posesión de Sujetos Obligados (LGPDPSSO)[7], que establece las bases, principios y procedimientos para garantizar el derecho a la protección de sus datos personales, señalando como sujetos obligados a cualquier autoridad en el ámbito federal, estatal y municipal, entidad, órgano y organismo de los poderes ejecutivo, legislativo y judicial, órganos autónomos, partidos políticos, fideicomisos y fondos públicos. Este requerimiento normativo por parte de las autoridades mexicanas, plantea una estrategia que sirva para dar cumplimiento a la Ley y que, a su vez, sea parte de la implementación de un sistema de gestión que garantice la seguridad de la información.

El objetivo de este trabajo es dar un breve panorama del entorno en las universidades mexicanas respecto a la adopción de buenas prácticas de seguridad de la información. Por otro lado, describimos los resultados obtenidos al llevar a cabo éste ejercicio en la Facultad de Ingeniería de la Universidad Autónoma de San Luis Potosí (UASLP), esto como parte de un modelo que se aplicaría de la misma universidad. El SGSI, como lo recomienda la norma ISO 27001:2013, permitiría a las instituciones educativas, emplear prácticas adecuadas de seguridad de la información, concientizando a sus usuarios sobre los riesgos y amenazas actuales a través de prácticas, procedimientos, guías y lineamientos documentados y liderados por la alta gerencia. Lo anterior conlleva naturalmente, a un impacto positivo en la reducción de costos y optimización de procesos.

II. ANTECEDENTES

A fin de salvaguardar la información en las instituciones y garantizar la confidencialidad, disponibilidad e integridad se encuentran disponibles diferentes metodologías. La norma ISO 27001:2013 marca las directrices requeridas para el diseño de un modelo de SGSI. La metodología está diseñada bajo un esquema de mejora continua Planear, Hacer, Verificar y Actuar (Plan, Do, Check y Act (PDCA)). De este modo para abarcar desde el aseguramiento de la información al cumplimiento del marco legal, se utilizan controles administrativos y tecnológicos. La determinación de los controles a implementar se deriva de un análisis de riesgos.

En México, cuatro de cada diez Instituciones de Educación Superior (IES) no tienen políticas de seguridad informática de acuerdo con el Estudio ejecutivo 2016 de la Asociación Nacional de Universidades e Instituciones de Educación Superior

(ANUIES) [8]. Para el estudio fueron aplicadas encuestas a casi 150 IES del país. Otro aspecto importante para considerar es que 5 de cada 10 IES no llevan a cabo auditorías de seguridad informática. Lo que parece indicar que se tiene la creencia de que los incidentes de seguridad de la información solo se dan en áreas del sector comercial, financiero, productivo, y que el sector educativo está exento. Sin embargo, no debemos olvidar que en la actualidad la protección de la información se ha convertido en un aspecto de primera necesidad en todos los aspectos de la vida cotidiana, incluido el ámbito académico.

De acuerdo a la última evaluación de US News Education en 2017 [9], se posicionó a la UASLP en el tercer lugar dentro de las mejores universidades a nivel nacional. Ante este reto se obtiene la urgente necesidad y obligación de alinearse a la legislación vigente y adoptar las mejores prácticas para salvaguardar la información transferida por el personal administrativo, académico y universitario que la integra. Para este fin es indispensable un conjunto de políticas de administración y control de información, que conformen un SGSI.

A. SGSI en Instituciones de Educación Superior

Tener un acceso directo a cada una de las IES tanto locales como nacionales es una tarea laboriosa. Por lo que para obtener una cifra aproximada que nos ayude a reflejar el número de certificaciones en el entorno educativo que hay en México fue realizada a través de las estadísticas del mismo ISO. En un estudio realizado por dicho Instituto en 2016, muestra el número de certificaciones a nivel mundial [10]. Estos datos nos ayudan a calcular el número aproximado de instituciones educativas que están certificadas a nivel mundial y en particular en México. En la Tabla I se puede observar que a nivel mundial solamente existían 109 certificaciones ISO 27001 en entidades educativas, dato que de manera inmediata nos revela un cálculo aproximado de las certificaciones nacionales. Al llevar a cabo un cálculo probabilístico sobre los datos reales, se obtiene como resultado la probabilidad de las instituciones educativas que tienen la certificación ISO 27001 respecto al total de su nivel.

Por ejemplo, en el continente americano en el 2016 había 2,033 certificaciones ISO 27001, por lo tanto, se puede calcular que aproximadamente habría cerca de 6 entidades educativas que tienen esta certificación. Estos cálculos se realizan asumiendo que se mantiene una proporción igual con respecto a las certificaciones a nivel mundial, y así se obtienen las aproximaciones para Norteamérica y México, tal y como se muestra en la Tabla I. De esta manera podemos asumir que en el país solamente hay una institución educativa certificada en el ISO 27001.

Estos datos parecen más reales si tomamos en cuenta casi toda la literatura se enfoca solamente en el entorno empresarial y son escasos los trabajos publicados en el ámbito educativo [11][12][13].

B. Entorno legal

Uno de los aspectos más importantes dentro de un SGSI es el saber con qué normas de Seguridad de Datos se deben

TABLA I
CERTIFICACIONES ISO 27001

| Nivel | Certificaciones ISO 27001 | Porcentaje | Certificaciones ISO 27001 en el sector educativo |
|----------------------|---------------------------|------------|--|
| Mundial | 37804 | 100% | 109 |
| Continente Americano | 2033 | 5.37% | 5.85 (Aproximado) |
| América del Norte | 1469 | 3.88% | 4.23 (Aproximado) |
| México | 237 | 0.6% | 0.6 (Aproximado) |

cumplir. En México, como ya se mencionó anteriormente, existen dos principales leyes: LFPDPPP y la LGPDPPSO.

Se realizó una investigación exhaustiva en las páginas web de 481 universidades ubicadas en los 32 estados de la República Mexicana para verificar los avisos de privacidad de la información que manejan. De éste análisis se obtuvieron los siguientes resultados:

- 382 IES no mencionaban sus leyes de protección de datos
- 49 IES se rigen por la LFPDPPP.
- 50 IES se rigen por la LGPDPPSO.

En la Figura 1 podemos observar que casi el 80% de las Universidades no cuentan con avisos de privacidad de datos, y el resto está dividido entre las leyes mencionadas con anterioridad.

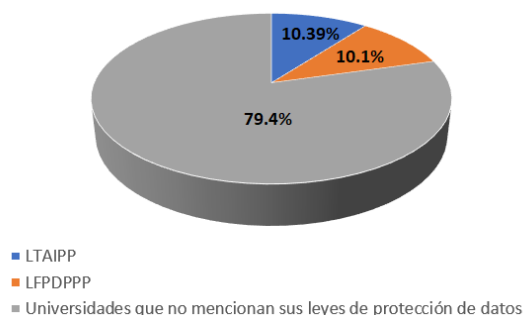


Fig. 1. Leyes usadas por universidades

III. IMPLEMENTACIÓN DE UN MODELO DE SGSI EN LA UASLP

Para el desarrollo del modelo basado en la norma ISO 27001, se identificaron los procesos de entradas y las salidas de información que posee la Facultad de Ingeniería de la UASLP (FI-UASLP) tal como se muestra en la Figura 2. La gran dimensión de la FI-UASLP radica en su oferta académica conformada por 15 programas de estudio de licenciatura, 8 de maestría y 3 de doctorado. Los cuales cada uno presenta procesos independientes, que para su comprensión se realizó un análisis del ciclo PDCA que se especifica en la Figura 3 .

Desde el inicio de la investigación se visualiza a la FI-UASLP como un solo proceso que engloba todos los procesos académicos y administrativos que la conforman, esto con la finalidad de trabajar con mayor entendimiento y facilidad al implementarán los fundamentos de la norma ISO 27001:2013.



Fig. 2. Procesos FI-UASLP

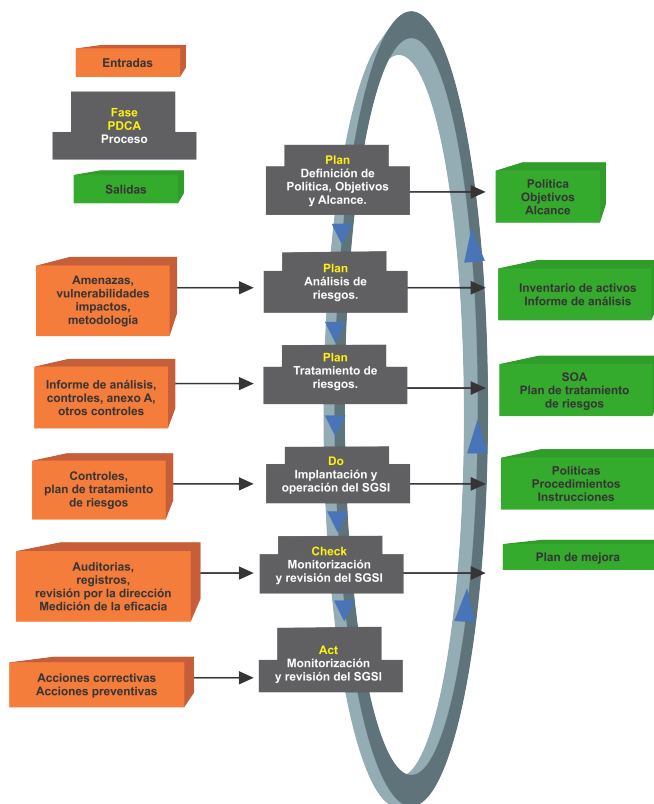


Fig. 3. Fases PDCA (iso27000.es)

Dentro de la Norma ISO 27001, se presentan los siguientes documentos:

- Análisis de riesgos
- Políticas de Seguridad
- Plan de continuidad de negocio
- Declaración de Aplicabilidad (SOA)

A. Análisis de riesgos

Durante el análisis de riesgo se obtuvo un listado de los posibles riesgos que presenta la FI-UASLP, sobre el control de accesos físicos y lógicos. Este listado mostró que los accesos físicos a la FI-UASLP son vulnerables debido a que se comparten entradas con otras facultades dentro de la zona universitaria, lo que significa un riesgo de prioridad alta, y deja al descubierto que los procesos administrativos (registro de la entrada de personas externas a la FI-UASLP), no cumplen su función de asegurar la protección de los activos informáticos. Respecto a los accesos lógicos, no se presenta una situación distinta, ya que la conexión de redes VLAN y acceso a internet de las instalaciones, no cumple las medidas

mínimas (8 caracteres que contenga una letra mayúscula, un número y un carácter especial), aunado a que no existe un tiempo establecido para que se realice el cambio de dichas contraseñas.

Una vez identificados todos los posibles riesgos, se realizó la evaluación de cada uno de ellos, asignando probabilidad e impacto de una escala del 1 al 5, siendo 1 baja probabilidad y 5 alta probabilidad, como se ilustra en la Tabla II. En comparativa respecto al impacto, 1 correspondería a bajo impacto, y 5 alto impacto, como se indica en la Tabla III.

TABLA II
EVALUACIÓN DE PROBABILIDAD

| Probabilidad | Descripción | Valor |
|--------------|---|-------|
| Alta | Es muy factible que el hecho se presente | 5 |
| Media | Es factible que el hecho se presente | 3 |
| Baja | Es poco factible que el hecho se presente | 1 |

TABLA III
EVALUACIÓN DE IMPACTO

| Impacto | Descripción | Valor |
|---------|--|-------|
| Alto | Si el hecho llega a presentarse tendría alto impacto o efecto dentro de la institución | 5 |
| Medio | Si el hecho llega a presentarse tendría un impacto o efecto medio dentro de la institución | 3 |
| Bajo | Si el hecho llega a presentarse tendría bajo impacto o efecto dentro de la institución | 1 |

Con este ejercicio se pudo determinar los riesgos más críticos para llevar a cabo el Plan de Respuesta a Riesgos, clasificando cada uno de ellos dentro de las siguientes acciones: i) evitar; ii) transferir; iii) mitigar; iv) asumir.

A partir de este análisis de riesgos se desarrollaron las políticas de seguridad adecuadas para garantizar disponibilidad, integridad, y confidencialidad de la información.

B. Políticas de seguridad

Dentro de las políticas de seguridad que comprenden el aspecto lógico y físico en la FI-UASLP se identificaron las siguientes:

- Políticas de acceso a áreas de cómputo.
- Políticas de protección de los equipos de computación.
- Políticas de administración de archivos (aspecto lógico).
- Políticas de uso de medios de almacenamiento.
- Políticas de instalación de software.
- Políticas de administración de la configuración de redes.
- Políticas de cumplimiento y sanciones.
- Políticas de uso de recursos informáticos.
- Políticas de contraseña.
- Políticas de uso de la información.
- Políticas del uso de Internet y correo electrónico.
- Políticas de uso de la intranet y portal web universitario.
- Políticas para desarrolladores de software.
- Políticas para administradores de sistemas.
- Políticas de copias de respaldo.
- Políticas de uso de cortafuegos.
- Políticas de acceso físico.



Fig. 4. Tratamientos de riesgos

Las políticas planteadas abarcan las responsabilidades de todas las personas físicas y morales las cuales tienen relación con la FI-UASLP, ya sean estudiantes, personal administrativo, académicos y/o proveedores.

Estas políticas, regulan el aspecto de control de los laboratorios de la FI-UASLP, así como el manejo y control de los programas y dispositivos de almacenamiento de información que pueden estar relacionados a las actividades de cada uno de los laboratorios.

Además de políticas de control de las contraseñas de cada uno de los roles que presenta las redes de la UASLP (alumnos, profesores e invitados), y respaldos de información ya sea mediante los correos electrónicos o almacenada en servidores.

Aunado a las políticas de soporte para los equipos computacionales que han sido adquiridos por la FI-UASLP.

C. Plan de continuidad de negocio

El Plan de continuidad marca los procesos que deben de ser seguidos en caso de que alguno de los riesgos se presente ya sea desde la realización de verificación de identidad de los visitantes hasta el procedimiento de reacción ante un desastre natural. Esto se realiza mediante la metodología de tratamiento de riesgos que detalla la figura 4.

D. Declaración de Aplicabilidad (SOA)

A continuación se enumeran los controles propuestos para el SGSI de la FI-UASLP a partir del análisis de riesgos:

- Declaración de responsabilidades para usuarios (internos y externos).

- Política de uso de dispositivos de almacenamiento de información dentro de áreas restringidas.
- Plan de Concienciación, educación y capacitación de alumnos, profesores y administrativos de la FI-UASLP.
- Declaración y clasificación de activos informáticos.
- Política de control de accesos.
- Control de acceso a las redes.
- Declaración de restricciones de acceso.
- Gestión de contraseñas de usuarios.
- Controles físicos de entrada y seguridad en oficinas.
- Instalaciones de suministro eléctrico.
- Mantenimiento de los equipos.
- Tratamiento contra el código malicioso.
- Control de copias de seguridad de la información.
- Política sobre instalación del software.
- Controles de red.
- Acuerdos de confidencialidad .
- Políticas de intercambio de información.
- Gestión de cambios en los servicios prestados por terceros.
- Notificación y respuesta de los eventos.
- Protección de datos y privacidad de la información personal.

IV. RESULTADOS

Sobre los hallazgos obtenidos se puede asumir que el escenario actual de la FI-UASLP en relación a la seguridad informática, no difiere a las demás IES.

En general, los riesgos de alto impacto y poca probabilidad de ocurrir son atribuidos a debilidades en las estrategias de seguridad. Es importante que la concientización de este problema empiece desde la alta gerencia para que ésta se extienda por toda la institución.

En esta situación, se plantea crear un SGSI que permita cubrir las necesidades básicas de los datos de alumnos, profesores, administrativos y entidades externas relacionadas.

V. CONCLUSIONES

Durante el desarrollo de la investigación pudimos notar que uno de los principales problemas era la falta de concienciación en la FI-UASLP sobre la importancia y necesidad de la protección de datos personales. Otro aspecto relevante que pudimos observar al realizar el análisis de riesgos fue el área de oportunidad que hay en las medidas de accesos físicos a las instalaciones de la FI-UASLP para el control de entradas y salidas de personas ajenas a la universidad.

Cabe rescatar que este proyecto abre la posibilidad de poder brindar una guía y orientación para el desarrollo de políticas y controles que puedan regular y asegurar la información que posee la FI-UASLP en apego a la LFPDPPP.

En el trabajo futuro se plantea que se pueda realizar una proyección de efectividad de los controles a un mediano plazo como la evaluación de las políticas planteadas en el manejo y trata de los datos personales de los cuerpos administrativo, docente, estudiantil, proveedores y entidades externas.

AGRADECIMIENTOS

Este proyecto ha sido financiado por el Programa para el Desarrollo Profesional Docente (PRODEP) con clave 511-6/17-7930.

REFERENCES

- [1] So/iec 27000 family - information security management systems. [Online]. Available: <https://www.iso.org/isoiec-27001-information-security.html>
- [2] INEGI. (2016) Encuesta nacional sobre disponibilidad y uso de tecnologías de la información en los hogares 2016. [Online]. Available: <http://www.beta.inegi.org.mx/proyectos/enchogares/regulares/dutih/2016/>
- [3] S. de Economía. (2018) Reporte de competitividad global 2017-2018. [Online]. Available: https://www.gob.mx/cms/uploads/attachment/file/261265/An_lisis_ejecutivo_ICG_2017-2018.pdf
- [4] G. Federal. (2013, nov) Estrategia digital nacional. [Online]. Available: https://framework-gb.cdn.gob.mx/data/institutos/edn/Estrategia_Digital_Nacional.pdf
- [5] OECD, "Oecd digital economy outlook 2017." [Online]. Available: </content/book/9789264276284-en>
- [6] H. C. de la Unión. (2010, jul) Ley federal de protección de datos personales en posesión de los particulares. [Online]. Available: <http://www.diputados.gob.mx/LeyesBiblio/pdf/LFPDPPP.pdf>
- [7] ——. (2017, jan) Ley general de protección de datos personales en posesión de sujetos obligados. [Online]. Available: http://www.diputados.gob.mx/LeyesBiblio/ref/lgpdpso/LGPDPSO_orig_26ene17.pdf
- [8] J. L. P. López, "Estado actual de las tecnologías de la información y las comunicaciones en las instituciones de educación superior en México," 2016.
- [9] U. N. . W. Report. (2017) Best global universities in Mexico. [Online]. Available: <https://www.usnews.com/education/best-global-universities/mexico>
- [10] I. O. of Standardization, "The iso survey of management system standard certifications 2016." [Online]. Available: <https://isotc.iso.org/livelink/livelink?func=ll&objId=18808772-&objAction=browse&viewType=1>
- [11] W. Chi-Hsiang and T. Dwen-Ren, "Integrated installing iso 9000 and iso 27000 management systems on an organization," in *43rd Annual 2009 International Carnahan Conference on Security Technology*. IEEE, oct 2009. [Online]. Available: <http://ieeexplore.ieee.org/abstract/document/5335527/>
- [12] L. P. Tixeco, G. S. Pérez, and J. de Jesús Vázquez Gómez. (2015, jun) Gestión de seguridad de la información basado en el maagtici para programas académicos en instituciones de educación superior. [Online]. Available: <http://ru.tic.unam.mx:8080/tic/bitstream/handle/123456789/1690-135.pdf?sequence=1&isAllowed=y>
- [13] V. D. Freitas, "Propuesta de metodología de gestión de seguridad de las tics para el sector universitario venezolano," *Espacios*, vol. 31, no. 1, p. 16, 2010. [Online]. Available: <http://www.revistaespacios.com/a10v31n01/10310151.html>

Variante del Algoritmo de Proyecciones Afines con Error Codificado y Stop and Go para Comunicación 16-QAM

F. Zamora Cuevas, C. E. López Reyes, J. G. Avalos Ochoa, G. Avalos Arzate

Instituto Politécnico Nacional

ESIME Culhuacán

Col. San Fco. Culhuacán, C.P.04430, Ciudad de México

fzamorac1300@alumno.ipn.mx, clopezr1305@alumno.ipn.mx, javaloso@ipn.mx, gavalos580@hotmail.com

Resumen— La variante de proyecciones afines con Stop and Go (Affine Projection-Like, APL-SG), es un algoritmo que presenta como principal característica una alta velocidad de convergencia y una baja complejidad computacional, sin embargo, para aplicarlo en ecualizadores ciegos de canal es necesario el uso de diferentes técnicas que le permitan un correcto funcionamiento cuando se usan señales QAM, lo cual provoca un aumento en la carga computacional haciendo que las implementaciones en hardware sean mas complejas. En este artículo se presenta una variante del algoritmo APL-SG, en el cual se aplica la teoría de la codificación del error con el fin de establecer un umbral que sirva para controlar el número de ocasiones que el algoritmo actualiza sus coeficientes. El algoritmo propuesto se simuló en un sistema de transmisión de señales de 16-QAM, los resultados demuestran una mayor velocidad de convergencia con respecto a la versión original del algoritmo y una reducción en el número de actualizaciones, disminuyendo así considerablemente la carga computacional.

Keywords—Filtrado adaptativo, Ecualizador ciego, Algoritmo de proyecciones afines.

I. INTRODUCCIÓN

En los sistemas de comunicación digital, generalmente se presenta un problema conocido como interferencia entre símbolos (Inter Symbols Interference - ISI), este es un efecto que se produce debido al uso de múltiples trayectorias o bien a errores de sincronización causados por canales con un ancho de banda insuficiente.

Para compensar los efectos producidos por este tipo de problemas, se ha implementado el uso de ecualizadores ciegos de canal, los cuales se encargan de recuperar la señal transmitida mediante el uso de filtros adaptativos, obteniendo así una señal similar a la transmitida sin algún tipo de alteración. Para lograr esto, los ecualizadores comúnmente realizan una etapa previa de entrenamiento o bien usan una señal de referencia para tener conocimiento de la señal deseada, sin embargo, en ocasiones se desconoce esta señal por lo que es necesario el uso de filtros adaptativos ciegos [1].

Actualmente existen diversos algoritmos aplicados en ecualizadores de canal que utilizan técnicas para reducir la complejidad computacional. Por ejemplo, en [2] se presenta el

algoritmo de proyecciones afines con error codificado de modulo constante (ECAP-CM), en el cual se usa la teoría del error codificado con el fin de disminuir el número de operaciones de punto flotante requeridas para el cálculo de sus coeficientes, no obstante, su complejidad computacional aún permanece elevada debido a las operaciones de inversión de matriz necesarias para ejecutar el algoritmo, además, sólo es implementado para señales 4-QAM.

En [3] se presenta la variante del algoritmo de proyecciones afines con modulo constante (APL-CM), el cual posee una alta velocidad de convergencia, así como una baja complejidad computacional en comparación con algunas versiones convencionales, sin embargo, este algoritmo está dedicado únicamente a señales QPSK.

Otro método que ha sido utilizado para reducir la carga computacional es la teoría de conjunto de membresías (Set-Membership) [4], la cual al aplicarse reduce las operaciones matemáticas debido a que el cálculo de los coeficientes solo se realiza cuando el valor de la señal de error es mayor a la de un umbral establecido. No obstante, el cálculo del umbral puede resultar complicado, ya que su valor esta relacionado con la varianza de la señal de error, la cual en algunas ocasiones no esta disponible para el diseñador.

Se han desarrollado también algoritmos que se basan en la técnica de decisión directa “stop and go”, mediante la cual se ha podido implementar ecualizadores ciegos de canal para señales QAM [5], mostrando resultados favorables. A pesar de ello, estas implementaciones cuentan con un nivel de desajuste elevado además de una velocidad de convergencia lenta.

La variante del algoritmo de proyecciones afines con stop and go (APL-SG) [6], presenta buenas características de convergencia en la ecualización de señales 16-QAM, ya que posee una velocidad de convergencia alta y una baja complejidad computacional debido a que no requiere el uso de operaciones complejas como la inversión de matriz, sin embargo, el número de operaciones y tiempo de ejecución pudiera ser elevado para algunas implementaciones en hardware.

Por otra parte, la variante del algoritmo de proyecciones afines con error codificado ECAPL [7], presenta resultados favorables ya que mantiene una alta velocidad de convergencia y disminuye el número de actualizaciones requeridas para el cálculo de sus coeficientes en más de un 90% en comparación con el algoritmo APL, siendo ésta una mejora notable ya que no se requiere un hardware con demasiadas prestaciones, no obstante, ésta variante no ha sido implementada en aplicaciones de ecualización ciega de canal.

En este trabajo se presenta la variante del algoritmo de proyecciones afines con stop and go utilizando la teoría del error codificado (ECAPL-SG). El proceso de codificación no modifica el algoritmo, manteniendo así las características del mismo y reduciendo considerablemente la carga computacional. Este algoritmo fue probado en un ecualizador ciego de canal, los resultados demuestran que alcanza un nivel de desajuste similar al del algoritmo APL-SG y una mayor velocidad de convergencia.

II. VARIANTE DEL ALGORITMO DE PROYECCIONES AFINES CON STOP AND GO (APL-SG).

La ecuación (1) muestra el cálculo de los coeficientes para cada iteración del algoritmo APL [8].

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{|\mathbf{X}(k)\mathbf{e}(k)|^2}{|\mathbf{X}^t(k)\mathbf{X}(k)\mathbf{e}(k)|^2} \mathbf{X}(k)\mathbf{e}(k) \quad (1)$$

donde $\mathbf{w}(k)$ representa los coeficientes del filtro, $\mathbf{X}(k)$ la matriz que contiene los vectores de la señal de entrada al filtro $\mathbf{X}(k) = [x(k), x(k-1), \dots, x(k-L+1)]$, L representa el número de vectores de re-uso (también llamado orden de proyección), el superíndice t denota transposición y $\mathbf{e}(k)$ es la señal de error. Para transmisión de señales QAM la señal de error se puede definir por la técnica de decisión directa stop and go [9], como se muestra en la ecuación (2).

$$e(k) = f(k)e_d(k) \quad (2)$$

donde $e_d(k)$ representa la señal de error directo del algoritmo denotado como:

$$e_d(k) = s(k-\Delta) - \mathbf{X}^T(k)\mathbf{w}(k+1) \quad (3)$$

donde $s(k)$ es la señal transmitida. Por otra parte, $f(k)$ es una bandera que indica si este cálculo es correcto mediante la siguiente comparación:

$$f(k) = \begin{cases} 1, & \text{si } \text{sgn}(e_d(k)) = \text{sgn}(e_s(k)) \\ 0, & \text{si } \text{sgn}(e_d(k)) \neq \text{sgn}(e_s(k)) \end{cases} \quad (4)$$

El cálculo de la bandera se realiza a partir de una comparación de la función signo (sgn) de dos señales de

error, el error directo y el error secundario; el error secundario está representado por la ecuación (5):

$$e_s(k) = \gamma \text{sgn}(\mathbf{X}^T(k)\mathbf{w}(k+1)) - \mathbf{X}^T(k)\mathbf{w}(k+1) \quad (5)$$

donde $\gamma = \mathbf{E}|s|^2 / \mathbf{E}|s|_1$

Como se puede observar en la ecuación (1) el algoritmo no necesita realizar operaciones de inversión de matriz, como se hace en la versión original del algoritmo de proyecciones afines (APA) [10], por otra parte, es necesario el uso de la técnica stop and go presente en la ecuación (2) para que el algoritmo pueda ser aplicado a señales QAM.

III. VARIANTE DEL ALGORITMO DE PROYECCIONES AFINES CON ERROR CODIFICADO Y STOP AND GO (ECAPL-SG).

Con el fin de reducir la carga computacional del algoritmo APL-SG, se propone codificar la señal del error como se muestra en la ecuación (6).

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{|\mathbf{X}(k)\mathbf{e}(k)|^2}{|\mathbf{X}^t(k)\mathbf{X}(k)\mathbf{e}(k)|^2} \mathbf{X}(k)\mathbf{C}[\mathbf{e}(k)] \quad (6)$$

La codificación se realiza al cuantizar una muestra de la señal de error para posteriormente asignarle un código digital. El proceso de codificación se realiza por medio de la ecuación (7) [7].

$$\mathbf{C}[\mathbf{e}(k)] = \text{round} \left[\frac{\mathbf{e}(k)}{Res} \right] \quad (7)$$

donde round representa una función de redondeo y Res denota la resolución del codificador, la cual es calculada como:

$$Res = \left[\frac{e_{max}}{2^{b-1}} \right] \quad (8)$$

e_{max} representa el error máximo permitido, el cual no es mayor del 90% de la amplitud de la señal de entrada y b es la cantidad de bits usados para la codificación del error.

Cabe mencionar que entre mayor sea el número de bits, mayor será la precisión de la resolución, esto afecta directamente la señal del error presente en la ecuación (7), haciendo que el valor del error real sea más elevado, lo cual ocasiona que la velocidad de convergencia aumente considerablemente, sin embargo, si la cantidad de bits es muy alta, el algoritmo podría divergir.

Al codificar el error no se altera la ecuación de actualización de coeficientes, por lo que el algoritmo mantiene las propiedades de la versión original.

Otro aspecto importante, es que al aplicar la codificación del error en el algoritmo APL-SG, la señal del error codificado puede llegar a obtener valores entre "0" o "1" cuando el algoritmo se encuentra en el estado estacionario. De esta forma, es posible establecer una regla para la actualización de sus coeficientes, como se muestra en (9).

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{w}(k) + \frac{\|\mathbf{X}(k)\mathbf{e}(k)\|^2}{\|\mathbf{X}^T(k)\mathbf{X}(k)\mathbf{e}(k)\|^2} \mathbf{X}(k)C[\mathbf{e}(k)], & \text{para otros valores} \\ \mathbf{w}(k), & \text{si } C[\mathbf{e}(k)] = 0 \text{ o } 1 \end{cases} \quad (9)$$

Así, cuando el error codificado sea diferente de “0” o “1”, el algoritmo actualizará sus coeficientes, de lo contrario se mantendrá el coeficiente actual. De esta forma, la carga computacional se reduce considerablemente.

IV. PRUEBAS Y RESULTADOS

El algoritmo propuesto se evaluó en un sistema ecualizador ciego de canal como el que se muestra en la Fig. 1. Las pruebas se realizaron en el software MATLAB, en donde se transmitió una señal $s(k)$ compuesta de una secuencia aleatoria de 16-QAM y el error se codificó a 8 bits. El canal considerado para transmitir la señal tiene la siguiente función de transferencia: $C(z)=0.5+1.2z^{-1}+1.5z^{-2}+z^{-3}$.

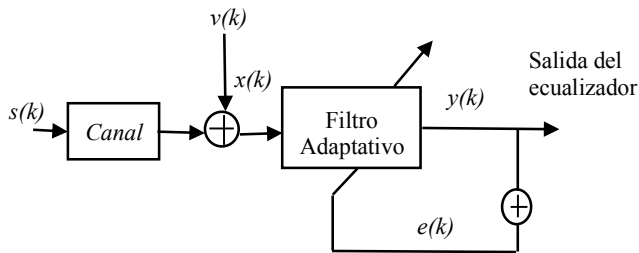


Fig. 1. Diagrama a bloques del ecualizador ciego de canal.

A la salida del canal $x(k)$ se agregó un ruido Gaussiano blanco $v(k)$ con varianza unitaria y con una SNR (Relación señal a ruido) de 30 dB.

En la Fig. 2 se observa la señal transmitida $s(k)$ mientras que en la Fig. 3 se aprecia la señal recibida a la salida del canal, en la que es posible observar que la señal transmitida está totalmente distorsionada.

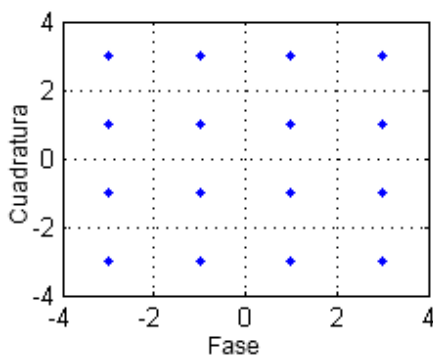


Fig. 2. Señal transmitida.

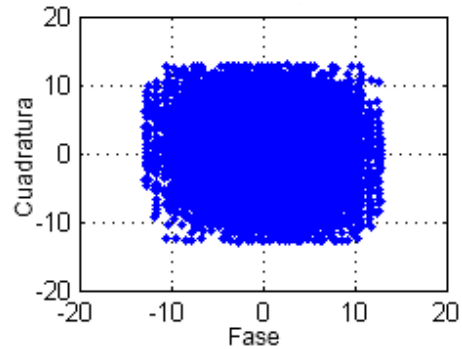


Fig. 3. Señal a la salida del canal.

Para la ecualización de la señal se utilizaron los algoritmos APL-SG y el algoritmo propuesto. Los resultados de la simulación se obtuvieron al realizar el promedio de 100 experimentos individuales con 30000 iteraciones.

En las Figs. 4 y 5 se muestran las señales a la salida del filtro adaptativo. Se observa que los algoritmos realizan de manera adecuada el proceso de ecualización y presentan una señal muy similar a la original. Los resultados de ambos algoritmos son semejantes lo cual comprueba que el algoritmo propuesto entrega un resultado equiparable con respecto al algoritmo APL-SG.

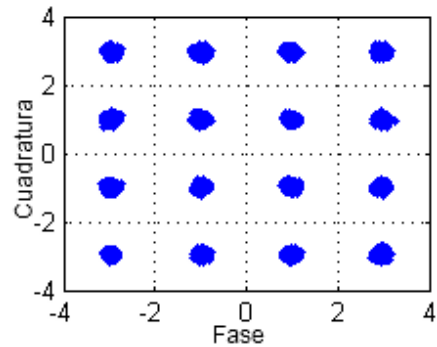


Fig. 4. Señales ecualizadas usando el algoritmo APL-SG.

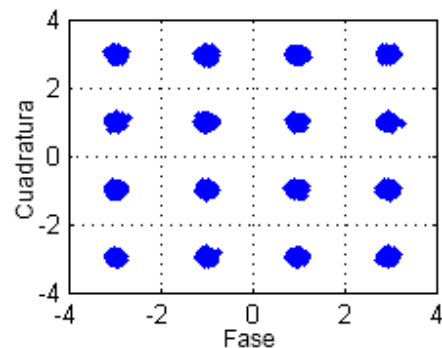


Fig. 5. Señales ecualizadas usando el algoritmo ECAPL-SG.

En la Fig. 6 se muestran las curvas de aprendizaje obtenidas para ambos algoritmos. Donde se observa que el

algoritmo propuesto presenta una velocidad de convergencia mayor que la del algoritmo APL-SG y un nivel mínimo de error cuadrático medio (Mean Square Error- MSE) muy similar.

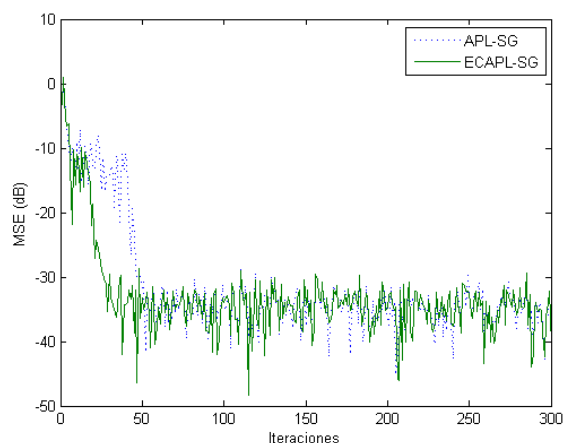


Fig. 6. Curvas de aprendizaje de los algoritmos APL-SG y ECAPL-SG.

En la Tabla 1 se muestra el tiempo y el número de iteraciones en las que se ejecutan los algoritmos. Los resultados demuestran que el algoritmo ECAPL-SG reduce en un 75% el tiempo de ejecución y un 80% el número de actualizaciones con respecto al algoritmo APL-SG. Las pruebas fueron realizadas en un equipo de cómputo con procesador Core 2 Duo T3700 @ 2.00 GHz y memoria RAM de 3 GB.

TABLA 1. NUMERO DE ITERACIONES Y TIEMPO DE EJECUCIÓN DE LOS ALGORITMOS.

| Algoritmo | Iteraciones | Tiempo total |
|-----------|------------------|--------------|
| APL-SG | 2,996,600 (100%) | 42.473s |
| ECAPL-SG | 574,416 (19%) | 10.484s |

V. CONCLUSIONES

En este trabajo se presentó una variante del algoritmo de proyecciones afines con error codificado y stop and go para la equalización ciega de canal. El algoritmo propuesto mantiene una baja complejidad computacional y una alta velocidad de convergencia ya que no se modifica la ecuación de actualización de coeficientes del algoritmo base.

Los resultados muestran que existe un aumento en la velocidad de convergencia comparado con el algoritmo APL-SG y un menor tiempo de ejecución debido al uso de la técnica de error codificado, la cual reduce considerablemente el número de actualizaciones del algoritmo.

Así mismo el algoritmo presentado permite ser implementado en hardware de manera más eficiente ya que no

se necesitarían demasiados recursos a nivel procesador, los cuales pueden ser aprovechados para efectuar otro tipo de tareas. A partir de los resultados obtenidos se comprueba que el uso del algoritmo propuesto es factible en aplicaciones de equalización ciega de canal en comunicación de datos 16-QAM.

REFERENCIAS

- [1] Z. Ding and Y. Li, "Blind equalization and identification". CRC press, 2011.
- [2] A. A. Vázquez, J. G. Avalos, F. A. Serrano and G. Avalos. "Algoritmo de Proyecciones Afines con Error Codificado de Modulo Constante". Revista Aristas, Vol. 6, Número II, 2017.
- [3] F. Zamora, C. López, E. Pichardo, J. G. Avalos, G. Avalos. "Una clase de algoritmo de proyecciones afines de módulo constante". Impulso Tecnológico. En prensa.
- [4] Y. Li, "A set-membership approach to blind channel equalization algorithm". In Fifth International Conference on Machine Vision (ICMV 12) (pp. 87841W-87841W). International Society for Optics and Photonics, 2012.
- [5] S. Abrar, "Stop-and-go algorithms for blind channel equalization in QAM data communication systems". In NCET2004 National Conference on Emerging Technologies. 2004.
- [6] C. E. López-Reyes, M. Avalos-Vega, F. Zamora-Cuevas, J. G. Avalos-Ochoa, F. Serrano-Orozco, "Variante del algoritmo de Proyecciones afines con stop and go para comunicación de datos 16-QAM", Congreso Nacional de Ingeniería Electromecánica y de Sistemas (CNIES 2017).
- [7] A. Rodríguez, J. C. Sanchez, J. G. Avalos. "Error Coded Affine Projection Like Algorithm". International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE 2015), pp. 70-75, 2015.
- [8] M. Z. A. Bhotto and A. Antoniou. "Affine-projection-like adaptive-filtering algorithms using gradient-based step size". IEEE Transactions on Circuits and Systems I: Regular Papers, 61(7), 2048-2056, 2014.
- [9] G. Picchi and G. Prati. "Blind Equalization and Carrier Recovery using a 'Stop-and-Go' Decision-Directed Algorithm" IEEE Trans. Commun., COM-35:877-887, 1987.
- [10] P. Diniz. Adaptive Filtering: Algorithms and Practical Implementation. Springer, 2013.

Modelado Básico con Autómatas Celulares para Epidemias de Gusanos Informáticos

Eliezer Erosa García

Sistemas en Tiempo Real SEPI ESIME Culhuacan
Instituto Politécnico Nacional
Ciudad de México
elikz18@gmail.com

Jose Luis Cano Rosas

Sistemas en Tiempo Real SEPI ESIME Culhuacan
Instituto Politécnico Nacional
Ciudad De México
lucskyr@gmail.com

Diana Lizet González Baldovinos

Sistemas en Tiempo Real SEPI ESIME Culhuacan
Instituto Politécnico Nacional
Ciudad De México
glez_lizet@hotmail.com

Pedro Guevara López

Sistemas en Tiempo Real SEPI ESIME Culhuacan
Instituto Politécnico Nacional
Ciudad De México
pguevara@real-time.com.mx

Resumen— Las redes de cómputo son un objetivo de ataques por malware como virus y gusanos informáticos, siendo estos últimos muy importantes debido a su evolución altamente peligrosa y a su autopropagación en una red. Por ello es necesario el entendimiento de su forma de operar, efectos y velocidad de propagación; una herramienta para ese estudio son los modelos epidemiológicos clásicos que generalmente son del tipo cuantitativo y basados en ecuaciones diferenciales pero que no son capaces de presentar el patrón de la epidemia; por ello se plantea el modelado cuantitativo y cualitativo basado en un autómatas celular, presentando como caso de estudio el modelo Susceptible-Infectado.

Keywords— *autómata celular; gusano informático; epidemiología, topología;*

I. INTRODUCCIÓN

Internet se ha convertido en una herramienta de trabajo y comunicación indispensable para la mayoría de las empresas a nivel mundial, ésto ha generado una gran evolución, ya que Internet, con su variedad de tecnologías, ha sido promotor del cambio y principal constructor de una nueva sociedad global en la era digital. En este sentido todos los ámbitos de las sociedades modernas actuales han sufrido una transformación derivada de la penetración de las redes de cómputo y en particular de la conexión a gran escala de los distintos sistemas que integran Internet. Para la gran mayoría de las personas, Internet se ha vuelto un modo de vida en el cual el uso de las redes convergentes de alcance mundial les permite llevar a cabo actividades tales como transacciones bancarias a distancia, acceder a sistemas de reservación de vuelos, realizar transacciones de compra y venta en línea e incluso recibir servicios diversos del tipo B2C (Business to Consumer), entre ellos, servicios educativos. Las nuevas generaciones han nacido prácticamente en un mundo de comunicación y conectividad global que pone a su alcance nuevas formas de desarrollo social y económico. Por todo esto es necesario mantener altos niveles de seguridad ante malware como los

gusanos informáticos; por ello la necesidad de su estudio respecto a su velocidad y patrones de propagación.

A. Gusanos Informáticos

Se suele definir a los gusanos informáticos como programas autónomos que son capaces de replicarse a sí mismos a través de la red sin que la intervención del usuario sea requerida, simplemente requieren una conexión a otros hosts para que la infección pueda propagarse.

Debido a su naturaleza, un gusano informático tiende a propagarse más rápido que un virus informático ordinario, pues estos últimos si suelen requerir cierta interacción del usuario para poder propagarse, como de menciona en [1].

B. Epidemiología

La epidemiología se encarga de estudiar el comportamiento y desarrollo de enfermedades infecciosas que afectan a la población, por lo cual ha sido un área de interés para el ser humano y así poder determinar formas de combatirlas, como se menciona en [2] y en [3].

1) Modelo epidemiológico SI

En 1927, Kermack y McKendrick desarrollaron el modelo SIR, un modelo matemático basado en un sistema de ecuaciones diferenciales que describe el comportamiento de una infección sobre una población (N). Este modelo sirvió como referencia para que posteriormente se desarrollaran variaciones, como por ejemplo el modelo SI, una versión simplificada del modelo SIR [2][4].

El modelo SI considera una población (N) que puede ser separada en 2 grupos: a) susceptibles S(t) y b) infectados I(t), a partir de lo cual se tiene que:

$$S(t) + I(t) = N \quad (1)$$

Siendo β la velocidad de infección de la enfermedad en individuos/segundo, se tiene que:

$$\beta S(t) I(t) = d I(t) / dt \quad (2)$$

Finalmente, se tiene que la relación entre los individuos infectados y los individuos susceptibles es:

$$d S(t) / dt = -d I(t) / dt \quad (3)$$

Donde a partir de (3) podemos determinar que las poblaciones $I(t)$ y $S(t)$ cumplen con el principio de proporcionalidad, como se describe en [5].

II. DESARROLLO

El caso de estudio en esta investigación es el modelo SI, de acuerdo a [6] asume que una red con n hosts, los hosts en estado susceptible S entran en contacto directo con hosts en estado infectado I quienes pueden contagiar o cambiar su estado también a I con una velocidad de infección. Finalmente, el número de hosts n es igual a la suma de los hosts en estado infectado I más los hosts en estado susceptible S . En este trabajo se plantea el uso de un autómata celular de dos dimensiones para el conteo de hosts y patrón epidemiológico.

A. Autómatas celulares

De acuerdo con [7], un autómata celular es un modelo discreto utilizado en diferentes áreas de estudio para representar sistemas complejos. Se compone de un conjunto de células dispuestas en una rejilla denominada lattice, en donde cada célula posee un estado de entre un conjunto de estados posibles. Además, cada célula puede interactuar con las otras células en un área determinada llamada vecindario a través de reglas de transición, que determinan la dinámica del sistema, como se menciona en [8].

B. Modelado del autómata celular

1) Lattice

La construcción de una red de host por medio de una lattice supone el reto de representar algunas de las características más importantes de la red a costa de otras que pudieran resultar difíciles de representar por este mismo modelo.

En este caso se consideran a los hosts activos de la red y se posicionan en la lattice procurando que las células sean conexas con respecto al vecindario y considerando su relación más próxima con otros hosts, si pertenecen al mismo segmento, por ejemplo. Todos los elementos como switch o routers que formen parte de la red en cuestión, no se consideran en la representación de la lattice, puesto que para la propagación de gusanos informáticos estos elementos no actúan como hosts, si no como un medio de transmisión.

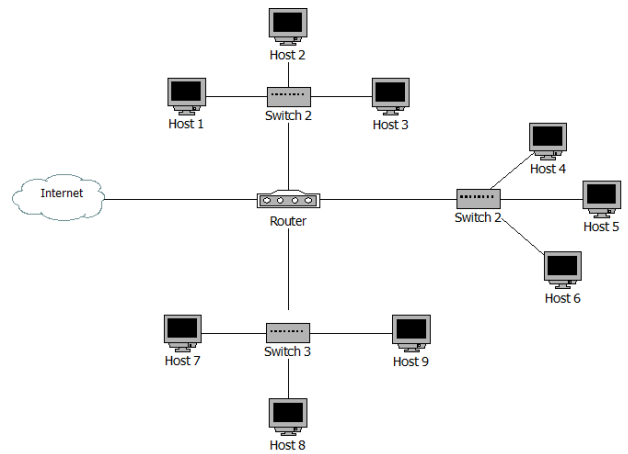


Fig. 1 Diagrama representativo de conexión de nodos conexas en una red local.

Consideramos una lattice de 2 dimensiones de tamaño $N \times M$ y en cada célula donde exista un host activo toma el valor de 1, los hosts inactivos toman el valor 0. El host que contraiga la infección inicial toma el valor 2 y la propagación comienza.

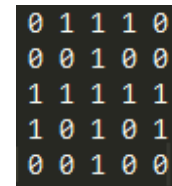


Fig. 2 Representación inicial de una red de hosts mediante un autómata celular de 5×5 .

2) Vecindario

El vecindario define el radio de acción que tiene una célula sobre su entorno. Dependiendo de cómo se caracterice el sistema, el tamaño y forma del vecindario puede variar.



Fig. 3 Diferencia entre una transición utilizando un vecindario de Von Neumann (izquierda) y utilizando un vecindario de Moore (Derecha).

Como se observa en la figura 3, células a las que un tipo de vecindario puede llegar en una sola evolución, puede tomar 2 o más evoluciones para otro tipo de vecindario. Este factor puede interpretarse como la velocidad que le toma a un gusano informático propagarse a través de una red.

3) Reglas de Transición

Las reglas de transición determinan las condiciones de la dinámica del autómata, en contraste con el modelo SI, las reglas de transición describen la facilidad con la que se propaga la infección.

Se considera que cada célula está en uno de 3 estados posibles a) inactivo, b) susceptible o c) infectado. En primera instancia, todas las células que no sean inactivas son inmediatamente susceptibles.

Para cada célula susceptible, si existe una célula infectada en su vecindario, la célula actual pasará al estado infectado inmediatamente. Las células que estén en estado infectado no podrán cambiar de ese estado.

4) *Conteo de población infectada y susceptible*

En cada evolución del autómata se realiza el conteo de células totales infectadas y células susceptibles. Esto lo usaremos para observar

C. *Programación del Autómata Celular*

Para la construcción del autómata celular se optó por utilizar el lenguaje de programación Python, que ofrece alternativas asequibles para la representación gráfica del autómata, así como otras virtudes, entre ellas la simplicidad del código.

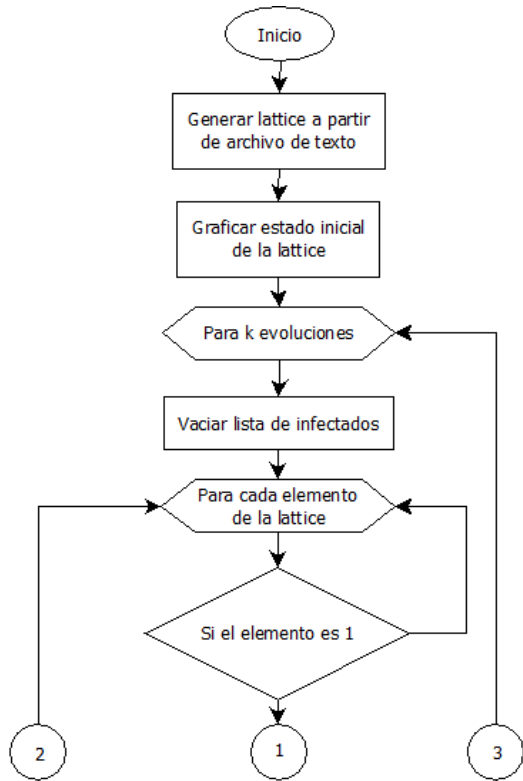


Fig 4.1. Primera etapa del diagrama de flujo del funcionamiento del autómata celular.

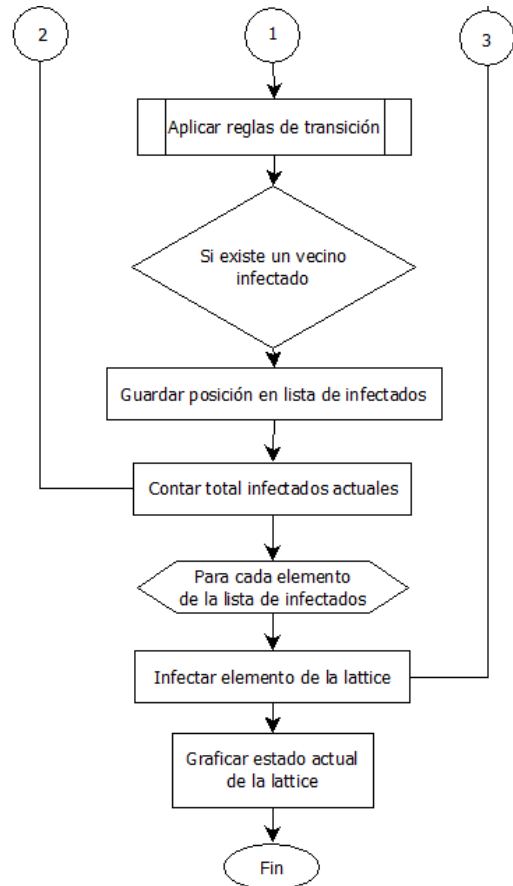


Fig 4.2. Segunda etapa del diagrama de flujo del funcionamiento del autómata celular.

En las fig. 4.1 y 4.2 se describe el funcionamiento del programa por un diagrama de flujo, donde la lattice se lee desde un archivo de texto que puede ser modificado directamente para proponer las condiciones de la lattice que se requieran. El programa muestra una animación en línea del proceso de infección.

III. RESULTADOS

A continuación, se muestran los resultados obtenidos en una simulación para una red propuesta de hosts con una infección cerca de la parte central de la red.

Automaton

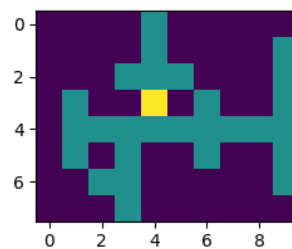


Fig 5. Estado inicial del autómata celular. La célula infectada se muestra en amarillo, las susceptibles se muestran en azul.

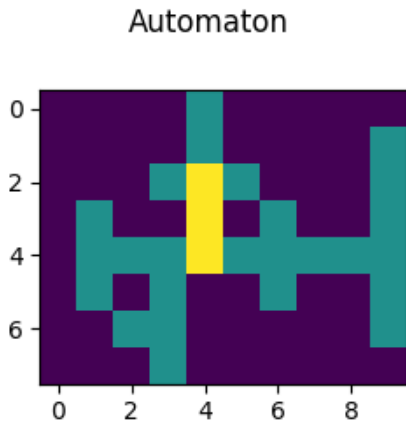


Fig 6. Autómata celular al cabo de 1 evolución.

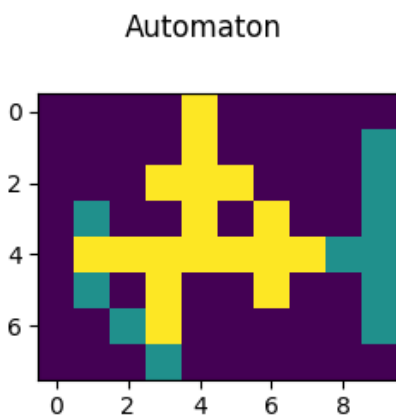


Figura 7. Autómata celular al cabo de 4 evoluciones.

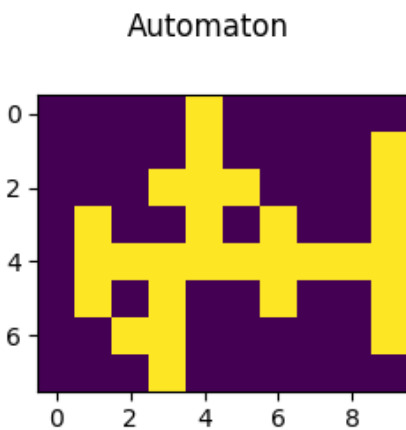


Fig 8. Autómata celular una vez que se han infectado todos los hosts de la red.

Resulta interesante observar el patrón de propagación mediante el autómata celular ya que permite observar el comportamiento de la epidemia sin hacer uso de algún modelo por ecuaciones diferenciales, así mismo resalta el hecho de que

podemos obtener diferentes patrones de propagación modificando los criterios de vecindario y/o reglas de transición del autómata, lo cual deriva en infinitas posibilidades.

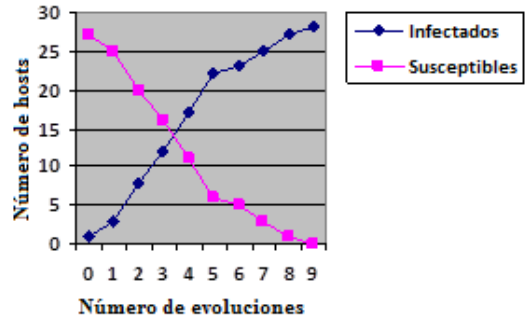


Fig 9. Gráfica del conteo de hosts susceptibles e infectados de la simulación del autómata celular.

Mediante los resultados obtenidos en la figura 9 se puede observar la similitud que existe en el comportamiento del a epidemia del modelo por autómatas celulares con relación al modelo SI.

IV. CONCLUSIONES

Es importante destacar la similitud que se presenta en el comportamiento del modelo SI, que es un modelo cuantitativo y el comportamiento del modelo con autómata celular, que es un modelo cualitativo, esto permitirá inferir en un futuro que características se requieren modificar en el presente modelo con autómatas celulares para obtener un resultado más acercado a uno cualitativo.

REFERENCIAS

- [1] N. Weaver, V. Paxson, S. Staniford and R. Cunningham, "A Taxonomy of Computer Worms."
- [2] W. Kermack and A. McKendrick, A contribution to the mathematical theory of epidemics. [S.l.]: [s.n.].
- [3] S. Ching Fu and G. Milne, "Epidemic Modelling Using Cellular Automata."
- [4] L. Misici and F. Santarelli, "Epidemic Propagation: An Automaton Model as the Continuous SIR Model", Applied Mathematics, vol. 04, no. 10, pp. 84-89, 2013.
- [5] P. Guevara López, J. Valdez Martínez, J. Audelo González and G. Delgado Reyes, "Aproximación numérica del modelo epidemiológico SI para la propagación de gusanos informáticos, simulación y análisis de su error", Tecnura, vol. 18, pp. 12-23, 2014.
- [6] Doracelly Hincapié, Juan Ospina, "Bases para la Modelación de Epidemias: el Caso del Síndrome Respiratorio Agudo Severo en Canadá", Revista de Salud Pública, Número 1, Volumen 9, año 2007. Universidad Nacional de Colombia, Bogotá, Colombia.
- [7] Y. Naigong, L. Mingai and R. Xiaogang, "Applications of cellular automata in complex system study", international journal of information and systems sciences, vol. 1, pp. 302-310, 2005.
- [8] J. Schiff, Introduction to cellular automata.

Impacto de dos Mecanismos de Planificación de Tiempo Real con RT-Linux sobre una Computadora Embebida

Diana Lizet González Baldovinos

Sistemas en Tiempo Real SEPI ESIME Culhuacan
Instituto Politécnico Nacional
Ciudad de México
glez_lizet@hotmail.com

Eliezer Erosa García

Sistemas en Tiempo Real SEPI ESIME Culhuacan
Instituto Politécnico Nacional
Ciudad de México
elikz18@gmail.com

Jose Luis Cano Rosas

Sistemas en Tiempo Real SEPI ESIME Culhuacan
Instituto Politécnico Nacional
Ciudad de México
lucskyr@gmail.com

Pedro Guevara López

Sistemas en Tiempo Real SEPI ESIME Culhuacan
Instituto Politécnico Nacional
Ciudad de México
pguevara@real-time.com.mx

Resumen— Esta investigación analiza el impacto de dos mecanismos de planificación en tiempo real sobre los tiempos de respuesta en la extensión al kernel de Linux `PREEMPT_RT`: `FIFO` y `Round Robin`. El banco de prueba se integró por la ejecución de una tarea o proceso de alta complejidad computacional temporal, generada a través de un algoritmo de inversión de matrices por Gauss-Jordan. Los experimentos se realizaron con inversiones de matriz de distinta dimensión, cada uno con 1000 eventos; se implementaron ambos mecanismos de planificación con niveles de prioridad, baja, media y alta. De los experimentos se observó que utilizando la política `Round Robin` se obtienen tiempos de respuesta menores, y varianzas pequeñas a diferencia de la política `FIFO`.

Palabras clave — *tiempos de respuesta; tiempo real; mecanismo de planificación; RT-Linux; PREEMPT_RT*

I. INTRODUCCION

Los Sistemas en Tiempo Real de acuerdo a [1], por definición interactúan con el mundo real; para desarrollarlos se pueden utilizar Computadoras de Placa Reducida SBC (Single Board Computer), debido a que tienen soporte para dispositivos de entrada/salida, son pequeñas, de bajo consumo energético, de bajo costo y cuentan con soporte para software libre; estas computadoras están fabricadas en un solo circuito, bajo una arquitectura SoC (System on a Chip [2]). Este tipo de computadoras puede ser diseñado como dispositivos de propósito general y usualmente están disponibles con sistemas operativos de tiempo compartido como GNU/Linux. Como ejemplo se pone Raspbian, que es una distribución del sistema operativo GNU/Linux basado en Debian Jessie (Debian 8.0) para la placa computadora Raspberry Pi, el núcleo de este sistema operativo es Linux y el tipo de núcleo es monolítico. Raspbian es un sistema operativo de tiempo compartido, por lo tanto, para obtener funcionalidad de tiempo real existe una serie de parches al kernel de Linux los cuales contienen código en C y necesitan compilarse.

La configuración del kernel de Linux para obtener funcionalidades de tiempo real utilizada en este proyecto se obtiene con el parche `PREEMPT_RT`, éste proporciona la capacidad de ejecutar tareas de tiempo real y manejador de interrupciones en la misma máquina que Linux estándar. Estas tareas y los manejadores se ejecutan cuando se necesita desalojo de lo que estuviera ejecutando Linux. El peor caso de tiempo es cuando se detecta la interrupción hardware y el procesador ejecuta la primera instrucción del manejador de la interrupción. Este tiempo es del orden de los 10 microsegundos en la plataforma x86. Con ésta extensión el kernel nativo Linux se ejecuta como un thread (hilo de ejecución) de menos prioridad que las tareas de tiempo real. Con este diseño, las tareas de tiempo real y los manejadores de interrupciones nunca se ven retrasados por operaciones que no son de tiempo real.

De acuerdo a [3], las características de RT-Linux son las siguientes: Soporte de múltiples arquitecturas y válida para arquitecturas multiprocesador. Gestión de procesos: Planificación, amplio rango de prioridades, creación y borrado de threads, etc. Acceso directo al hardware (puertos e interrupciones). Comunicación con procesos: Semáforos, Mutex, control de inversión de prioridades, memoria compartida y FIFO's. Estructura modular para crear sistemas pequeños. Tiempo y relojes: Resolución de nanosegundos. No relojes de usuario. Facilidades para añadir nuevos relojes hardware.

La planificación es un conjunto de políticas y mecanismos incorporados al sistema operativo que rigen el orden en que los procesos deben ejecutarse y administrar el uso de CPU, el objetivo de la planificación es optimizar el rendimiento del sistema operativo. Para indicar al Sistema Operativo que un proceso es urgente o crítico, se define un valor que identifica la urgencia con que debe ejecutarse, esta es la prioridad que tiene frente a los otros procesos que se estén ejecutando en el sistema. La prioridad se expresa como un número entero. Si un proceso A tiene asignada una prioridad p y otro una prioridad q , siendo

$p > q$, el proceso A será más prioritario que el B y por lo tanto se verá más favorecido durante su ejecución [4]. De acuerdo a [5] GNU/Linux clasifica en dos a las prioridades de procesos, estas son: prioridades de tiempo real, que abarcan un rango de valores desde 1-99 y prioridades de tiempo compartido o para tareas normales, mismas que incluyen valores desde 100 hasta 140. En el sistema operativo GNU/Linux, la política de planificación asignada por defecto es `SCHED_OTHER`, la cual involucra a la política estándar de tiempo compartido Round Robin con `timeslice` (rodaja de tiempo) variable, administra procesos convencionales y el valor de prioridad por defecto es 0.

La parte central de este trabajo se enfoca en el planificador por desalojo de prioridades, observando el comportamiento de los tiempos de respuesta con el uso de mecanismos de planificación de tiempo real. En la metodología se presenta la implementación del mecanismo de planificación, para establecer políticas de tiempo real y se presentan las herramientas para análisis estadístico de los tiempos de respuesta, generados por el proceso de inversión de matrices, como tarea de alta complejidad computacional temporal.

II. METODOLOGIA

A. Algoritmo de inversión de matrices

Apoyado en el trabajo de [6], [7] y [8] se utiliza un algoritmo como objeto de prueba programado en ANSI C denominado `matrices.h`, el cual se describe como una biblioteca de funciones para la inversión de matrices, empleando el método de Gauss-Jordan. Las funciones que integra `matrices.h` son las siguientes: `iniciar_matriz()`, `llenar_matriz()`, `suma_matriz()`, `producto_matriz()` e `inversa()`. El algoritmo tiene el propósito de generar un proceso de alta carga computacional temporal de complejidad NP y se emplea para hacer la medición de los tiempos de ejecución, la inversión de matrices se efectúa con las siguientes dimensiones: 2×2 , 4×4 , 8×8 , 16×16 , 32×32 , 64×64 y 128×128 , para cada dimensión se realizan 1000 inversiones, con el objetivo de observar el comportamiento de los tiempos de ejecución como métrica cuantitativa. se eligieron operaciones matriciales debido a que éstas tienen muchas aplicaciones en áreas como control, filtrado, imágenes, robótica, videojuegos, etc. En la Figura 1 se muestra el pseudocódigo del programa de inversión de matrices.

1. Inicio
2. Definir dimensión de matriz: M
3. Definir variable $ITERACIONES=1000$
4. Declarar variables para tiempos y k
5. Crear fichero de datos
6. Llamar función `iniciar_matriz()`
7. Desde $k=0$ hasta $k < ITERACIONES$ Hacer
 - Medir Tiempo inicial
 - Llamar función `matriz_inversa()`
 - Medir Tiempo final
 - Convertir Tiempo inicial y final a milisegundos
 - Tiempo de ejecución = Tiempo final - Tiempo inicial
 - Mostrar Tiempo de ejecución
 - Guardar Tiempo de ejecución en fichero de datos
8. Fin

Fig. 1 Pseudocódigo del programa de inversión de matrices

La medición de los tiempos de respuesta se realiza dentro del algoritmo, para efectuar las mediciones, existe una serie de funciones dentro de la biblioteca `time.h`, la función que se utiliza para medir tiempos de ejecución es `clock_gettime()` dando una resolución en la medición en el orden de los nanosegundos, los tiempos de respuesta medidos se guardan en un archivo de texto, para ser graficados fuera de línea.

B. Tiempos de desalojo y tiempos de respuesta

El tiempo total de desalojos $P_{i,k}$ de la k -ésima instancia de la i -ésima tarea $J_{i,k}$, es la suma de los desalojos $P_{i,k}$ por parte de tareas de mayor prioridad, esto es:

$$P_{i,k} = \sum P_{g,k} \mid g, k \in \mathbb{Z}^+ \text{ con } i \neq g \quad (1)$$

El tiempo de respuesta $R_{i,k}$ de la k -ésima tarea $J_{i,k}$, es la diferencia entre el tiempo de finalizado $f_{i,k}$ y el tiempo de inicio $s_{i,k}$ independientemente del tiempo total de desalojos $P_{i,k}$ de la instancia (ver Figura 2).

$$R_{i,k} = C_{i,k} + O_{i,k} + P_{g,k} \quad (2)$$

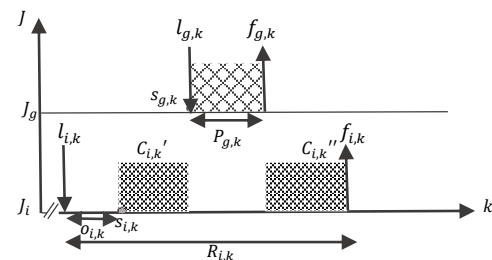


Fig. 2 Tiempo de Respuesta $R_{i,k}$ para n (número total de eventos) de un proceso [9]

Donde:

- $l_{i,k}$ es el tiempo de arribo,
- $O_{i,k}$ es el tiempo de operación del planificador
- $s_{i,k}$ es tiempo de inicio
- $f_{i,k}$ es tiempo de finalizado
- $P_{g,k}$ es una tarea que desaloja a $C_{i,k}$ para poder ejecutarse,
- i número de tarea y k número de evento

El tiempo de respuesta del sistema computacional depende del comportamiento de los tiempos de ejecución $c_{i,k}$ de proceso J_i ; sin embargo, este comportamiento fluctúa debido a varios factores como los expresados en F. Stappert and P. Altenbernd en [10] y G. Bernat et al en [11].

Los tiempos de respuesta o finalizado se pueden disminuir tomando en cuenta lo siguiente:

- Manejar prioridad para proceso
- Utilizar un Sistema Operativo de Tiempo Real
- Hacer uso de algún mecanismo de planificación, por ejemplo: FIFO, Round Robin, EDF o Adaptivo

C. Planificación mediante Políticas de Tiempo Real

Para priorizar el proceso de inversión de matrices se utilizan mecanismos de planificación de tiempo real, es decir se realiza

planificación a corto plazo, con prioridad estática. Se utiliza `sched_setscheduler()` de la biblioteca `<sched.h>`, ésta llamada establece tanto el mecanismo de planificación, como los parámetros para el hilo cuyo ID se especifica en el `pid`, éste mecanismo de planificación requiere permisos de superusuario, por lo tanto, el proceso de inversión de matrices debe ejecutarse con ese permiso desde terminal, con el comando `sudo`, en la Figura 3 se muestra la sintaxis de esta llamada al sistema. `Pid` es el ID de proceso al que se asignará la prioridad, puede establecerse con valor 0 para indicar que la prioridad se asignará al proceso actual. `policy` es la política de planificación, en éste trabajo son de utilidad las siguientes: `SCHED_FIFO` – planificador First-In First-Out, es un planificador de prioridad fija en el que un proceso de mayor prioridad se ejecuta hasta que desaloja el CPU voluntariamente o sea desalojado por un proceso de más alta prioridad que este. `SCHED_RR` – planificador Round Robin similar a `SCHED_FIFO`, a excepción de que los procesos que tienen el mismo nivel de prioridad se les asigna un `timeslice` cada 4 veces el periodo de reloj. Finalmente, `param` es un puntero a la estructura `sched_param` el cual contiene la prioridad que se requiere asignar al proceso. En éste argumento se especifica si se requiere la prioridad máxima o mínima del planificador utilizado, esto se hace mediante las funciones `sched_get_priority_min(policy)` y `sched_get_priority_max(policy)` las cuales retornan el valor entero de 1 y 99 respectivamente.

```
#include <sched.h>

int sched_setscheduler(
    pid_t pid,
    int policy,
    const struct sched_param *param);
```

Fig. 3 Sintaxis de la llamada al Sistema `sched_setscheduler()` [12]

De acuerdo a [13], `SCHED_FIFO` y `SCHED_RR` tienen prioridad sobre `SCHED_OTHER`, tienen baja latencia (para alta prioridad). `SCHED_RR` asigna un `timeslice` fijo para todos los procesos, la política `SCHED_FIFO` tiene prioridad fija y hasta que el proceso termine de ejecutarse puede desalojar el CPU.

Para el proceso de inversión de matrices, se utilizan estas políticas de planificación de tiempo real. Las pruebas se realizan estableciendo de forma experimental tres niveles de prioridad, en la tabla 1 se expresan esos niveles, asimismo se muestra la forma en que se asignan y el valor entero que retorna cada una.

TABLA 1 NIVELES DE PRIORIDAD PARA MECANISMOS DE PLANIFICACIÓN `SCHED_FIFO/SCHED_RR`

| Niveles de prioridad para Mecanismo de Planificación <code>SCHED_FIFO/SCHED_RR</code> | | |
|---|---|------------------|
| Nivel de prioridad | Asignación | Valor de retorno |
| Mínimo | <code>sched_get_priority_min(policy)</code> | 1 |
| Medio | <code>sched.sched_priority=50</code> | 50 |
| Máximo | <code>sched_get_priority_max(policy)</code> | 99 |

De la Figura 4 a la Figura 6, se muestra la implementación de los niveles de prioridad establecidos para las políticas de tiempo real.

```
#include <sched.h>
...
struct sched_param param;
param.sched_priority=
sched_get_priority_min(policy);
sched_setscheduler(0, policy, &param);
...
```

Fig. 4 Implementación de prioridad mínima en `sched_setscheduler()`

```
#include <sched.h>
...
struct sched_param param;
param.sched_priority=50;
sched_setscheduler(0, policy, &param);
...
```

Fig. 5 Implementación de prioridad media en `sched_setscheduler()`

```
#include <sched.h>
...
struct sched_param param;
param.sched_priority=
sched_get_priority_max(policy);

sched_setscheduler(0, policy, &param);
...
```

Fig. 6 Implementación de prioridad máxima en `sched_setscheduler()`

El valor de prioridad medio se decidió establecer en 50 de forma experimental, pues no hay una función predefinida en la biblioteca `<sched.h>` que determine un valor de prioridad medio. Al ejecutar el proceso de inversión de matrices bajo estos diferentes niveles y políticas de planificación de tiempo real, se observa el comportamiento de los tiempos de respuesta, de igual forma, esto permite saber la cantidad de varianza que hay entre los experimentos. Ésta métrica permite visualizar cómo es que el planificador por desalojo de prioridades, actúa y qué tan predecible puede ser el sistema, de acuerdo a la varianza de los tiempos de respuesta.

D. Análisis Estadístico de los Tiempos de Respuesta

Considerando R_k como una variable aleatoria, el primer momento de probabilidad o la media de una variable aleatoria se considera como una cantidad numérica alrededor de la cual los valores de la variable aleatoria tienden a agruparse [14] y la función con la cual se obtiene, se denota por la expresión (3):

$$\mu(R_k) = \frac{1}{k} \sum_{k=1}^k R_k \quad (3)$$

La expresión (3) proporciona el valor medio de todo el conjunto de mediciones, sin embargo, es de importancia observar el valor medio del tiempo de respuesta a medida que evoluciona el algoritmo hasta llegar a las 1000 iteraciones. Para ello es necesario tener la ecuación de la media aritmética de forma recursiva, con el objeto de conocer el comportamiento global y observar la dinámica de los tiempos de respuesta. Para observar la dinámica del experimento se aplica diferencias

finitas a (3) para obtener la ecuación de la media recursiva, como se representa en (4):

$$\mu(R_k) = \frac{(k-1)\mu(R_{k-1}) + R_k}{k} \quad (4)$$

El segundo momento de probabilidad o **varianza** del tiempo de respuesta R_k es una medida de la dispersión de sus valores alrededor de la media μ y se denota por σ^2 . La expresión matemática de la varianza se presenta en la ecuación (5) [14].

$$\sigma^2 = \frac{1}{n} \sum_{k=1}^n (R_k - \mu)^2 \quad (5)$$

De igual forma que el primer momento de probabilidad, la expresión (5) entrega el valor de varianza de los tiempos de respuesta del número total de iteraciones. No obstante, es necesario analizar la varianza de los tiempos de respuesta a medida que evoluciona el algoritmo, para observar de modo global la dinámica de todo el experimento y determinar si los tiempos de respuesta tienen estabilidad computacional, y a su vez los datos obtenidos proporcionan información para saber si el sistema operativo con extensión de Tiempo real puede ser

predecible o no. Para lograr realizar éste análisis, se necesita obtener la ecuación de la varianza recursiva, para ello se parte de la ecuación original (5), y se aplica diferencias finitas para obtener la varianza recursiva representada por la ecuación (6):

$$Var(R_k) = \frac{(k-1)Var(R_{k-1}) + (R_k - \mu(R_k))^2}{k} \quad (6)$$

III. RESULTADOS EXPERIMENTALES

A través del estudio de los tiempos de respuesta R_k de un proceso o tarea se puede analizar el impacto del uso de mecanismos de planificación en tiempo real. En este apartado, se presenta una selección de gráficas, correspondientes a los tiempos de respuesta, medidos al realizar inversión de matrices en baja carga computacional de dimensión 16×16 y alta carga computacional, 128×128 . De igual forma, se muestran las gráficas del primer momento de probabilidad superpuesta en los tiempos de respuesta, y segundo momento de probabilidad de cada experimento.

Tiempos de Respuesta con inversion de matrices de 16×16 en PREEMPT RT

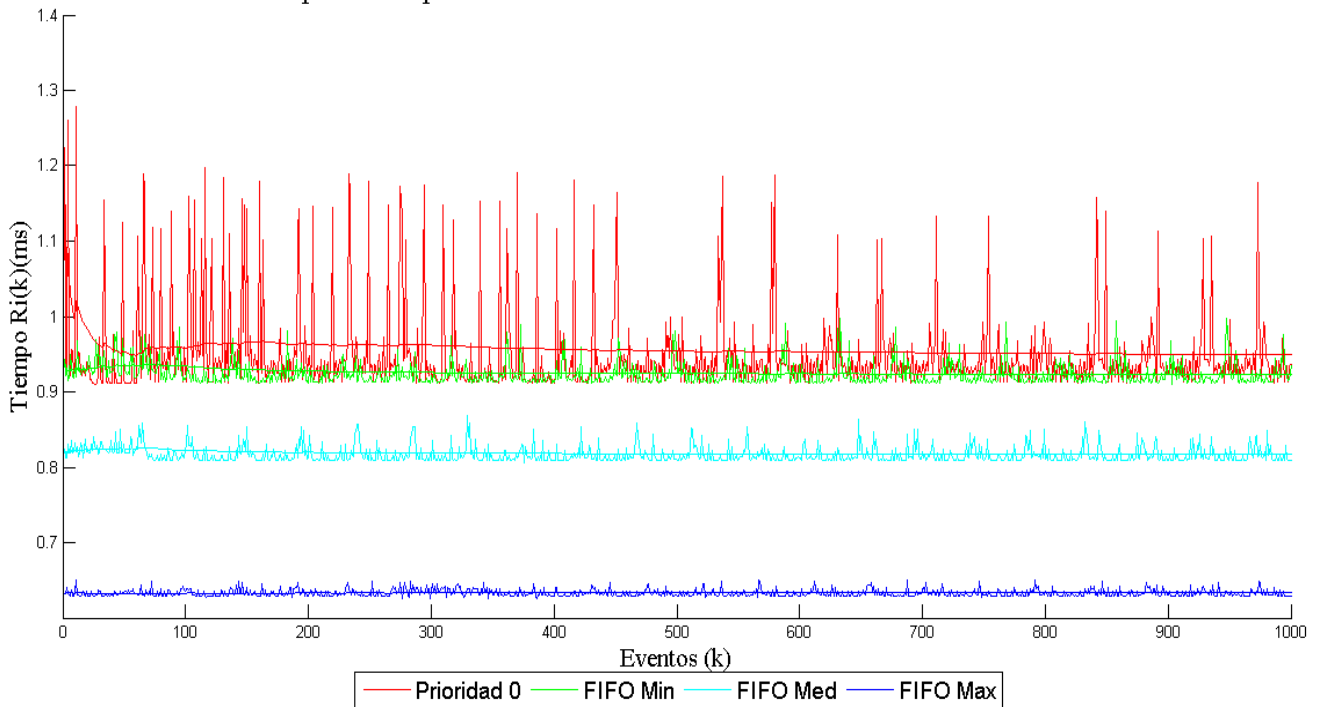


Fig. 7 Gráfica de Tiempos de Respuesta R_k en PREEMPT_RT, usando mecanismo de planificación FIFO con diferentes niveles de prioridad, para inversión de matriz de 16×16

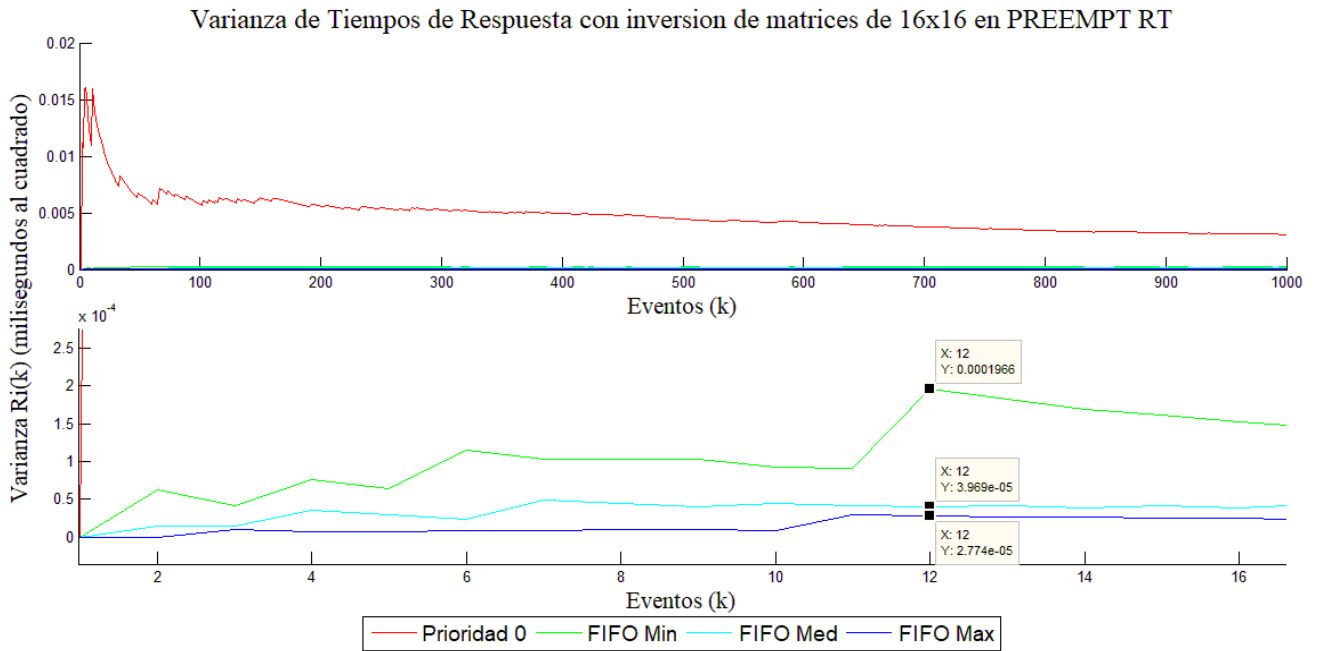


Fig. 8 Gráfica de Varianza de Tiempos de Respuesta R_k en PREEMPT_RT, usando mecanismo de planificación FIFO con diferentes niveles de prioridad, abajo zoom de la gráfica, para inversión de matriz de 16×16 .

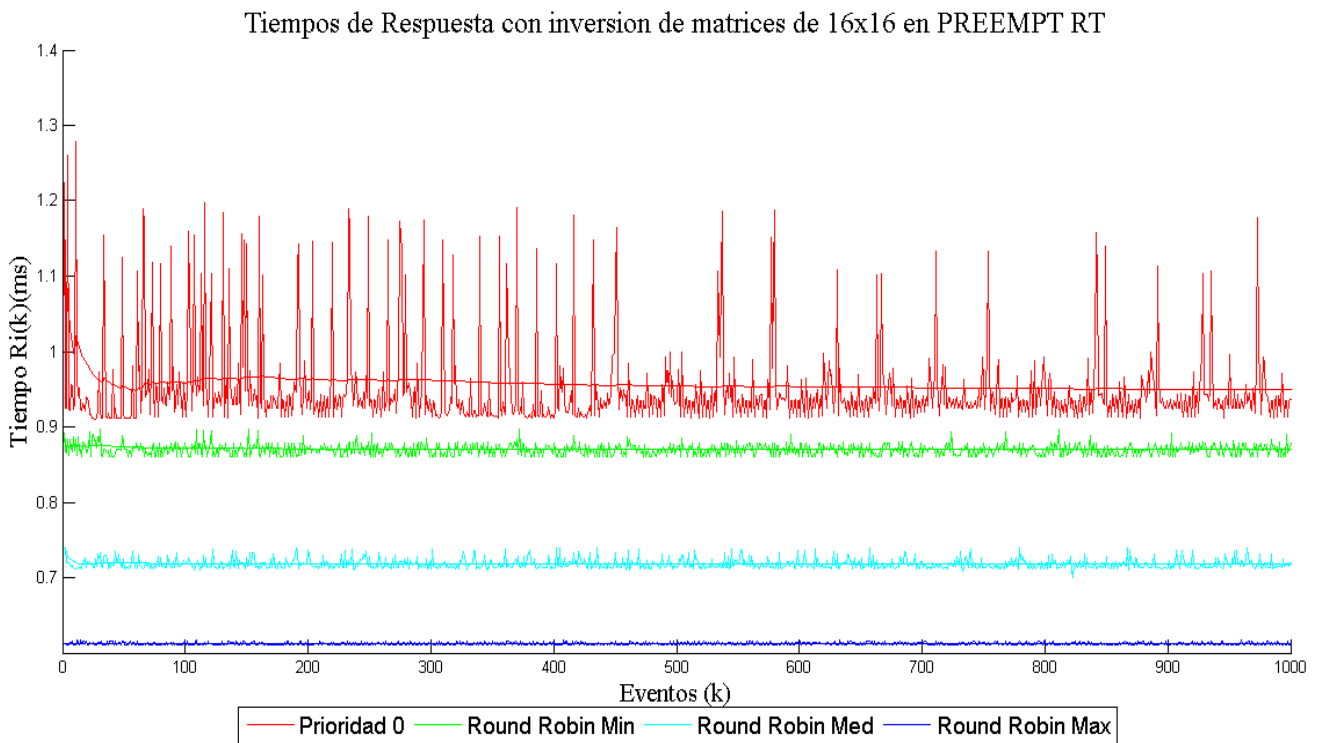


Fig. 9 Gráfica de Tiempos de Respuesta R_k en PREEMPT_RT, usando mecanismo de planificación Round Robin con diferentes niveles de prioridad, para inversión de matriz de 16×16 .

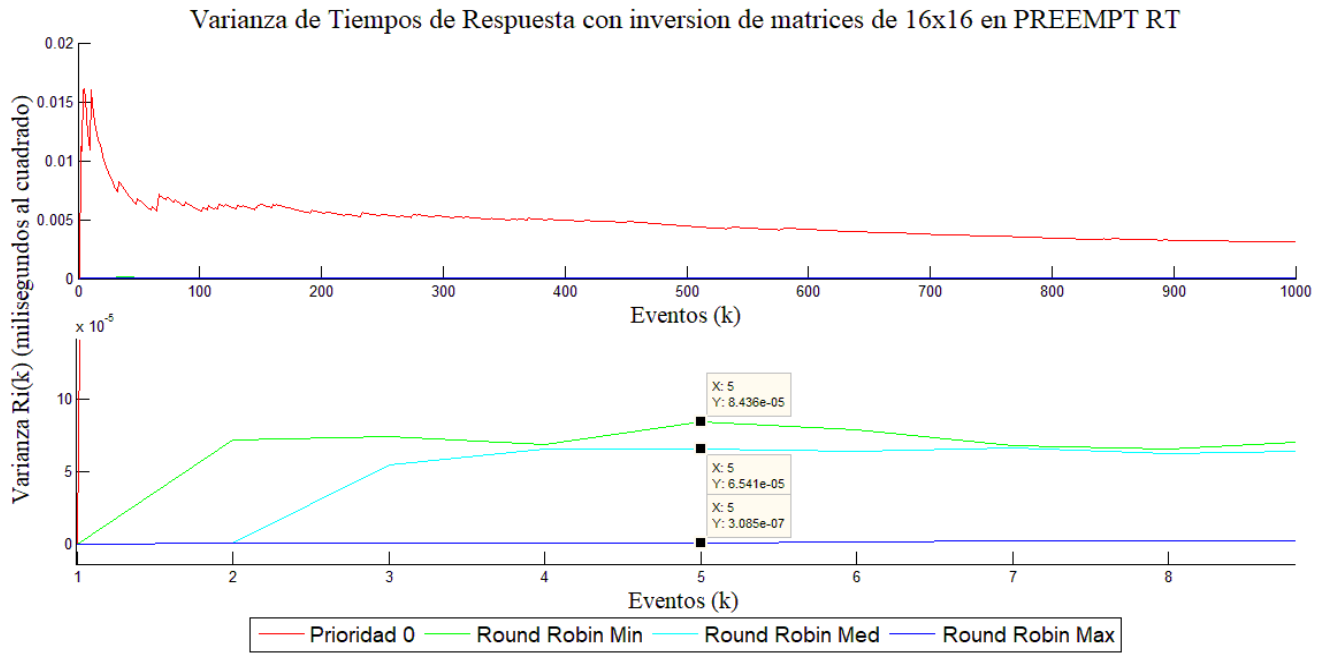


Fig. 10 Gráfica de Varianza de Tiempos de Respuesta R_k en PREEMPT_RT, usando mecanismo de planificación Round Robin con diferentes niveles de prioridad, abajo zoom de la gráfica, para inversión de matriz de 16×16 .

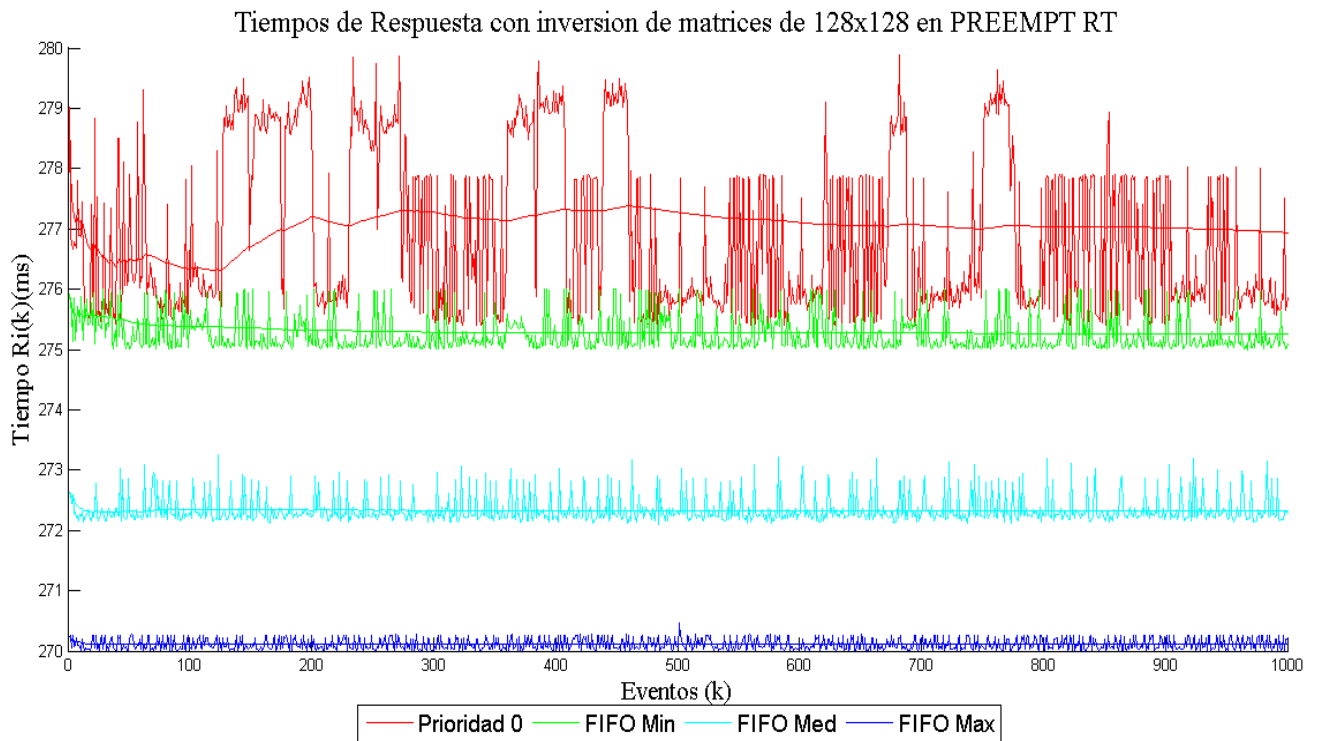


Fig. 11 Gráfica de Tiempos de Respuesta R_k en PREEMPT_RT, usando mecanismo de planificación FIFO con diferentes niveles de prioridad, para inversión de matriz de 128×128

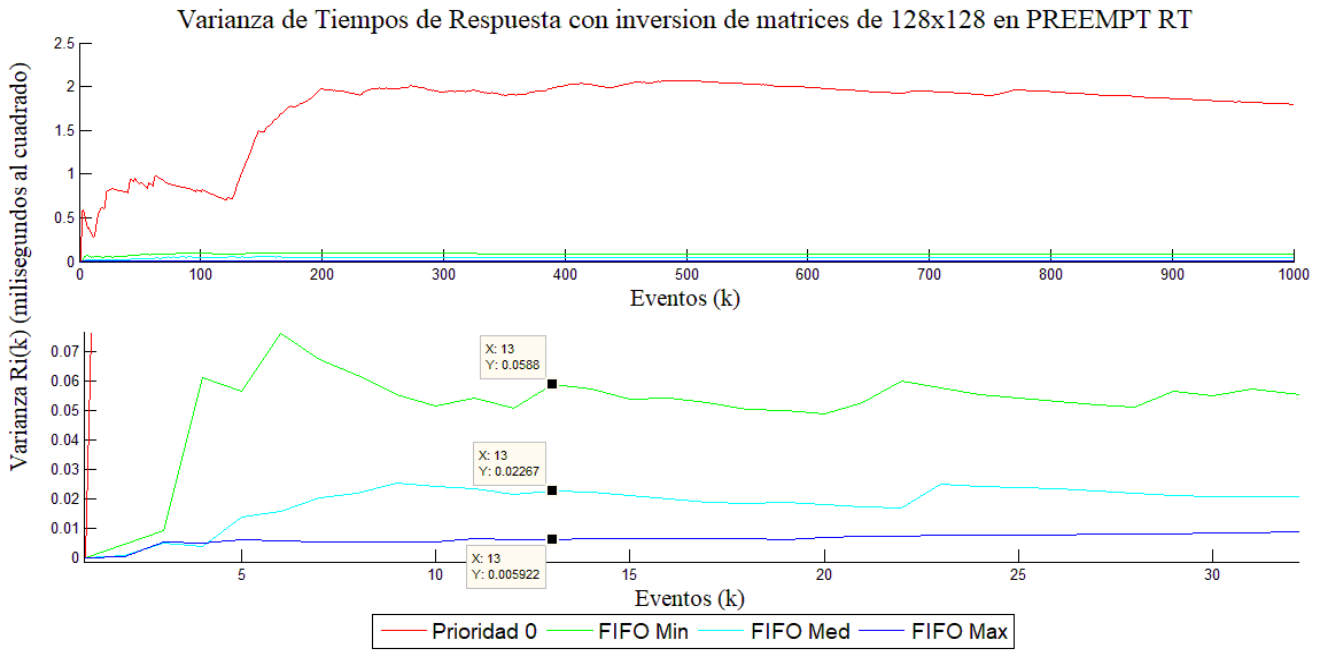


Fig. 12 Gráfica de Varianza de Tiempos de Respuesta R_k en PREEMPT_RT, usando mecanismo de planificación FIFO con diferentes niveles de prioridad, abajo zoom de la gráfica, para inversión de matriz de 128x128.

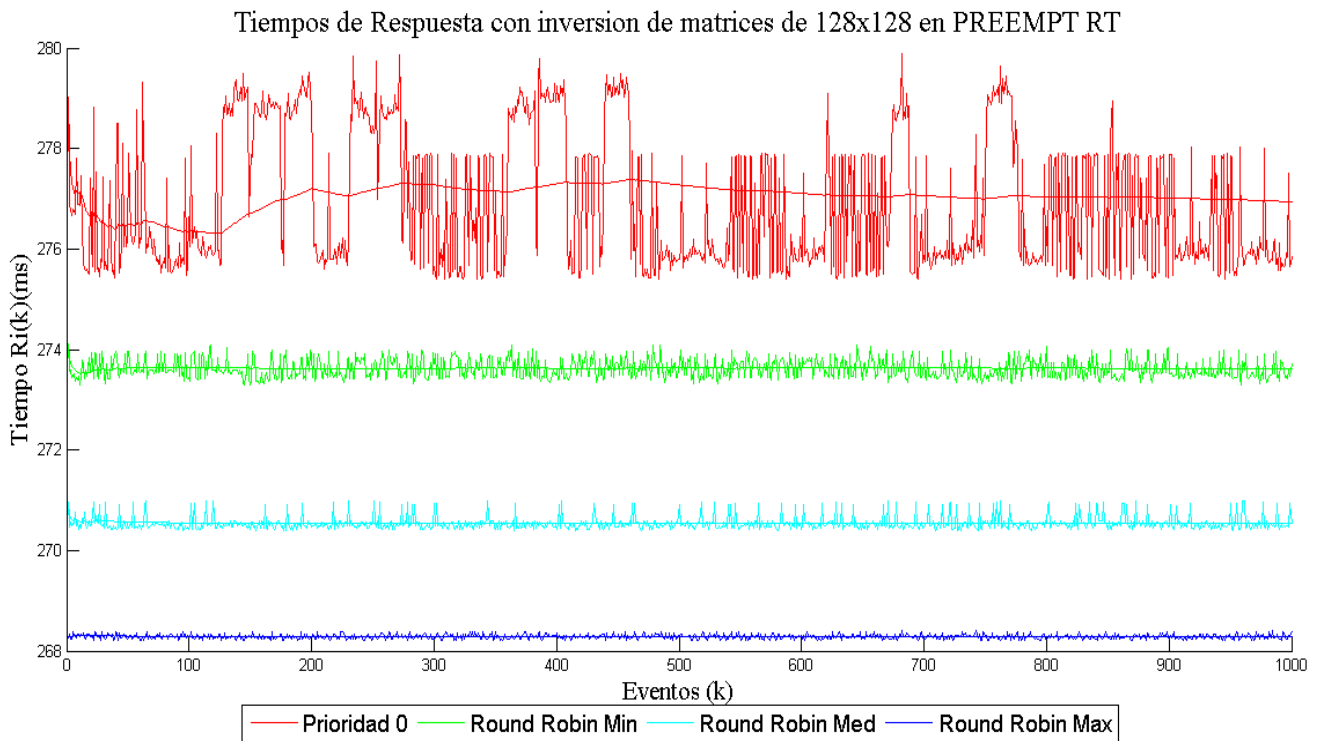


Fig. 13 Gráfica de Tiempos de Respuesta R_k en PREEMPT_RT, usando mecanismo de planificación Round Robin con diferentes niveles de prioridad, para inversión de matriz de 128x128

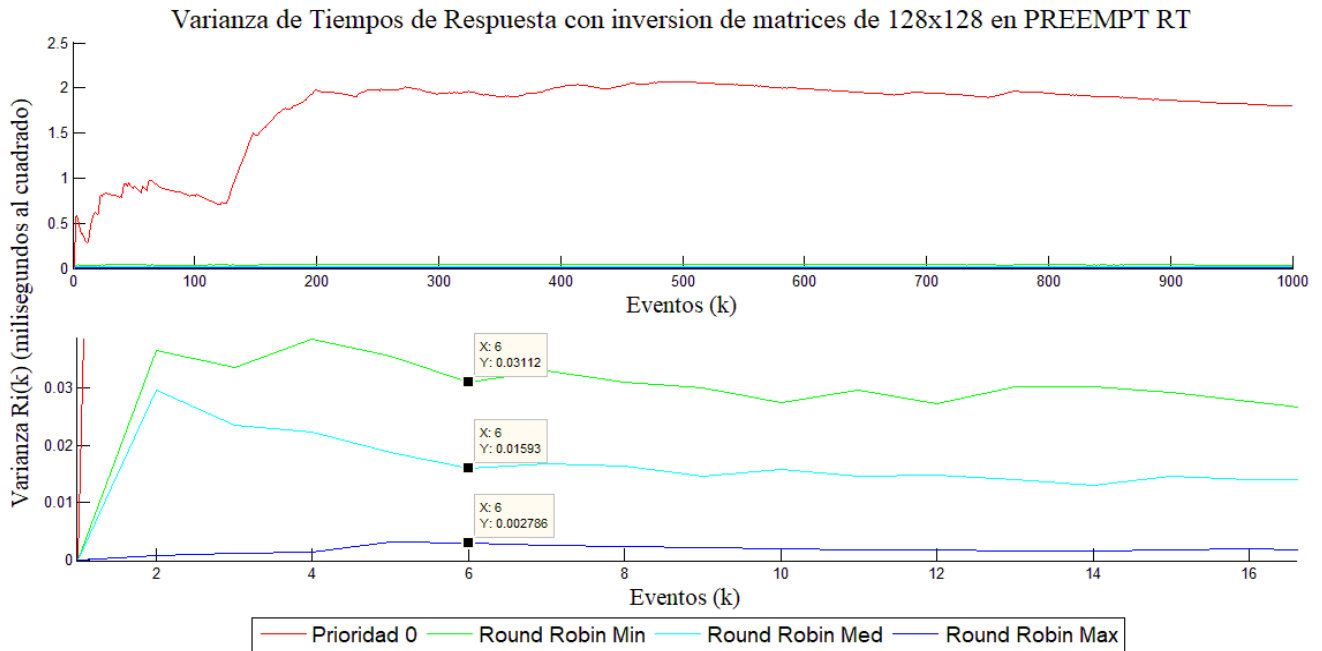


Fig. 14 Gráfica de Varianza de Tiempos de Respuesta en PREEMPT_RT, usando mecanismo de planificación Round Robin con diferentes niveles de prioridad, abajo zoom de la gráfica, para inversión de matriz de 128x128.

En las gráficas de la figura 15, se muestra una comparativa del comportamiento de los tiempos de respuesta con los mecanismos de planificación FIFO y Round Robin con prioridad máxima. En la segunda gráfica, se observa que el primer momento de probabilidad para el planificador FIFO es de 0.6398 milisegundos ligeramente mayor a Round Robin que es de 0.6126 milisegundos, obtenidos de los tiempos de respuesta mostrados en la primera gráfica. Para el segundo momento de probabilidad temporal mostrado en la tercera gráfica, para planificador FIFO se obtuvo una varianza de 21.8460×10^{-6} milisegundos al cuadrado mientras que para Round Robin fue de 3.9735×10^{-6} milisegundos al cuadrado. Se observa que la diferencia del primer momento de probabilidad entre cada planificador es de 0.0206 milisegundos, en el segundo momento de probabilidad se observa mayor variación con el planificador FIFO, siendo 17.8725×10^{-6} milisegundos al cuadrado, mayor a Round Robin.

En las gráficas de la figura 16 se muestra una comparativa del comportamiento de los tiempos de respuesta con los mecanismos de planificación FIFO y Round Robin con prioridad máxima. En la segunda gráfica, se observa que el primer momento de probabilidad temporal para el planificador

FIFO es de 270.1150 milisegundos ligeramente mayor a Round Robin que es de 268.2876 milisegundos, obtenidos de los tiempos de respuesta R_k mostrados en la primera gráfica. Para el segundo momento de probabilidad temporal mostrado en la tercera gráfica, para planificador FIFO se obtuvo una varianza de 0.0089 milisegundos al cuadrado mientras que para Round Robin fue de 0.0022 milisegundos al cuadrado. Se observa que la diferencia del primer momento de probabilidad entre cada planificador es de 1.8274 milisegundos, en el segundo momento de probabilidad se observa mayor variación con el planificador FIFO, siendo 0.0067 milisegundos al cuadrado, mayor a Round Robin.

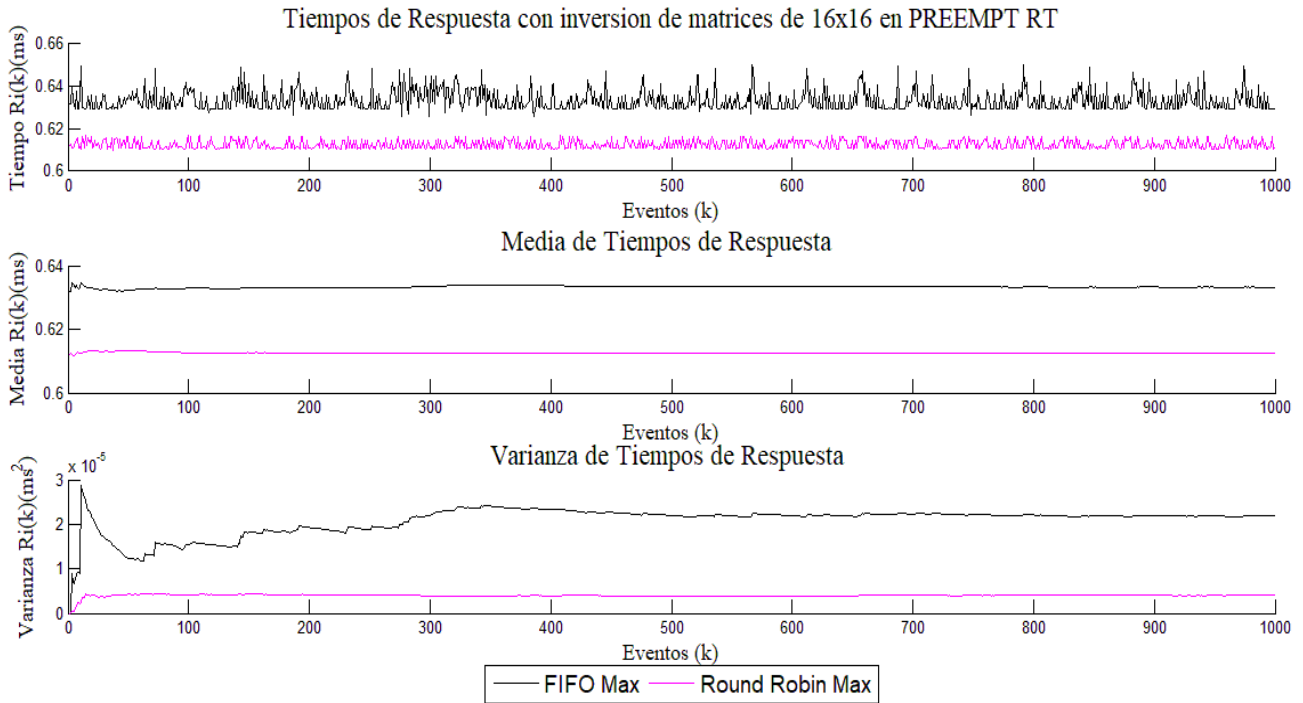


Fig. 15 Gráfica Comparativa de Tiempos de Respuesta R_i en PREEMPT_RT, usando mecanismo de planificación FIFO y Round Robin con prioridad máxima, para inversión de matriz de 16×16 .

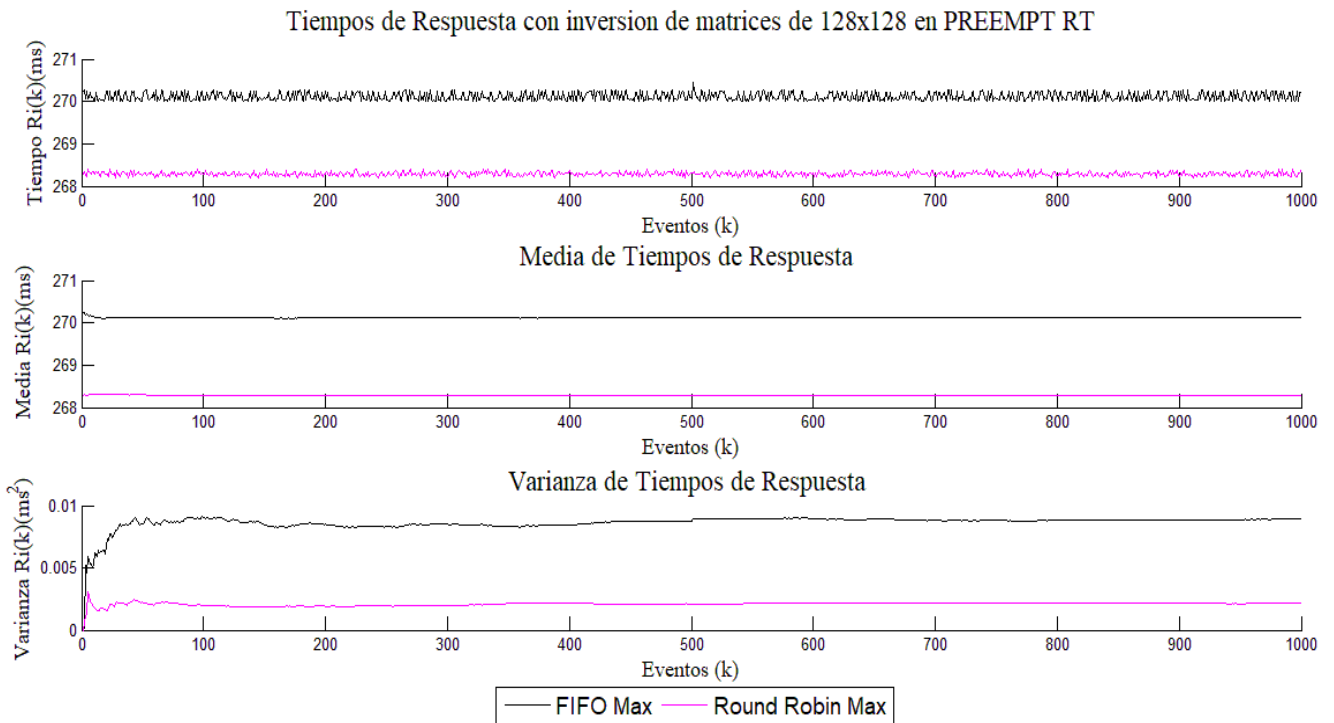


Fig. 16 Gráfica Comparativa de Tiempos de Respuesta en PREEMPT_RT, usando mecanismo de planificación FIFO y Round Robin con prioridad máxima, para inversión de matriz de 128×128 .

IV. CONCLUSIONES

Las gráficas obtenidas en cada experimento, muestran que a medida que aumenta la dimensión de matriz, aumenta el primer y segundo momentos de probabilidad, esto se atribuye a que el algoritmo de inversión de matrices tiene complejidad de tipo No Polinomial (NP), pues los tiempos van creciendo de forma exponencial. Al final de cada experimento, se realizó la comparativa de las políticas FIFO y Round Robin, utilizando el nivel máximo de prioridad, para visualizar cuánto disminuye entre cada una, el tiempo de respuesta y los primeros momentos de probabilidad. En la tabla 2 se resume tal comparativa, considerando el último valor del primer y segundo momentos de probabilidad, en cada experimento, utilizando política FIFO y Round Robin.

TABLA 2 COMPARATIVA DE MOMENTOS DE PROBABILIDAD DE CADA EXPERIMENTO, CON MECANISMO DE PLANIFICACIÓN FIFO Y ROUND ROBIN MÁXIMOS, OBTENIDOS EN PREEMPT_RT.

| Exp. | SCHED_FIFO | | SCHED_RR | |
|---------|------------|----------------------------|----------|----------------------------|
| | μ ms | σ^2 ms ² | μ ms | σ^2 ms ² |
| 16×16 | 0.6332 | 2.1846x10 ⁻⁵ | 0.6126 | 3.9735x10 ⁻⁶ |
| 128×128 | 270.1150 | 0.0089 | 268.2876 | 0.0022 |

En todos los experimentos, se observó que utilizando política Round Robin se obtienen tiempos de respuesta menores, y varianzas pequeñas a diferencia de la política FIFO. Con lo anterior, se concluye que para la ejecución de procesos que requieran atención especial y deban dar respuestas correctas en plazos definidos, es conveniente ejecutarlos utilizando mecanismo de planificación de tiempo real Round Robin con máxima prioridad, para que la ejecución de la tarea, nunca sea desalojada por otras que soliciten recurso del procesador.

REFERENCIAS

- [1] Medel Juárez J.J, Guevara López P. & Cruz Pérez D. (2007). *Temas Selectos de Sistemas en Tiempo Real*, Instituto Politécnico Nacional, México, Dirección de Publicaciones IPN.
- [2] System-on-chip technology comes of age | EE Times. (n.d.). [En línea]. Disponible en: http://www.eetimes.com/document.asp?doc_id=1279975 [Accesado: Junio 2017].
- [3] López D. Z. (2004). “Análisis de Sistemas Operativos en Tiempo Real”. Documento distribuido bajo términos de GNU Free Documentation License.
- [4] Pascual Morera J. & Pérez-Campanero J. (2002). *Conceptos de sistemas operativos* (Vol. 14). Univ Pontificia Comillas.
- [5] Rannou Fernando. (2015). *El Scheduler O(1) de Linux*. Departamento de Ingeniería Informática. Universidad de Santiago de Chile. [En línea] Disponible en: http://www.udesantiagovirtual.cl/moodle2/pluginfile.php?file=%2F101424%2Fmod_resource%2Fcontent%2F1%2Flinuxscheduler.pdf [Accesado: Octubre 2017]
- [6] Guevara López, P. (2004). “Filtrado Digital en Tiempo Real: Análisis Computacional para Estimación de Parámetros en Sistemas Estocásticos Lineales Estacionarios”. Tesis Doctoral. Centro de Investigación en Computación, IPN.
- [7] Valdez, J. S. (2015). “Medición, Caracterización y Reconstrucción de los Tiempos de Ejecución y Tiempos de Transporte para Sistemas de Telecontrol en Tiempo Real”. Tesis Doctoral. Sección de Estudios de Posgrado e Investigación. ESIME Culhuacan. IPN.
- [8] Cano, J.L. (2015). “Efecto del Overclocking sobre los tiempos de ejecución generados por inversión de matrices en una computadora embebida”. Tesis de Maestría. Sección de Estudios de Posgrado e Investigación. ESIME Culhuacan. IPN.
- [9] González D.L. (2018). “Análisis Experimental de los Tiempos de Respuesta en RT-Linux para una SBC”. Tesis de Maestría. Sección de Estudios de Posgrado e Investigación. ESIME Culhuacan. IPN.
- [10] Stappert F and Altenbernd P (2000). “Complete Worst-Case Execution Time Analysis of Straight-line Hard Real-Time Programs”. *Journal of Systems Architecture* V. 46, I. 4. Pp. 339–355. K. Elissa, “Title of paper if known,” unpublished.
- [11] Bernat G, Colin A, Petters SM (2003). “PW CET: A tool for probabilistic Worst-Case Execution Time Analysis of Real-Time Systems”. Technical Report YCS-2003-353. Department of Computer Science, University of York, Reino Unido, York. Pp. 1-18
- [12] QNX Neutrino Real-Time Operating System. *Library Reference*. (2007). Published under License by: QNX Software Systems International Corporation. Electronic edition. Kanata, Ontario, Canadá.
- [13] Tellería M. (2013). *Kernel: CPU y Memoria*. Diseño y Evaluación de Configuraciones (DEC). Facultad de Ciencias, Universidad de Cantabria.
- [14] Kreyszig, E. (2005). *Advanced Engineering Mathematics*. Wiley.

Implementación en FPGA de un sistema de identificación de parámetros de modelos MA utilizando un algoritmo Levinson-Durbin modificado

Durán Santos, MARTÍN¹(Autor)Palomino Merino,
AMPARO D.²(Autor)

Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Electrónica
Puebla, Puebla, México

¹martin.duran.mecatronica@gmail.com

²palomino@ece.buap.mx

Etcheverry, GIBRAN(Autor)

Universidad de las Américas Puebla
Escuela de Ingeniería

San Andrés Cholula, Puebla, México
gibran.etcveverry@udlap.mx

Resumen—Este trabajo presenta la implementación en FPGA con VHDL en Vivado 2016.2 de un sistema de identificación de parámetros de un modelo de Media Móvil (MA) de segundo orden, utilizando un algoritmo de Levinson-Durbin modificado. En la introducción se explica el algoritmo Levinson-Durbin. Posteriormente se da una explicación del algoritmo propuesto. En la implementación se describen los bloques elementales, así como el diseño final. Los resultados y conclusiones se presentan al final de este documento.

Palabras clave—Modelos de Media Móvil, Algoritmo Levinson-Durbin, Identificación de Parámetros, Implementación en FPGA

I. INTRODUCCIÓN

Dada una señal estacionaria en tiempo discreto $y(n)$, se puede realizar una estimación $\hat{y}(n)$ utilizando un modelo AR, MA o ARMA.

El Algoritmo Levinson-Durbin puede utilizarse para identificar parámetros de modelos autorregresivos de la forma:

$$\hat{y}(n) = \sum_{k=1}^N a_{k,N} y(n-k) \quad (1)$$

donde:

$\hat{y}(n)$: Señal estimada,

$a_{k,N}$: Parámetros del modelo,

$y(n-k)$: Señal estacionaria original con k retardos.

Considerando el valor esperado del error (J), es decir $J = E\{y(n) - \hat{y}(n)\}$, y derivando con respecto a los parámetros del modelo, se obtiene el siguiente sistema de ecuaciones:

$$\begin{bmatrix} R[0] & \cdots & R[N-1] \\ \vdots & \ddots & \vdots \\ R[N-1] & \cdots & R[0] \end{bmatrix} \begin{bmatrix} a_{1,N} \\ \vdots \\ a_{N,N} \end{bmatrix} = \begin{bmatrix} R[1] \\ \vdots \\ R[N] \end{bmatrix} \quad (2)$$

Donde $R = \begin{bmatrix} R[0] & \cdots & R[N-1] \\ \vdots & \ddots & \vdots \\ R[N-1] & \cdots & R[0] \end{bmatrix}$ es la matriz de autocorrelación, y es Toeplitz [1].

Los parámetros del modelo pueden ser obtenidos a partir de las ecuaciones recursivas [1]:

$$a_{k,N} = a_{k,N-1} - K_N a_{N-k,N-1} \quad (3)$$

$$a_{N,N} = K_N \quad (4)$$

$$K_N = \frac{1}{P_{N-1}} (R[N] - \sum_{k=1}^{N-1} a_{k,N-1} R(N-k)) \quad (5)$$

$$P_N = (1 - K_N^2) P_{N-1} \quad (6)$$

II. METODOLOGÍA

Se propone un modelo de media móvil de la forma:

$$\hat{y}(n) = \sum_{k=1}^N b_{k,N} x(n-k) \quad (7)$$

donde:

$\hat{y}(n)$: Señal estimada,

$b_{k,N}$: Parámetros del modelo de media móvil,

$x(n-k)$: Señal externa $x(n)$ con k retardos.

Dado que $x(n)$ es una entrada externa, y queremos representar a la señal original como un modelo MA, entonces el sistema de ecuaciones para obtener los parámetros del modelo es:

$$\begin{bmatrix} R_{1,1} & R_{1,2} & \cdots & R_{1,N} \\ R_{1,2} & R_{2,2} & \cdots & R_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ R_{1,N} & R_{2,N} & \cdots & R_{N,N} \end{bmatrix} \begin{bmatrix} b_{1,N} \\ b_{2,N} \\ \vdots \\ b_{N,N} \end{bmatrix} = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_N \end{bmatrix} \quad (8)$$

donde:

$$R_{p,q} = E\{x(n-p), x(n-q)\} \quad (9)$$

$$S_r = E\{y(n), x(n-r)\} \quad (10)$$

y la matriz de autocorrelación generada no es Toeplitz.

Utilizando la Ortogonalización Gram-Schmidt:

$$xN(n-N) = x(n-N) - xN(n-1)k_{1,N} - \dots - xN(n-N)k_{N-1,N} \quad (11)$$

donde los coeficientes de proyección son:

$$k_{j,k} = \frac{xN(n-j) \cdot x(n-k)}{xN(n-j) \cdot xN(n-j)} \quad (12)$$

tal que:

$$xN(n-1) \perp xN(n-2) \perp \dots \perp xN(n-N) \quad (13)$$

Por tanto, obtenemos el siguiente sistema de ecuaciones:

$$\begin{bmatrix} RN_{1,1} & 0 & \dots & 0 \\ 0 & RN_{2,2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & RN_{N,N} \end{bmatrix} \begin{bmatrix} b_{1,N} \\ b_{2,N} \\ \vdots \\ b_{N,N} \end{bmatrix} = \begin{bmatrix} SN_1 \\ SN_2 \\ \vdots \\ SN_N \end{bmatrix} \quad (14)$$

Donde

$$RN_{N,N} = E\{xN(n-N), xN(n-N)\} \quad (15)$$

$$SN_N = E\{y(n), xN(n-N)\} \quad (16)$$

El valor de los parámetros del sistema queda determinado por:

$$b_{N,N} = \frac{SN_N}{RN_{N,N}} \quad (17)$$

$$b_{j,k} = b_{j,k-1} - b_{k,N}k_{j,k} \quad (18)$$

III. IMPLEMENTACIÓN

A. Formato

El formato utilizado en este trabajo se basa en palabras de 20 bits (de 0 a 19) de punto fijo, donde el bit más significativo (19) es el bit de signo, los bits 10 a 18 representan la parte entera, y los bits 0 a 9 representan la parte decimal. De esta manera, el número decimal que se quiere representar en este formato debe multiplicarse por 1024 [2].

El rango de este formato es desde -2^{19} hasta $2^{19} - 1$.

B. Estructura General

El lenguaje de programación es VHDL [3].

La implementación del sistema de identificación de parámetros de modelos de Media Móvil de orden 2 se basa en 4 diferentes bloques elementales:

- 1) Retardo
- 2) Unidad MAC
- 3) División
- 4) Proyección

Se requieren dos bloques de retardo, de los cuales obtendremos las señales $x(k-1)$ y $x(k-2)$.

Para las señales de autocorrelación $R_{1,1}$, $RN_{1,1}$ y $RN_{2,2}$ se utilizan 2 unidades MAC (Multiplica y Acumula), ya que $R_{1,1} = RN_{1,1}$. Para las señales de correlación cruzada $R_{1,2}$, S_1 , SN_1 y SN_2 se utilizan 3 unidades MAC, ya que $S_1 = SN_1$.

Para los coeficientes $b_{1,1}$, $k_{1,2}$ y $b_{2,2}$ se requieren 3 bloques de división.

Finalmente, para las señales $b_{1,2}$ y $xN(n-2)$ se utilizan 2 bloques de proyección, que realizan la operación $a - bk$, donde a y b son señales de entrada de 20 bits cada una, y k es una constante también de 20 bits.

C. Implementación de bloques elementales

1) Retardo

Funcionamiento: El bloque *retardo* mostrado en la Fig. 1 recibe un dato de entrada y lo guarda por un instante de tiempo, lo que provoca un retardo de un ciclo de reloj.

Entradas:

clk: Señal de reloj,

x1: Señal de entrada de 20 bits $x(k)$.

Salidas:

r1: Señal de salida de 20 bits $x(k-1)$.

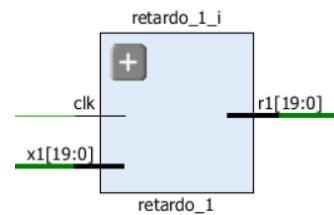


Fig. 1. Bloque retardo.

2) Unidad MAC (Multiplica y Acumula)

Funcionamiento: El bloque *Unidad MAC*, el cual se muestra en la Fig. 2, realiza la operación de correlación o autocorrelación, dependiendo las entradas. Esta unidad multiplica los datos de entrada a y b y el resultado lo guarda y lo muestra en la salida. Posteriormente vuelve a multiplicar las entradas y el resultado se suma al valor anterior, y así sucesivamente.

Entradas:

clk: Señal de reloj,

entrada_a: Señal de entrada 1 de 20 bits,

entrada_b: Señal de entrada 2 de 20 bits.

Salidas:

salida_MAC: Señal de salida de 20 bits.

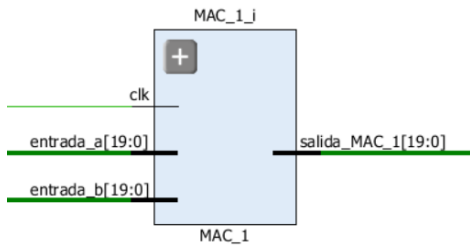


Fig. 2. Bloque Unidad MAC.

3) División

Funcionamiento: El bloque *división* mostrado en la Fig. 3 realiza la operación de división a/b .

Entradas:

- clk: Señal de reloj,
- a_div: Señal de entrada de 20 bits (dividendo),
- b_div: Señal de entrada de 20 bits (divisor),
- reset: Pone a sus condiciones iniciales y reinicializa el proceso.

Salidas:

- sal_div: Señal de salida de 20 bits,
- disparo: Pone su salida en 1 cuando termina de realizar la operación (se utiliza como activador del siguiente proceso).

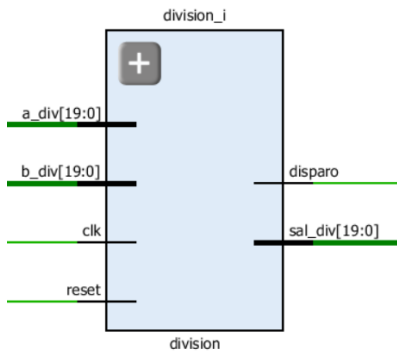


Fig. 3. Bloque division.

4) Proyección

Funcionamiento: El bloque *proyección* mostrado en la Fig. 4 realiza la proyección de vectores. Realiza la operación $sal = a - bk$.

Entradas:

- clk: Señal de reloj,
- a_div: Señal de entrada de 20 bits (a),
- b_div: Señal de entrada de 20 bits (b),
- constante: Señal de entrada de 20 bits (k).

Salidas:

- sal_proy: Señal de salida de 20 bits.

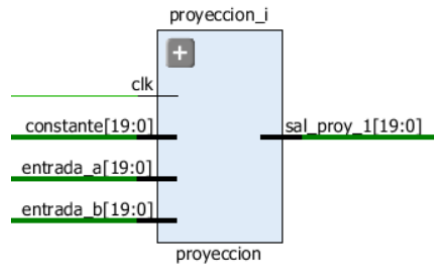


Fig. 4. Bloque proyeccion.

D. Implementación del sistema de identificación

Como se observa en la Fig. 5, la primera etapa es la lectura de los datos. Esta se realiza a través de las librerías `ieee.std_logic_textio.all` y `std.textio.all`, las cuales tienen la función de guardar los datos de entrada en formato .txt en una memoria ROM (señales $x(k)$ y $y(k)$, representadas por `dataout` y `dataout2`, respetivamente).

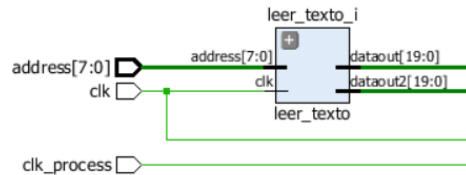


Fig. 5. Lectura de los datos, `dataout` representa la señal de entrada $x(k)$, y `dataout2` representa la señal de salida $y(k)$.

Como se observa en la Fig. 6, la señal de entrada $x(k)$ pasa por un retardo (`retardo_1_i`) para obtener $x(k - 1)$. A esta señal se le aplica el valor esperado $E\{x(k - 1), x(k - 1)\}$, el cual se obtiene utilizando el bloque `MAC_1_i`, que multiplica las entradas y acumula el resultado a cada flanco de subida del reloj. La señal obtenida será llamada *R11* y representa la autocorrelación del primer retardo de la señal de entrada $x(k)$.

Como el sistema es de segundo orden, se requieren dos retardos de la señal de entrada. El bloque `MAC_3_i` obtiene la señal $R12 = E\{x(k - 1), x(k - 2)\}$, la cual es la autocorrelación entre el primer retardo $x(k - 1)$ y el segundo retardo $x(k - 2)$, señal obtenida a partir del bloque `retardo_2_i`.

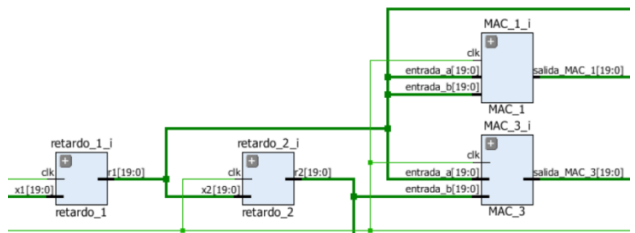


Fig. 6. Bloques de retardo (`retardo_1_i`, `retardo_2_i`), y de autocorrelación (`MAC_1_i`, `MAC_3_i`).

Como se puede observar en la Fig. 7, el bloque MAC_2_i obtiene la correlación cruzada entre el primer retardo de la señal de entrada $x(k-1)$ con la señal de salida $y(k)$, es decir $E\{y(k), x(k-1)\}$. La señal de salida de este bloque será llamada $S1$.

Posteriormente, se utiliza la ortogonalización Gram-Schmidt para obtener el coeficiente de predicción $k12 = \frac{R12}{R11}$. El bloque proyeccion_i obtiene el nuevo vector $xN(k-2) = x(k-2) - xN(k-1)k12$, donde $xN(k-1) = x(k-1)$.

Se requieren obtener las señales $R22N$ y $S2N$, las cuales se calculan utilizando los bloques MAC_R22N_i y MAC_S2N_i, respectivamente.

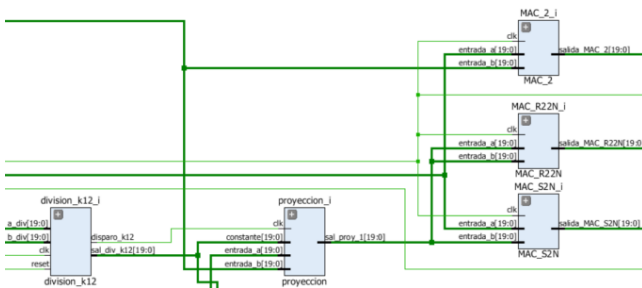


Fig. 7. Bloques de división (division_k12_i), de proyección (proyeccion_i) y de autocorrelación (MAC_2_i, MAC_R22N_i, MAC_S2N_i).

Finalmente, como se puede ver en la Fig. 8, se calculan los parámetros del sistema a través de 2 bloques de división, y 1 bloque de proyección.

El primer parámetro del modelo MA de primer orden ($b11$) se obtiene como: $b11 = k1 = \frac{S1}{R11}$.

El segundo parámetro del modelo MA de segundo orden ($b22$), se obtiene utilizando el bloque división_k2_i, el cual realiza la operación $b22 = k22 = \frac{S2N}{R22N}$.

El primer parámetro del modelo MA de segundo orden ($b12$) se obtiene al utilizar el bloque proyección_b21_i, el cual realiza la operación $b12 = b11 - b22k12$.

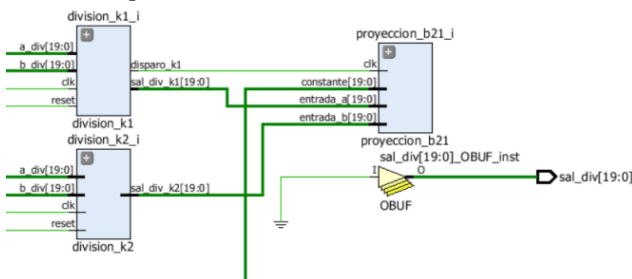


Fig. 8. Bloques de división (division_k1_i, division_k2_i), y de proyección (proyeccion_b21_i).

IV. ANÁLISIS DE RESULTADOS

A partir de la implementación descrita en el apartado anterior, se realizaron dos pruebas diferentes.

Para la primera prueba, las señales de entrada y salida son:

$$x(k) = \sin(k) \quad (19)$$

$$y(k) = -0.3333x(k-1) + 0.7500x(k-2) \quad (20)$$

La identificación de los parámetros puede observarse en la Fig. 9:

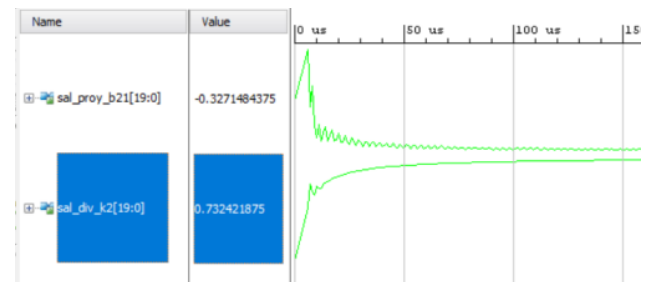


Fig. 9. Identificación de los parámetros $b_{1,2}$ para la primera prueba.

Al inicio de la identificación existe un comportamiento abrupto. Esto se debe a la poca información que se tiene, y el sistema no puede realizar la identificación correctamente con pocos datos. Sin embargo, conforme transcurre el tiempo más datos van siendo procesados, y el sistema tiene un comportamiento convergente hacia un valor. El valor para $b_{2,2}$ es 0.732421875, donde existe un error de 0.0176. El otro parámetro ($b_{1,2}$) tiene un valor de -0.3271484375, con error de 0.0062.

Este error se debe principalmente al tamaño de la palabra que se utiliza (20 bits), así como a los truncamientos realizados en la multiplicación en los bloques MAC, y en las divisiones.

Para la segunda prueba, la señal de entrada está dada por la ecuación (19), y la salida está dada por:

$$y(k) = 0.5000x(k-1) - 0.8800x(k-2) \quad (21)$$

La identificación de los parámetros puede observarse en la Fig. 10:

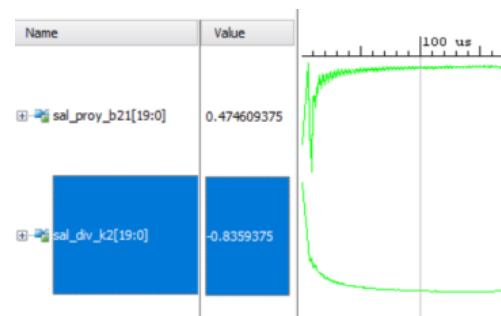


Fig. 10. Identificación de los parámetros $b_{1,2}$ para la segunda prueba.

El valor para $b_{2,2}$ es -0.8359375 , donde existe un error de 0.0441 . El otro parámetro ($b_{1,2}$) tiene un valor de 0.474609375 , con error de 0.0254 .

De la misma manera, estos errores se deben al tamaño del formato utilizado, y al truncamiento que se produce en los bloques división, Unidad MAC y proyección.

V. CONCLUSIONES

Se realizó la implementación de un sistema de identificación de parámetros de un modelo de Media Móvil de orden 2.

La implementación en FPGA del sistema de identificación descrito en este trabajo puede ser dividida de acuerdo al orden del modelo. Es decir, como primera etapa se implementó un sistema de identificación de parámetros de un modelo MA de orden 1, y el parámetro obtenido es identificado por $b_{1,1}$. Posteriormente, los coeficientes $b_{2,1}$ y $b_{2,2}$ del modelo de media móvil de orden 2 se calculan a partir del parámetro anterior ($b_{1,1}$) y del coeficiente de predicción $k_{1,2}$.

El algoritmo realizado tiene la flexibilidad de aumentar el orden del sistema. Es decir, para la implementación de un sistema de identificación de modelos MA de orden 3, se utilizan los parámetros obtenidos del modelo de orden 2 y se realiza la ortogonalización Gram-Schmidt para obtener un sistema de ecuaciones de orden 3, y se resuelve con la metodología aquí propuesta.

A partir de los resultados obtenidos, es importante considerar si el margen de error es aceptable dependiendo de la aplicación. Si se requiere una tolerancia menor, entonces es recomendable aumentar el tamaño de la palabra, por ejemplo a 32 bits, para aumentar la precisión de los resultados.

Este trabajo puede ser implementado con cómputo hereditario (Etcheverry G. and Reyes García C., INAOE, Department of Computation Science 2009), y también utilizar filtros Autorregresivos (Duran Santos M. et al., CIINDET-2018) lo cual permitiría el cálculo de parámetros de señales no lineales y no estacionarias en tiempo real; por ejemplo, las obtención de las características de señales como el sonar de un murciélago o el llanto de un recién nacido.

REFERENCIAS

- [1] Papoulis A. "Levinson's Algorithm, Wold's Decomposition, and Spectral Estimation". Society for Industrial and Applied Mathematics. Vol. 27, No. 3, September 1985.
- [2] F. Mora. "Implementación de la Transformada Discreta de Fourier IFFT/FFT en un sistema embebido para automatizar un sistema de comunicación OFDMA". Tesis presentada como requisito para obtener

el título en Maestría en Ciencias de la Electrónica. Puebla, México. Diciembre 2015.

- [3] F. Pardo. "Lenguaje para síntesis y modelado de circuitos". Alfaomega. Tercera edición. México, D.F. 2011.

MODELO CINEMATICO DIRECTO BASADO EN CUATERNIONES DUALES DEL MANIPULADOR ROBOTICO ABB IRB6400

Gerardo Hernández Valencia, Iván Altamirano
Olguín, Ariel Fuerte Hernández, Juan Pablo Campos
López, Omar Ismael López Suárez
División de Ingeniería Mecatrónica
UPVM
Tultitlán, México

Abstract—En este trabajo se presenta el modelo cinemático directo basado en cuaterniones duales del robot manipulador industrial IRB 6400 de la marca ABB, el cual consiste en un robot de configuración antropomórfica de seis grados de libertad con articulaciones rotacionales. Los sistemas de coordenadas se asignaron de acuerdo a la convención de Denavit-Hartenberg y se obtuvieron los cuaterniones duales que relacionan al sistema de coordenadas anterior con el posterior, estos se multiplicaron de forma secuencial para obtener las ecuaciones cinemáticas del manipulador. El modelo obtenido se programó en el software Matlab como una función simbólica, la cual permite obtener la posición y orientación del extremo del robot a partir de las coordenadas angulares de cada una de las articulaciones del manipulador. Finalmente, se establecieron los valores de HOME para el robot y se realizó una transformación de cuaternión dual a matriz de transformación homogénea para validar los resultados obtenidos, los cuales demuestran que los cuaterniones duales permiten representar la posición y orientación de un sistema de coordenadas respecto a otro de una manera más eficiente y robusta que las matrices de transformación homogénea, siendo esta última, la herramienta más difundida en la literatura especializada sobre robótica.

Keywords— *Cuaternión dual, Matriz de Transformación Homogénea, Robótica, Cinemática.*

I. INTRODUCCIÓN

Para que un robot pueda realizar las tareas de manipulación que le son programadas es necesario contar con una serie de herramientas matemáticas que permitan determinar la posición y orientación de su extremo final con respecto a un sistema de coordenadas de referencia. La herramienta con mayor difusión en robótica para representar de manera conjunta, posición y orientación del efector final del robot son las matrices de transformación homogénea [1]. Sin embargo, en la práctica resultan ineficientes, debido a que su implementación computacional en comparación con otro tipo de representaciones, requiere de un mayor número de operaciones, además de que arrojan inconsistencias numéricas debido a la representación altamente redundante de la rotación (la cual requiere de nueve elementos para mostrar la orientación en un

espacio de tres dimensiones), y a la condición de que en todo momento se tiene que preservar la orto normalidad de la matriz [2]. Los cuaterniones son, por otra parte, una herramienta matemática de gran versatilidad computacional para trabajar con giros y orientaciones, ya que permiten representar las rotaciones mediante solo cuatro elementos, frente a los nueve utilizados en las matrices de rotación [3], sin embargo, presentan como desventaja el no expresar desplazamientos respecto a un sistema de referencia, no obstante, su relación con los números duales permiten obtener una extensión a través de dos cuaterniones, donde uno de ellos contiene la información sobre la rotación y el otro contiene la información sobre la traslación de manera que se obtiene una representación matemática más compacta, robusta y eficiente para las ecuaciones cinemáticas del robot [4].

II. MARCO TEÓRICO

Los cuaterniones se expresan como una extensión de los numero complejos por medio de las expresiones (1) y (2).

$$q_0 = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \quad (1)$$

$$p_0 = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} \quad (2)$$

Definiéndose las operaciones básicas de suma y producto entre ellos como:

$$q_0 + p_0 = \begin{bmatrix} q_1 + p_1 \\ q_2 + p_2 \\ q_3 + p_3 \\ q_4 + p_4 \end{bmatrix} \quad (3)$$

$$q_0 * p_0 = \begin{bmatrix} q_1 * p_1 - q_2 * p_2 - q_3 * p_3 - q_4 * p_4 \\ q_1 * p_2 - q_2 * p_1 - q_3 * p_4 - q_4 * p_3 \\ q_1 * p_3 - q_2 * p_4 - q_3 * p_1 - q_4 * p_2 \\ q_1 * p_4 - q_2 * p_3 - q_3 * p_2 - q_4 * p_1 \end{bmatrix} \quad (4)$$

Los cuaterniones duales fueron desarrollados por Clifford en 1873 como una extensión de los cuaterniones, con la finalidad de representar de manera conjunta la rotación y traslación de un cuerpo en el espacio [4]. Cada cuaternión dual se compone de ocho elementos o dos cuaterniones de la forma:

$$Dq_0 = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \\ q_7 \\ q_8 \end{bmatrix} \quad (5)$$

$$Dp_0 = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_7 \\ p_8 \end{bmatrix} \quad (6)$$

Y el producto entre ellos se realiza como se presenta a continuación:

$$Dq_0 * Dp_0 = \begin{bmatrix} q_1 * p_1 - q_2 * p_2 - q_3 * p_3 - q_4 * p_4 \\ q_2 * p_1 + q_1 * p_2 - q_4 * p_3 + q_3 * p_4 \\ q_3 * p_1 + q_4 * p_2 + q_1 * p_3 - q_2 * p_4 \\ q_4 * p_1 - q_3 * p_2 + q_2 * p_3 + q_1 * p_4 \\ q_5 * p_1 - q_6 * p_2 - q_7 * p_3 - q_8 * p_4 + q_1 * p_5 - q_2 * p_6 - q_3 * p_7 - q_4 * p_8 \\ q_6 * p_1 - q_5 * p_2 - q_8 * p_3 + q_7 * p_4 + q_2 * p_5 - q_1 * p_6 - q_4 * p_7 + q_3 * p_8 \\ q_7 * p_1 + q_8 * p_2 + q_5 * p_3 - q_6 * p_4 + q_3 * p_5 + q_4 * p_6 - q_1 * p_7 - q_2 * p_8 \\ q_8 * p_1 - q_7 * p_2 + q_6 * p_3 + q_5 * p_4 + q_4 * p_5 - q_3 * p_6 + q_2 * p_7 + q_1 * p_8 \end{bmatrix} \quad (7)$$

Un cuaternión dual unitario que representa una rotación pura se define como (8):

$$Dq_R = \begin{bmatrix} \cos \frac{\theta}{2} \\ n_x \sin \frac{\theta}{2} \\ n_y \sin \frac{\theta}{2} \\ n_z \sin \frac{\theta}{2} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (8)$$

Por otro lado, para representar una traslación pura, la parte real se toma como la identidad y la parte dual representa la traslación, como se presenta a continuación (9):

$$Dp_t = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{t_x}{2} \\ \frac{t_y}{2} \\ \frac{t_z}{2} \end{bmatrix} \quad (9)$$

Denavit y Hertenberg en 1955 propusieron un método matricial sistemático para establecer los sistemas de coordenadas asociados a cada eslabón y pasar de uno al siguiente mediante cuatro transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón [5]. En este trabajo se utiliza la convención estándar para la asignación de los sistemas de coordenadas del robot.

La siguiente ecuación permite obtener el cuaternión dual que relaciona el elemento $i-1$ con el elemento i de una cadena cinemática abierta DQ_i^{i-1} compatible con los parámetros de Denavith y Hartenberg (10) [7].

$$DQ = \begin{bmatrix} \cos \frac{\theta_i}{2} * \cos \frac{\alpha_i}{2} \\ \cos \frac{\theta_i}{2} * \sin \frac{\alpha_i}{2} * i \\ \sin \frac{\theta_i}{2} * \sin \frac{\alpha_i}{2} * j \\ \sin \frac{\theta_i}{2} * \cos \frac{\alpha_i}{2} * k \\ \left(\frac{-d_i}{2} * \sin \frac{\theta_i}{2} * \cos \frac{\alpha_i}{2} - \frac{a_i}{2} * \cos \frac{\theta_i}{2} * \sin \frac{\alpha_i}{2} \right) * \mathcal{E} \\ \left(\frac{-d_i}{2} * \sin \frac{\theta_i}{2} * \sin \frac{\alpha_i}{2} + \frac{a_i}{2} * \cos \frac{\theta_i}{2} * \cos \frac{\alpha_i}{2} \right) * i\mathcal{E} \\ \left(\frac{d_i}{2} * \cos \frac{\theta_i}{2} * \sin \frac{\alpha_i}{2} + \frac{a_i}{2} * \sin \frac{\theta_i}{2} * \cos \frac{\alpha_i}{2} \right) * j\mathcal{E} \\ \left(\frac{d_i}{2} * \cos \frac{\theta_i}{2} * \cos \frac{\alpha_i}{2} - \frac{a_i}{2} * \sin \frac{\theta_i}{2} * \sin \frac{\alpha_i}{2} \right) * k\mathcal{E} \end{bmatrix}$$

De tal manera que para modelar la cinemática directa de un manipulador robótico basta con multiplicar los cuaterniones duales que relacionan cada par de elementos adyacentes de acuerdo a la siguiente expresión (11):

$$DQ_n^0 = DQ_1^0 DQ_2^1 \dots DQ_n^{n-1} \quad (11)$$

Donde n es el número de grados de libertad.

La visualización geométrica de los cuaterniones duales no es trivial, es por ello que se realiza una transformación del espacio del cuaternión dual al espacio cartesiano por medio de una matriz de transformación homogénea, esta transformación fue propuesta por Ramírez [8] y se presenta a continuación junto con su desarrollo (12):

$$T = \begin{bmatrix} t_{11} & t_{12} & t_{13} & t_{14} \\ t_{21} & t_{22} & t_{23} & t_{24} \\ t_{31} & t_{32} & t_{33} & t_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12)$$

Donde:

$$\begin{aligned} t_{11} &= 1 - 2(q_2^2 + q_3^2) \\ t_{12} &= 2(q_1q_2 - q_3q_4) \\ t_{13} &= 2(q_1q_3 + q_2q_4) \\ t_{14} &= 2(-q_1q_8 + q_4q_5 - q_3q_6 + q_2q_7) \\ t_{21} &= 2(q_1q_2 + q_3q_4) \\ t_{22} &= 1 - 2(q_1^2 + q_3^2) \\ t_{23} &= 2(q_2q_3 - q_1q_4) \end{aligned}$$

$$\begin{aligned} t_{24} &= 2(-q_2q_8 + q_3q_5 + q_4q_6 - q_1q_7) \\ t_{31} &= 2(q_1q_3 - q_2q_4) \\ t_{32} &= 2(q_2q_3 + q_1q_4) \\ t_{33} &= 1 - 2(q_1^2 + q_2^2) \\ t_{34} &= 2(-q_3q_8 - q_2q_5 + q_1q_6 + q_4q_7) \end{aligned}$$

III. CINEMÁTICA DIRECTA BASADA EN CUATERNIONES DUALES

Para el cálculo de la cinemática directa se obtuvo un modelo unifilar del robot IRB6400, en el cual se asignaron los sistemas de coordenadas de acuerdo a la Fig. 1.

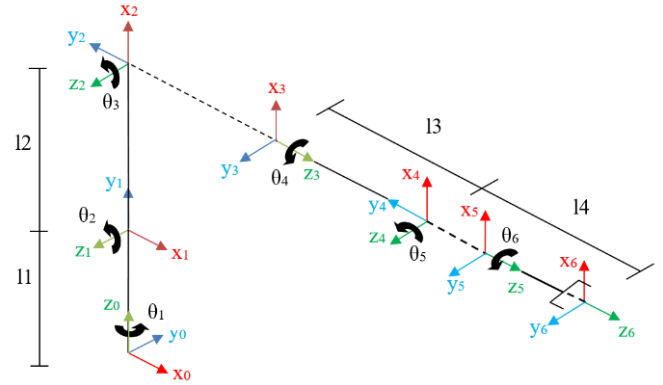


Fig. 1: Sistemas de coordenadas del robot IRB6400.

Posteriormente, se obtuvieron los parámetros de Denavit-Hartenberg (D-H) los cuales se agruparon en la tabla 1.

Tabla 1: Parámetros D-H del robot IRB6400

| i | θ_i | α_i | a_i | d_i |
|---|------------|------------|--------------|--------------|
| 1 | $q_1=0$ | 90 | 0 | $l_1 = 900$ |
| 2 | $q_2=90$ | 0 | $l_2 = 1175$ | 0 |
| 3 | $q_3=0$ | 90 | 0 | 0 |
| 4 | $q_4=0$ | -90 | 0 | $l_3 = 1300$ |
| 5 | $q_5=0$ | 90 | 0 | 0 |
| 6 | $q_6=0$ | 0 | 0 | $l_4 = 200$ |

Estos parámetros se utilizan para obtener seis cuaterniones duales que relacionan a cada par de sistemas de coordenadas adyacentes, los cuales se presentan en las ecuaciones (13) a (18):

$$DQ_1^0 = \begin{bmatrix} 2^{\frac{1}{2}} * \cos\left(\frac{q_1}{2}\right) / 2 \\ \frac{2^{\frac{1}{2}} * \cos\left(\frac{q_1}{2}\right)}{2} i \\ \frac{2^{\frac{1}{2}} * \sin\left(\frac{q_1}{2}\right)}{2} j \\ \frac{2^{\frac{1}{2}} * \sin\left(\frac{q_1}{2}\right)}{2} k \\ -225 * 2^{\frac{1}{2}} * \sin\left(\frac{q_1}{2}\right) \mathcal{E} \\ -225 * 2^{\frac{1}{2}} * \sin\left(\frac{q_1}{2}\right) i\mathcal{E} \\ 225 * 2^{\frac{1}{2}} * \cos\left(\frac{q_1}{2}\right) j\mathcal{E} \\ 225 * 2^{\frac{1}{2}} * \cos\left(\frac{q_1}{2}\right) k\mathcal{E} \end{bmatrix} \quad (13)$$

$$DQ_4^3 = \begin{bmatrix} 2^{\frac{1}{2}} * \cos\left(\frac{q_4}{2}\right) / 2 \\ -\frac{2^{\frac{1}{2}} * \cos\left(\frac{q_4}{2}\right)}{2} i \\ -\frac{2^{\frac{1}{2}} * \sin\left(\frac{q_4}{2}\right)}{2} j \\ \frac{2^{\frac{1}{2}} * \sin\left(\frac{q_4}{2}\right)}{2} k \\ -325 * 2^{\frac{1}{2}} * \sin\left(\frac{q_4}{2}\right) \mathcal{E} \\ 325 * 2^{\frac{1}{2}} * \sin\left(\frac{q_4}{2}\right) i\mathcal{E} \\ -325 * 2^{\frac{1}{2}} * \cos\left(\frac{q_4}{2}\right) j\mathcal{E} \\ 325 * 2^{\frac{1}{2}} * \cos\left(\frac{q_4}{2}\right) k\mathcal{E} \end{bmatrix} \quad (16)$$

$$DQ_2^1 = \begin{bmatrix} \cos\left(\frac{q_2}{2}\right) \\ 0 i \\ 0 j \\ \sin\left(\frac{q_2}{2}\right) k \\ 0 \mathcal{E} \\ \frac{1175 * \cos\left(\frac{q_2}{2}\right)}{2} i\mathcal{E} \\ \frac{1175 * \sin\left(\frac{q_2}{2}\right)}{2} j\mathcal{E} \\ 0 k\mathcal{E} \end{bmatrix} \quad (14)$$

$$DQ_5^4 = \begin{bmatrix} 2^{\frac{1}{2}} * \cos\left(\frac{q_5}{2}\right) / 2 \\ \frac{2^{\frac{1}{2}} * \cos\left(\frac{q_5}{2}\right)}{2} i \\ \frac{2^{\frac{1}{2}} * \sin\left(\frac{q_5}{2}\right)}{2} j \\ \frac{2^{\frac{1}{2}} * \sin\left(\frac{q_5}{2}\right)}{2} k \\ 0 \mathcal{E} \\ 0 i\mathcal{E} \\ 0 j\mathcal{E} \\ 0 k\mathcal{E} \end{bmatrix} \quad (17)$$

$$DQ_3^2 = \begin{bmatrix} 2^{\frac{1}{2}} * \cos\left(\frac{q_3}{2}\right) / 2 \\ \frac{2^{\frac{1}{2}} * \cos\left(\frac{q_3}{2}\right)}{2} i \\ \frac{2^{\frac{1}{2}} * \sin\left(\frac{q_3}{2}\right)}{2} j \\ \frac{2^{\frac{1}{2}} * \sin\left(\frac{q_3}{2}\right)}{2} k \\ 0 \mathcal{E} \\ 0 i\mathcal{E} \\ 0 j\mathcal{E} \\ 0 k\mathcal{E} \end{bmatrix} \quad (15)$$

$$DQ_6^5 = \begin{bmatrix} \cos\left(\frac{q_6}{2}\right) \\ 0 i \\ 0 j \\ \sin\left(\frac{q_6}{2}\right) k \\ -100 * \sin\left(\frac{q_6}{2}\right) \mathcal{E} \\ 0 i\mathcal{E} \\ 0 j\mathcal{E} \\ 100 * \cos\left(\frac{q_6}{2}\right) k\mathcal{E} \end{bmatrix} \quad (18)$$

Posteriormente, se multiplican entre sí los cuaterniones duales de acuerdo a la ecuación (19):

$$DQ_6^0 = DQ_1^0 DQ_2^1 DQ_3^2 DQ_4^3 DQ_5^4 DQ_6^5 \quad (19)$$

$$dq_6 = \frac{1175 * \cos\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right)}{8} - \frac{1175 * \cos\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right)}{8} - \frac{1175 * \cos\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right)}{8} - \frac{1175 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right)}{8} + \frac{1175 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right)}{8} + \frac{1175 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right)}{8} + \frac{1175 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right)}{8} + 75$$

$$+ 75 \sin\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) + 75 \sin\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) - 250 \sin\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) + 250 \sin\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) + 75 \sin\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) + 75 \sin\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) + 75 \sin\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) + 250 \sin\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) - 250 \sin\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) - 250 \sin\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right)$$

$$dq_7 = \frac{250 * \cos\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right)}{8} - \frac{75 * \cos\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right)}{8} - 75 \cos\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) - 75 \cos\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) - 75 \cos\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) - 250 \cos\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) + 250 \cos\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) + 250 \cos\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) + 1175 * \sin\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) + 1175 * \sin\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) + 1175 * \sin\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) + 1175 * \sin\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right)$$

$$dq_8 = \frac{25 * \cos\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} + \frac{q_6}{2}\right)}{8} - \frac{25 * \cos\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} + \frac{q_6}{2}\right)}{8} - 300 * \cos\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} + \frac{q_6}{2}\right) - 300 * \cos\left(\frac{q_1}{2} - \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} + \frac{q_6}{2}\right) + 25 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) - 25 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} + \frac{q_6}{2}\right) + 300 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} - \frac{q_6}{2}\right) + 300 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} + \frac{q_6}{2}\right) + 300 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} - \frac{q_6}{2}\right) + 300 * \cos\left(\frac{q_1}{2} + \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} + \frac{q_6}{2}\right) + 1175 * \sin\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} + \frac{q_6}{2}\right) + 1175 * \sin\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} + \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} + \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} - \frac{q_2}{2} + \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} + \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} - \frac{q_5}{2} + \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} - \frac{q_4}{2} + \frac{q_5}{2} + \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} - \frac{q_5}{2} + \frac{q_6}{2}\right) - 1175 * \sin\left(\frac{q_1}{2} + \frac{q_2}{2} - \frac{q_3}{2} + \frac{q_4}{2} + \frac{q_5}{2} + \frac{q_6}{2}\right)$$

V. CONCLUSIONES

Para validar el modelo anterior se calcula el cuaternión dual numérico para la configuración de home del manipulador con los siguientes valores de coordenadas articulares:

$$q_1 = q_3 = q_4 = q_5 = q_6 = 0$$

$$q_2 = 90$$

$$DQ_6^0 = \begin{bmatrix} 0 \\ 0.7071 \\ 0 \\ 0.7071 \\ -1264 \\ 0 \\ 203.29 \\ 0 \end{bmatrix}$$

Para visualizar el resultado, se realiza una transformación de cuaternión dual a matriz de transformación homogénea:

$$\begin{bmatrix} 0 & 0 & 1 & 1500 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 2075 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Se identifican 88 operaciones entre el producto de dos cuaterniones duales (48 multiplicaciones, 22 sumas y 18 restas)

en comparación con las 104 necesarias para el producto de dos matrices de transformación homogénea (64 multiplicaciones, 40 sumas), se demuestra la ventaja del uso de los cuaterniones duales, dicha ventaja basada en su eficiencia, compacidad y robustez para el modelado de robots.

REFERENCES

- [1] Craig, *Robótica*, Pearson, 2006.
- [2] Barrientos, *Fundamentos de robótica*, Mc Graw Hill, 2007.
- [3] Cayley, "On the application of quaternion to the theory of rotation", *The collected mathematical Papers*, Cambridge University Press, 1:405-409, 1889.
- [4] Ozgur, E., Mezouar, Y. "Kinematic modeling and control of a robot arm using unit dual quaternion", *Robotics and Autonomous Systems*, 77:66-73, 2016.
- [5] J. Denavit, B. and R. S. Hertenberg, "A kinematic Notation for Lower Pair Mechanism base on Matrices" *Journal of Applied Mechanics*, 1955.
- [6] Akyar "Dual quaternions in spatial kinematics in an algebraic sense", *Turkish Journal of mathematics*, 32:373-391, 2008.
- [7] Ramirez, J., Merchan, E. A., Lugo, E., Rodriguez, R. G., Ponce, R., Urriolagoitia, G., "Desarrollo de una nueva solución compacta a la cinemática de manipuladores robóticos basada en cuaterniones duales", *Revista iberoamericana de automática e informática industrial*, 8:334-344, 2011.
- [8] Ramírez J., Planeación de trayectorias en sistemas de manipuladores robóticos múltiples, Tesis de doctorado IPN SEPI ESIME ZAC, 2010.

Diseño e implementación de controlador para aeromodelo por retroalimentación de salida

José Ricardo Cárdenas Valdez
Instituto Tecnológico de Tijuana
ITT
Tijuana, B.C.
jose.cardenas@tectijuana.edu.mx

Luis Tupak Aguilar Bustos
Instituto Politécnico Nacional
CITEDI - IPN
Tijuana, B.C.
laguilarb@ipn.mx

Raúl Rascón Carmona
Facultad de Ingeniería UABC
Mexicali, Baja California
raul.rascon@uabc.edu.mx

Andrés Calvillo Téllez
Instituto Politécnico Nacional
CITEDI - IPN
Tijuana, B.C.
calvillo@citedi.mx

Abstract— El presente trabajo se enfoca al diseño y desarrollo del controlador para un avión a escala que opera en ruta programada. La primera etapa muestra el itinerario de la rutina de vuelo mediante la asignación de valores predefinidos a cuatro variables de control: aceleración, alerones, timón de dirección y timón de profundidad. Posteriormente se trata el sistema eléctrico. Se resuelve el problema de regulación de postura del aeroplano usando modos deslizantes de segundo orden asumiendo medición de salida. Finalmente se realizan las pruebas de funcionalidad del algoritmo en piso y en vuelo

Keywords— *aeromodelo; controlador por retroalimentación; modo deslizante*

I. INTRODUCCIÓN

En las últimas décadas se ha incrementado los esfuerzos en la investigación de vehículos autónomos. Un problema de control de movimiento es el caso de seguimiento de trayectorias, que concierne al diseño de leyes de control que fuerzan a un vehículo a alcanzar y seguir una referencia parametrizada en tiempo. Se destaca que la solución al problema de control de movimiento de sistemas completamente actuados ha sido estudiado por la comunidad científica [2]. El caso de estudio del artículo es un avión a escala.

Uno de los principales retos es que la aeronave debe ser robusta ante perturbaciones aerodinámicas y ruido en los sensores. Otro de los retos es que la respuesta de los actuadores de los alerones es muy lenta y particularmente suele ser afectada por zona muerta y fricción. Entonces el controlador diseñado debe lidiar con estas imperfecciones.

Mejorar el desempeño de un sistema, sea mecánico, eléctrico, biológico, económico, social, o la combinación entre algunos de ellos, siempre ha sido el principal objetivo de la teoría de control. Desde hace varias décadas se han ofrecido métodos y diseños de algoritmos para analizar el comportamiento de dichos sistemas y así ejercer en ellos una

acción o mando. La mayoría de los métodos y diseños de algoritmos de control están basados en el conocimiento del modelo matemático del sistema y asume la medición de todas sus variables, lo que implica la necesidad de contar con la mayor información que describa o caracterice su comportamiento.

Cuando se dice de la información que describe el comportamiento de un sistema, se refiere a lo que se conoce como variables de estado. Las variables de estado son señales o magnitudes de un sistema como posición, velocidad, temperatura, voltaje, fuerza, etc., y que pueden ser entradas, salidas, variables internas e incluso combinaciones de ellas mediante sumas o productos. La importancia de contar con todas las variables de estado de un sistema permite realizar lo que se conoce como realimentación de estado, acción necesaria para llevar a cabo el control o manipulación de las variables de cualquier sistema. En la mayoría de los sistemas físicos esta información no está presente por limitaciones prácticas que puede ser la ausencia de sensores, haciendo necesaria su estimación, que puede lograrse a través de observadores de estado o diferenciadores según sea el caso. Cuando nos referimos a la ausencia de sensores, éstas pueden ser para evitar el incremento del costo del sistema o simplemente puede ser riesgoso para el ser humano colocar el sensor en algún punto determinado de la planta.

Los controladores y estimadores de estructura variable son bien reconocidos por ser robustos y alcanzabilidad en tiempo finito. Durante los últimos veinte años ha habido un auge en el diseño e implementación de diferenciadores y observadores de estructura variable cuya importancia reside en la convergencia exacta de los estados estimados a los estados reales del sistema. En los observadores por modo deslizantes se utiliza una función discontinua cuyo argumento es la diferencia entre la salida observada y la salida medida que ofrece una solución atractiva a

este problema. En otras palabras, un observador por modos deslizantes puede forzar el error de estimación de salida a converger al origen en tiempo finito e inmunidad al ruido [6].

II. DESARROLLO

A. Originalidad del trabajo

Las contribuciones del presente trabajo se citan a continuación:

- Diseño y construcción de la electrónica que permite implementar controladores digitales en aeroplanos. Se destaca que la elaboración del prototipo no es trivial porque un exceso de sobrepeso en el avión altera fácilmente su dinámica.

- Se resuelve el problema de regulación de posición a través de modos deslizantes de segundo orden [3] usando información de la salida debido a que en la práctica las velocidades no están disponibles para ser retroalimentadas.

B. Algoritmo

Para deducir el algoritmo de operación se realiza la planeación de los tiempos y movimientos de operación de las siguientes cuatro variables: aceleración, alerones, timón de dirección y timón de profundidad. Luego se procederá a diseñar el sistema digital de control finalizando con la implementación y desarrollo del circuito empotrado en el aeromodelo

De entre las diferentes modalidades del aeromodelismo, la innovación que se abordó es la de sustituir la manipulación del modelo a escala por un sistema digital con un algoritmo de gobierno de vuelo preestablecido.

Desde un punto de partida denominado origen, se inicializa el avión. Con el motor encendido se procede a realizar la energización del sistema digital para abordar el proceso de despegue: - La trayectoria de despegue durar 4 segundos.

Se desplazan los servomotores de acelerador, timón de dirección y timón de para sustentación de vuelo.

- Trayectoria de ascensión con un tiempo de duración de segundos. Se desplazan los servomotores de acelerador, timón de dirección y timón de para sustentación de vuelo.

- Trayectoria de nivelado con un tiempo de duración de 3 segundos para estabilizar el avión a una altura determinada.

- Viraje de 180° a la izquierda con un tiempo de duración de 3 segundos. - Vuelo nivelado paralelo a pista de aterrizaje durante 10 segundos.

- Viraje a la derecha 360° indefinidamente..

C. Ecuaciones de Movimiento

La dinámica que describe la postura del aeroplano bajo estudio está dada por [4]:

$$\begin{aligned}\ddot{x} &= -\sin(\theta)u_1 + \varepsilon \cos(\theta)u_2 \\ \ddot{y} &= \cos(\theta)u_1 + \varepsilon \sin(\theta)u_2 \\ \ddot{\theta} &= u_2\end{aligned}\quad (1)$$

donde $x(t) \in \mathbf{R}^n$ es el desplazamiento horizontal, $y(t) \in \mathbf{R}$ es el desplazamiento vertical, $\theta(t) \in \mathbf{R}$ es el ángulo de rotación que describe el aeroplano con respecto a la horizontal, $u_1(t) \in \mathbf{R}$ es el empuje total y $u_2(t) \in \mathbf{R}$ es el empuje hacia afuera. La constante ε es un coeficiente positivo suficientemente pequeño que caracteriza el acoplamiento entre el momento de giro y la aceleración lateral del aeroplano.

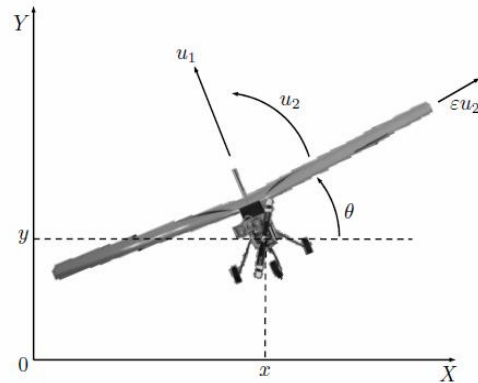


Fig. 1. Aeroplano a escala.

La constante -1 es la aceleración gravitacional normalizada. Asumiendo que $\varepsilon=0$ entonces

$$\ddot{x} = -\sin(\theta)u_1 \quad (2)$$

$$\ddot{y} = \cos(\theta)u_1 \quad (3)$$

$$\ddot{\theta} = u_2. \quad (4)$$

El problema de control se define de la siguiente manera: encontrar las entradas de control $u_1(t)$ y $u_2(t)$ tal que

$$\lim_{t \rightarrow \infty} \|\xi(t)\| = 0 \quad (5)$$

donde $\xi = [x, y, \theta]^T$, asumiendo que el vector de salida $\xi(t) \in \mathbf{R}^3$ se dispone únicamente para retroalimentación.

D. Diseño por medición de salida

Hasta el momento se ha diseñado un controlador que requiere información de todos los estados, sin embargo en la práctica esta información puede no estar disponible. Para este propósito se construye un estimador de velocidad que es aplicable a sistemas no lineales de la forma:

$$\begin{aligned}\dot{\eta}_1 &= \eta_2 \\ \dot{\eta}_2 &= f(\eta_1, \eta_2) \\ \xi &= \eta_1\end{aligned}\quad (6)$$

donde $\eta_1 = (x, y, \theta)^T$ son los estados, $\xi \in \mathbf{R}^3$ es la salida y la función $f(\eta_1, \eta_2)$ es acotada mas no necesariamente suave. Se desea estimar el estado ξ a través de un estimador por modos deslizantes [1]:

$$\begin{aligned}\hat{n}_1 &= \hat{n}_2 + \lambda_1 \text{sign}(\hat{n}_1 - \hat{n}_2) \\ \hat{n}_2 &= f(n_1, \hat{n}_2) + E_1 \lambda_2 \text{sign}(\hat{n}_2 - \hat{n}_2)\end{aligned}$$

$$\begin{aligned}\hat{\eta}_2 &= f(\eta_1, \hat{\eta}_2) + E_1 \lambda_2 \text{sign}(\hat{\eta}_2 - \hat{\eta}_2) \\ \hat{\xi} &= \hat{\eta}_1 \\ \hat{\eta}_2 &= \hat{\eta}_2 + E_1 \lambda_1 \text{sign}(\eta_1 - \hat{\eta}_1)\end{aligned}\quad (7)$$

donde $\hat{\eta}$ representa el estado estimado de η y $E_1=1$ si $\eta_1 = \hat{\eta}_1$ en caso contrario $E_1=0$ y, λ_1 y λ_2 son constantes positivas.

Para extender los resultados del observador por modos deslizantes al problema del PVTOL, considérense los estados $(x, y, \theta, \dot{x}, \dot{y}, \dot{\theta}) = (x_1, y_1, \theta_1, x_2, y_2, \theta_2)$. Entonces la dinámica del estimador para el PVTOL está dada por:

$$\begin{aligned}\hat{x}_1 &= \hat{x}_2 + \lambda_1 \text{sign}(\hat{x}_1) \\ \hat{x}_2 &= -\sin(\theta) \hat{u}_1 + E_1 \lambda_2 \text{sign}(E_1 \lambda_1 \text{sign}(\hat{x}_2)) \\ \hat{\xi}_1 &= \hat{x}_1 \\ \hat{x}_2 &= \hat{x}_2 + E_1 \lambda_1 \text{sign}(\hat{x}_1) \\ \hat{y}_1 &= \hat{y}_2 + \lambda_1 \text{sign}(\hat{y}_1) \\ \hat{y}_2 &= \cos(\theta) \hat{u}_1 + E_1 \lambda_2 \text{sign}(E_1 \lambda_1 \text{sign}(\hat{y}_2)) \\ \hat{\xi}_2 &= \hat{y}_1 \\ \hat{y}_2 &= \hat{y}_2 + E_1 \lambda_1 \text{sign}(\hat{y}_1) \\ \hat{\theta}_1 &= \hat{\theta}_2 + \lambda_1 \text{sign}(\hat{\theta}_1) \\ \hat{\theta}_2 &= \hat{u}_2 + E_1 \lambda_2 \text{sign}(E_1 \lambda_2 \text{sign}(\hat{\theta}_2)) \\ \hat{\xi}_3 &= \hat{\theta}_1 \\ \hat{\theta}_2 &= \hat{\theta}_2 + E_1 \lambda_1 \text{sign}(\hat{\theta}_1)\end{aligned}\quad (8)$$

donde $\hat{\xi}(t)$ representa el estado estimado de ξ , $\tilde{\xi} = \xi - \hat{\xi}$ denota el error de estimación. Debido a las limitaciones de espacio se omite el análisis de estabilidad y será presentado en próxima contribución. Las leyes de control aplicadas para el sistema de lazo cerrado son:

$$\begin{aligned}u_1 &= -\sin(\theta) r_1(x, \hat{x}) + \cos(\theta) (r_2(y, \hat{y}) + 1) \\ u_2 &= \frac{1}{1+\tan^2\theta} \left[-c_1 \hat{\delta} - c_2 \dot{\hat{\delta}} - c_3 \text{sign}(\hat{\delta}) - c_4 \text{sign}(\dot{\hat{\delta}}) + \ddot{\delta} \right] - 2\tan(\theta) \hat{\delta}^2\end{aligned}\quad (9)$$

donde

$$\delta = \tan(\theta) + \frac{r_1}{r_2+1}\quad (10)$$

$$\begin{aligned}r_1 &= -\sin(\theta) u_1 = -k_2 \dot{x} - k_1 x \\ r_2 &= \cos(\theta) u_1 - 1 = -k_3 \dot{y} - k_4 y\end{aligned}$$

donde k_i ($i=1, \dots, 4$) son constantes positivas. Por limitaciones de espacio se omite, por el momento, el procedimiento formal para encontrar el controlador así como su prueba de estabilidad realizada a través del criterio de Lyapunov.

III. RESULTADOS

Este artículo trata sobre el desarrollo en implementación de un prototipo de un sistema digital. La figura 1 muestra al sistema digital realizando las pruebas de funcionalidad de los servomotores y microcontrolador operando de acuerdo al algoritmo propuesto. Esta misma figura presenta al sistema digital, y el ensamblaje del aeromodelo y el sistema digital donde se verifica su funcionalidad antes de pasar a la prueba en pista y campo. En la figura 2 y 3 se presentan las imágenes del aeroplano en pleno vuelo gobernado por el algoritmo programado. La efectividad del control por modos deslizantes se probó a través de simulaciones usando SIMNON. Las ganancias del controlador (8) y (9) fueron ajustadas arbitrariamente a $c_1=15.0$, $c_2=c_3=8.0$, $c_4=5.0$, $\lambda_1=0.01$ y $\lambda_2=0.01$. La Figura 4 muestra las respuestas en el tiempo de la salida ξ .

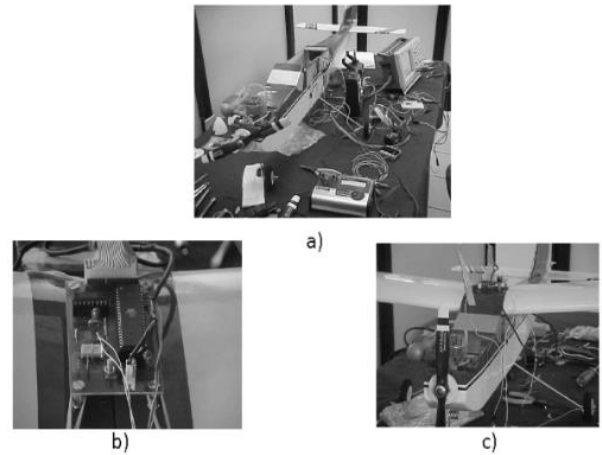


Fig. 2. (a) Prueba de funcionalidad de los servomotores y microcontrolador, (b) sistema digital controlador en el avión a escala, y (c) controlador digital en el avión a escala.

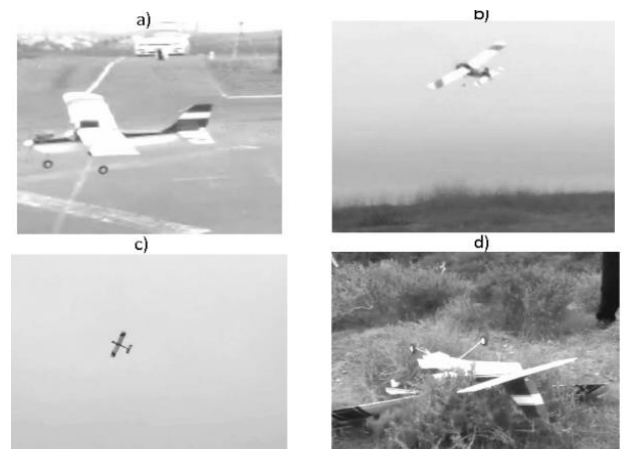


Fig. 3. (a) Trayectoria de elevación, (b) Trayectoria de giro a 180°, (c) trayectoria circular de 360° y (d) Resultado final del primer vuelo.

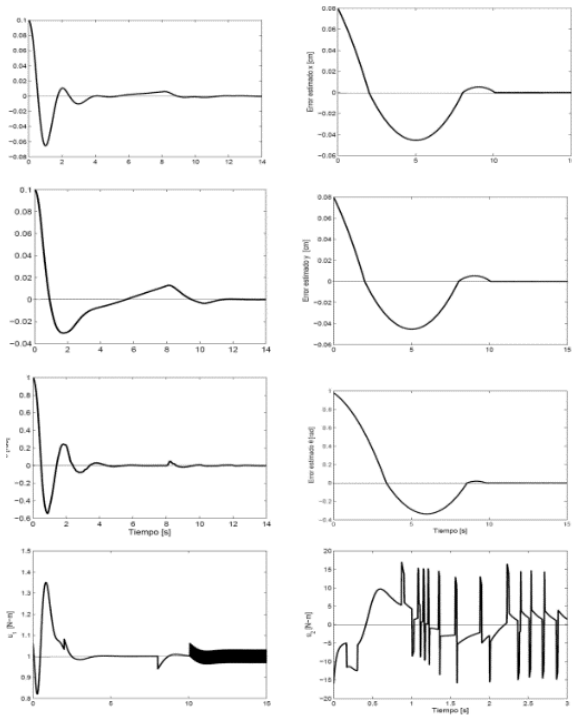


Fig. 4. Resultados de simulación para posiciones, velocidades y pares aplicados

Uno de los tópicos a abordar por la ciencia y la investigación es el empleo de aeronaves prototipos en la innovación de nuevos

modelos de manipulación y control. Lo que sigue en esta línea es incluir un sistema de posicionamiento global y una cámara digital como elementos de orientación. El uso de las herramientas de modos deslizantes para resolver el problema de estabilización de aeroplanos se presentó en éste trabajo. Entre las ventajas del método es robustez y convergencia en tiempo finito al punto de equilibrio en sistemas no tripulados. Se utilizó un estimador por modos deslizantes de segundo orden para determinar la velocidad. Los resultados de simulación muestran la efectividad del método.

BIBLIOGRAFÍA

- [1] J.P. Barbot, M. Djemai and T. Boukhobza, Sliding mode observers in Sliding Mode Control in Engineering, W. Perruquetti and J.P. Barbot (Eds.), pp. 103--130, 2002
- [2] I. Fantoni and R. Lozano, *Non-linear Control for Underactuated Mechanical System*. London, Springer, 2002
- [3] A. Levant, Higher-order sliding modes, differentiation and output-feedback control, *International Journal of Control*, vol. 76, no. 9, pp. 924--941, 2003
- [4] A. Zavala-Rio, I. Fantoni and R. Lozano, Global stabilization of a PVTOL aircraft model with bounded inputs, *International Journal of Control*, vol. 76, no. 18, pp. 1833--1844, 2003
- [5] <https://riunet.upv.es/bitstream/handle/10251/33064/Memoria.pdf?sequence=1>
- [6] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant, *Sliding Mode Control and Observation*. New York: Birkhauser, 2014.

Modelado Cinemático y Dinámico de un Robot Móvil No Holonómico Considerando Dinámica de Actuadores

†Armando Sáenz*, Tadeo Espinoza, Juan Rentería
 Facultad de Ingeniería, Ciencias y Arquitectura
 Universidad Juárez del Estado de Durango
 Gómez Palacio, Durango
 *Email: jsaenz@ujed.mx

†Javier Ollervides
 †Tecnológico Nacional de México
 Instituto Tecnológico de la Laguna
 Torreón, Coahuila

Resumen—El presente artículo muestra la forma de obtener los modelos cinemático y dinámico de un robot móvil no holonómico. El robot móvil cuenta con dos ruedas fijas desplazadas una misma distancia respecto al centro del robot. Para el modelo cinemático se realiza el análisis desde la obtención de las restricciones de movimiento de cada rueda. Para el modelo dinámico se emplea la formulación de Euler-Lagrange. Las restricciones de movimiento son incluidas al modelo dinámico mediante el uso de multiplicadores de Lagrange. Posteriormente se incluye la dinámica eléctrica de los motores CD que están acoplados a las ruedas. Se realizan las transformaciones necesarias para dejar el modelo expresado en coordenadas generalizadas y con el voltaje de armadura de cada motor DC como señal de entrada.

I. INTRODUCCIÓN

Un Robot Móvil con Ruedas (RMR) es un vehículo terrestre equipado con ruedas que tiene la capacidad de realizar movimientos autónomos (sin interacción humana) [1]. Los actuadores más comúnmente utilizados son motores de corriente directa (CD) que son controlados por una computadora a bordo. Para realizar el modelado del RMR se supone que los componentes que conforman al robot son rígidos (incluidas las ruedas), y que éste se mueve en un plano horizontal.

En [2], se menciona que un RMR está conformado por tres subsistemas (véase Figura 1): *Arreglo cinemático*, el cual se genera de las configuraciones relativas a la disposición de las ruedas en la estructura del robot. Los principales arreglos son: síncrono, triciclo, Ackerman, diferencial, omnidireccional y skid steer; *Sistema de actuadores*, que permite el movimiento de la estructura cinemática (e.g. motores CD, motores sin escobillas); *Convertidor de potencia*, que permite el accionamiento de los actuadores.

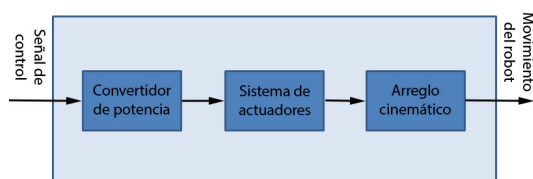


Figura 1. Subsistemas de un robot móvil

La mayoría de los RMRs se enmarcan en el área de sistemas no-holonómicos, los cuales se caracterizan por estar sujetos a restricciones no integrables en las velocidades; es decir, el plano de velocidades está restringido. Los problemas de control más importantes son los asociados al posicionamiento, regulación o estabilización, seguimiento de trayectorias y evasión de obstáculos.

En el área de control para RMRs, algunos controles se diseñan considerando únicamente la cinemática del robot y en otros casos un modelo dinámico, el cual tiene como entrada los torques aplicados en cada rueda. En [1] se presenta un método para obtener la dinámica de cualquier RMR que cumpla con ciertas condiciones, se emplea la formulación Lagrangiana junto con los multiplicadores de Lagrange para considerar las restricciones no holonómicas. En [3] se realiza un análisis de las propiedades holonómicas de distintos RMRs. En [4] se presenta la validación del modelo cinemático del RMR Rovio. En [5] se detalla el modelo cinemático de un RMR omnidireccional de 4 ruedas también con validación experimental. En [6] se diseña un control considerando los torques aplicados en cada rueda como la entrada del modelo dinámico.

El presente trabajo describe la obtención del modelo dinámico de un RMR diferencial, tomándose en consideración la dinámica de los actuadores. A continuación se describen las cinco secciones siguientes. Se comienza con la sección II mostrando las definiciones y teorema matemáticos que serán empelados. Posteriormente la sección III contiene la forma en como se describe la postura del robot y los pasos para obtener las restricciones de movimiento de las ruedas fijas. Después, en la sección IV, se presenta la demostración no holonómicidad por el teorema de Frobenius. En la sección V se presenta la metodología por el cual se obtiene el modelo dinámico de la parte mecánica del robot y posteriormente al modelo se le agrega la dinámica de los actuadores. Por último, en la sección VI se encuentran las conclusiones.

Nota: Las funciones trigonométricas $\sin(\epsilon)$ y $\cos(\epsilon)$ serán representadas como s_ϵ y c_ϵ , respectivamente.

II. PRELIMINARES MATEMÁTICOS

Las definiciones y el teorema que se presentan en esta sección pueden ser revisadas en [7]–[9].

Definición 2.1: Derivada temporal de una matriz de rotación. Sea $R_j^i(\theta) \in \mathbb{R}^{3 \times 3}$ es una matriz de rotación del marco j respecto al marco i y $\mathbf{u} = [u_1 \ u_2 \ u_3]^\top$ es el vector unitario en el que ocurre la rotación, $[\cdot]_x$ es la matriz antisimétrica asociada al vector \cdot y $\dot{\theta}$ es la velocidad angular. Para obtener la derivada temporal de la matriz de rotación ($\dot{R}_j^i(\theta)$) se utiliza la siguiente expresión

$$\dot{R}_j^i(\theta) = [\mathbf{u}\dot{\theta}]_x R_j^i(\theta) = \dot{\theta} \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} R_j^i(\theta) \quad (1)$$

Definición 2.2: Suma de velocidades angulares. Para conocer la velocidad angular de un marco n respecto a un marco 0 ($\omega_{0,n}^0$) con marcos intermedios que también tienen velocidad angular se emplea la siguiente expresión

$$\omega_{0,n}^0 = \sum_{i=0}^{n-1} R_i^0 \omega_{i,i+1}^i = \sum_{i=0}^{n-1} \omega_{i,i+1}^0,$$

donde R_i^0 es la matriz de rotación del marco i respecto al marco 0 y $\omega_{i,i+1}^i$ es la velocidad del marco $i+1$ respecto al marco i .

Definición 2.3: Distribución. Supóngase la existencia de i vectores (i.e. $f_1(x), \dots, f_i(x) \in \mathbb{R}^n$) definidos en $x \in U$. Entonces una distribución con dichos vectores se define como

$$\Delta(x) = \text{span}\{f_1(x), \dots, f_m(x)\}.$$

Una distribución es no singular dentro de un conjunto U , si existe un entero $d > 0$ tal que

$$\dim(\Delta(x)) = \text{rank}[f_1(x), \dots, f_m(x)] = d \quad \forall \quad x \in U.$$

Definición 2.4: Producto de Lie. Sean $f(x)$ y $g(x)$ dos vectores, ambos definidos sobre el conjunto U . A partir de ambos se puede construir un tercer vector, denotado como $[f, g](x)$ el cual está definido como

$$[f, g](x) = \frac{\partial g}{\partial x} f(x) - \frac{\partial f}{\partial x} g(x).$$

Definición 2.5: Distribución involutiva. Una distribución no singular, es involutiva si y sólo si

$$\text{rank}[f_1(x) \ \dots \ f_m(x)] = \text{rank}[f_1(x) \ \dots \ f_m(x) \ [f_i, f_j](x)] \\ i, j = 1, 2, \dots, m.$$

Teorema 2.1: Teorema de Frobenius. Una distribución no singular es completamente integrable si y sólo si ésta es involutiva.

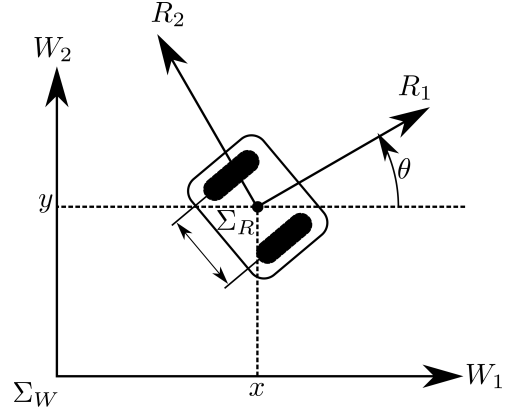


Figura 2. Marco inercial (Σ_w) y marco del robot (Σ_R)

III. RESTRICCIONES DE MOVIMIENTO DE LAS RUEDAS FIJAS

Para describir el movimiento de un RMR se define un marco inercial Σ_w denotado por $O_w - w_1 w_2$ siendo O_w su origen. También se define un marco Σ_R fijado al robot, denotado por $O_R - R_1 R_2$ con su origen en O_R (véase la Figura 2). La postura del robot es especificada por tres variables x, y, θ :

- x, y son las variables que indican la posición del origen del marco Σ_R con respecto al marco inercial Σ_w .

$$\mathbf{p} = [x \ y]^\top. \quad (2)$$

- θ indica la orientación del marco Σ_R respecto al marco Σ_w .

La postura del robot se define como $\xi = [x \ y \ \theta]^\top$. Sea $\dot{\xi} \in \mathbb{R}^3$ el vector de velocidades del carro visto desde el marco Σ_w . Para conocer las velocidades del carro vistas desde el marco Σ_R , representadas por $\eta \in \mathbb{R}^3$, se realiza lo siguiente:

$$\eta = \begin{bmatrix} \dot{x}^R \\ \dot{y}^R \\ \dot{\theta}^R \end{bmatrix} = R_w^R(\theta) \dot{\xi}, \quad R_w^R(\theta) = \begin{bmatrix} c_\theta & s_\theta & 0 \\ -s_\theta & c_\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

donde $c_\theta = \cos(\theta)$ y $s_\theta = \sin(\theta)$.

Considérese una rueda orientable como la mostrada en la Figura 3. Sea Σ_{R_i} un marco fijado a la i -ésima rueda en el punto de contacto del suelo con la rueda i ($i = 1, 2, 3, \dots$), donde \mathbf{R}_{i1} se coloca en dirección de avance de la rueda i . La posición de la rueda i respecto al marco Σ_R está descrita mediante los siguientes parámetros:

- l_{i1} es la distancia desde el origen del marco Σ_R hasta el punto de contacto de la rueda i con el suelo a lo largo del eje \mathbf{R}_{i1} .
- l_{i2} es la distancia desde el origen del marco Σ_R hasta el punto de contacto de la rueda i con el suelo a lo largo del eje \mathbf{R}_{i2} .
- α_i es el ángulo que se forma entre la componente \mathbf{R}_{i1} y \mathbf{R}_{i1} .

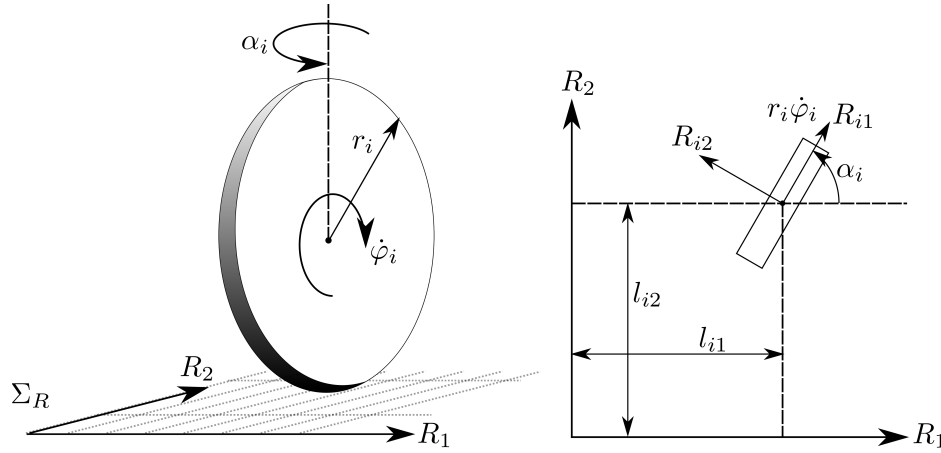


Figura 3. Parámetros cinemáticos en una rueda fija

La velocidad que debe existir en la rueda i cuando el robot se mueve a una velocidad η se calcula mediante:

$$\begin{bmatrix} \dot{O}_{R_{i1}} \\ \dot{O}_{R_{i2}} \\ 0 \end{bmatrix}^R = \eta + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta} \end{bmatrix} \times \begin{bmatrix} l_{i1} \\ l_{i2} \\ 0 \end{bmatrix} \quad (4)$$

$$\begin{bmatrix} \dot{O}_{R_{i1}} \\ \dot{O}_{R_{i2}} \\ 0 \end{bmatrix}^R = \begin{bmatrix} \dot{x}^R - l_{i2}\dot{\theta} \\ \dot{y}^R + l_{i1}\dot{\theta} \\ 0 \end{bmatrix}. \quad (5)$$

Las velocidades obtenidas en (5) están referenciadas al marco Σ_R . Se define la matriz de rotación $R_R^{r_i}$ para rotar del marco Σ_R al marco Σ_{r_i}

$$R_R^{R_i}(\alpha_i) = \begin{bmatrix} c_{\alpha_i} & s_{\alpha_i} & 0 \\ -s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (6)$$

Las velocidades vistas desde el marco Σ_{R_i} son

$$\begin{bmatrix} \dot{O}_{R_{i1}} \\ \dot{O}_{R_{i2}} \\ 0 \end{bmatrix}^{R_i} = R_R^{R_i}(\alpha_i) \begin{bmatrix} \dot{O}_{R_{i1}} \\ \dot{O}_{R_{i2}} \\ 0 \end{bmatrix}^R. \quad (7)$$

Se define $\dot{\varphi}_i$ como la velocidad de giro de la rueda i vista desde R_{i2} , r_i es el radio de la rueda i . La velocidad entregada por la rueda es $r_i\dot{\varphi}_i$ en la dirección de R_{i1} (suponiendo las restricciones de rodamiento puro y no deslizamiento). De manera que utilizando (5), (6) y (7) queda

$$\begin{bmatrix} r_i\dot{\varphi}_i \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} c_{\alpha_i} & s_{\alpha_i} & l_{i1}s_{\alpha_i} - l_{i2}c_{\alpha_i} \\ -s_{\alpha_i} & c_{\alpha_i} & l_{i1}c_{\alpha_i} + l_{i2}s_{\alpha_i} \\ 0 & 0 & 0 \end{bmatrix} \eta. \quad (8)$$

Las restricciones de movimiento son obtenidas de los dos primeros renglones del sistema de ecuaciones (8) esto es

$$[-c_{\alpha_i} \quad -s_{\alpha_i} \quad -l_{i1}s_{\alpha_i} + l_{i2}c_{\alpha_i}] \eta + r_i\dot{\varphi}_i = 0, \quad (9)$$

$$[-s_{\alpha_i} \quad c_{\alpha_i} \quad l_{i1}c_{\alpha_i} + l_{i2}s_{\alpha_i}] \eta = 0. \quad (10)$$

La ecuación (9) indica que la velocidad de la rueda debe de ser equivalente a la proporcionada por el giro de la rueda

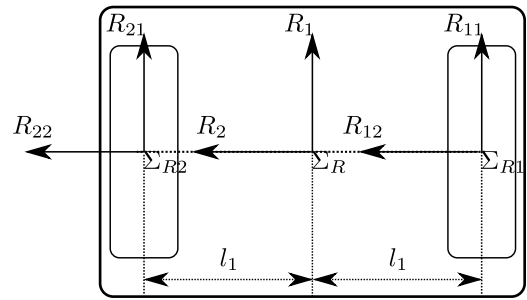


Figura 4. Parámetros del RMR

(rodamiento puro) y de la ecuación (10) se distingue que la componente de velocidad lineal en Σ_{R2} debe ser cero (no deslizamiento lateral).

De la ecuación (9) se obtiene el modelo cinemático de velocidad inverso de la siguiente manera. Las restricciones de rodamiento puro para el RMR son

$$\begin{bmatrix} -c_{\alpha_1} & -s_{\alpha_1} & -l_{11}s_{\alpha_1} + l_{12}c_{\alpha_1} \\ c_{\alpha_2} & s_{\alpha_2} & l_{21}s_{\alpha_2} - l_{22}c_{\alpha_2} \end{bmatrix} R_W^R(\theta)\dot{\xi} = r\dot{\varphi} \quad (11)$$

Los valores a sustituir son los siguientes (véase Figura 4): $\alpha_1 = 0$, $\alpha_2 = 0$, $l_{11} = 0$, $l_{21} = 0$, $l_{12} = -l_1$ y $l_{22} = l_1$. Sustituyendolos se obtiene el modelo cinemático inverso de velocidad

$$\dot{\varphi} = ER_W^R(\theta)\dot{\xi} \quad (12)$$

con

$$\begin{aligned} E &= -J_2^{-1}J_1 \\ \dot{\varphi} &= [\dot{\varphi}_1 \quad \dot{\varphi}_2]^T \\ J_1 &= \begin{bmatrix} -1 & 0 & -l_1 \\ -1 & 0 & l_1 \end{bmatrix} \\ J_2 &= \text{diag}\{r, r\} \end{aligned}$$

IV. ANÁLISIS CINEMÁTICO

En esta sección se presentan el modelo cinemático de velocidad directo junto con la prueba de integrabilidad de Frobenius para demostrar que es un sistema no holonómico. El modelo cinemático directo es el siguiente

$$\boldsymbol{\eta} = \begin{bmatrix} \frac{r}{2}c_\theta & \frac{r}{2}c_\theta \\ \frac{r}{2}s_\theta & \frac{r}{2}s_\theta \\ \frac{r}{2l_1} & -\frac{r}{2l_1} \end{bmatrix} \dot{\boldsymbol{\phi}} \quad (13)$$

donde l_1 es la distancia entre las ruedas y r es el radio de las ruedas. Este modelo presenta la siguiente distribución $\Delta(\boldsymbol{\xi})$

$$f_1(\boldsymbol{\xi}) = \begin{bmatrix} \frac{r}{2}c_\theta & \frac{r}{2}s_\theta & \frac{r}{2l_1} \end{bmatrix}^T \quad (14)$$

$$f_2(\boldsymbol{\xi}) = \begin{bmatrix} \frac{r}{2}c_\theta & \frac{r}{2}s_\theta & -\frac{r}{2l_1} \end{bmatrix}^T \quad (15)$$

$$\Delta(\boldsymbol{\xi}) = \{f_1, f_2\}. \quad (16)$$

La distribución $\Delta(\boldsymbol{\xi})$ es no singular ya que $\dim(\Delta(\boldsymbol{\xi})) = 2$ para cualquier punto $\boldsymbol{\xi}$. Obteniendo el producto de Lie como se muestra a continuación:

$$[f_1, f_2](\boldsymbol{\xi}) = \begin{bmatrix} -\frac{r^2 s_\theta}{2l_1} & \frac{r^2 c_\theta}{2l_1} & 0 \end{bmatrix}^T. \quad (17)$$

Con lo anterior ahora se tiene la distribución $\hat{\Delta}(\boldsymbol{\xi}) = \{f_1, f_2, [f_1, f_2]\}$ que es

$$\hat{\Delta}(\boldsymbol{\xi}) = \begin{bmatrix} \frac{r}{2}c_\theta & \frac{r}{2}c_\theta & -\frac{r^2 s_\theta}{2l_1} \\ \frac{r}{2}s_\theta & \frac{r}{2}s_\theta & \frac{r^2 c_\theta}{2l_1} \\ \frac{r}{2l_1} & -\frac{r}{2l_1} & 0 \end{bmatrix}. \quad (18)$$

La dimensión de $\hat{\Delta}(\boldsymbol{\xi})$ es 3 porque el determinante es $\frac{r^4}{2l_1^2} \forall \boldsymbol{\xi}$ y con lo cual no se cumple el Teorema 2.1 (i.e., $\dim(\Delta(\boldsymbol{\xi})) \neq \dim(\hat{\Delta}(\boldsymbol{\xi}))$). Con esto se demuestra que el sistema es no integrable y por lo tanto es no holonómico.

V. MODELO DINÁMICO

La técnica utilizada para obtener el modelo dinámico es Euler-Lagrange. Existen tres cuerpos a analizar: el cuerpo, la rueda 1 y la rueda 2. Se definen \boldsymbol{q} y $\dot{\boldsymbol{q}}$ de la siguiente manera

$$\boldsymbol{q} \triangleq \begin{bmatrix} x \\ y \\ \theta \\ \varphi_1 \\ \varphi_2 \end{bmatrix} \quad \dot{\boldsymbol{q}} \triangleq \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi}_1 \\ \dot{\varphi}_2 \end{bmatrix} \quad (19)$$

Los parámetros m e I son la masa y la inercia del cuerpo del robot, m_R es la masa de ambas ruedas e $\mathcal{I} = \text{diag}\{I_x, I_y, I_z\}$ es el tensor de inercia en ambas ruedas. Como se considera que

las ruedas son simétricas entonces $I_x = I_z$. Las velocidades rotacionales son las siguientes:

$$\omega_{W,R}^W = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta} \end{bmatrix} \quad (20)$$

$$\omega_{W,R_1}^W = \omega_{W,R}^W + \omega_{R_1,R_1}^W = \begin{bmatrix} -\dot{\varphi}_1 s_\theta \\ \dot{\varphi}_1 c_\theta \\ \dot{\theta} \end{bmatrix} \quad (21)$$

$$\omega_{W,R_2}^W = \omega_{W,R}^W + \omega_{R_2,R_2}^W = \begin{bmatrix} -\dot{\varphi}_2 s_\theta \\ \dot{\varphi}_2 c_\theta \\ \dot{\theta} \end{bmatrix} \quad (22)$$

La energía cinética de los tres cuerpos es

$$\mathcal{K}_R = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}\dot{\theta}^2 I,$$

$$\mathcal{K}_{R_1} = \frac{1}{2}m_R(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}(\omega_{W,R_1}^W)^\top R_{R_1}^W(\theta, \varphi_1) \mathcal{I} R_{R_1}^{R_1}(\theta, \varphi_1) \omega_{W,R_1}^W,$$

$$\mathcal{K}_{R_2} = \frac{1}{2}m_R(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}(\omega_{W,R_2}^W)^\top R_{R_2}^W(\theta, \varphi_2) \mathcal{I} R_{R_2}^{R_2}(\theta, \varphi_2) \omega_{W,R_2}^W.$$

La suma de energías \mathcal{L} es

$$\begin{aligned} \mathcal{L} &= \mathcal{K}_R + \mathcal{K}_{R_1} + \mathcal{K}_{R_2} \\ &= \frac{1}{2}(\dot{\theta}^2 I_R + m\dot{x}^2 + m\dot{y}^2 + I_y \dot{\varphi}_1^2 + I_y \dot{\varphi}_2^2) \\ &\quad + m_R \dot{x}^2 + m_R \dot{y}^2 + \dot{\theta}^2 I_x + \dot{\theta}^2 l_1^2 m_R \end{aligned} \quad (23)$$

Sea $F_x \in \mathbb{R}$ la fuerza lineal del cuerpo del robot en dirección del eje $R1$, $F_y \in \mathbb{R}$ la fuerza lineal del cuerpo del robot en dirección de $R2$, τ_θ el par en el cuerpo del robot perpendicular al plano de movimiento, τ_1 el par en la rueda 1, τ_2 el par en la rueda 2 y $\boldsymbol{\tau} = [F_x \ F_y \ \tau_\theta \ \tau_1 \ \tau_2]^\top$ el vector que contiene fuerzas y pares del robot. La ecuación de movimiento de Lagrange es

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}}} \right) - \frac{\partial \mathcal{L}}{\partial \boldsymbol{q}} = \boldsymbol{\tau} \quad (24)$$

Al resolver todas las operaciones se obtiene el siguiente modelo dinámico

$$M \ddot{\boldsymbol{q}} = \boldsymbol{\tau} \quad (25)$$

con

$$\begin{aligned} M &= \begin{bmatrix} M_R & 0 \\ 0 & M_r \end{bmatrix} \\ M_R &= \text{diag}\{m + 2m_R, m + 2m_R, 2m_R l_1^2 + I + 2I_x\}, \\ M_r &= \text{diag}\{I_y, I_y\}. \end{aligned}$$

El modelo dinámico 25 tiene todos los estados desacoplados, esto es debido a que las restricciones de movimiento todavía no son incluidas. Para incluir la restricciones se hace uso de los multiplicadores de Lagrange. Retomando la ecuación (9) se reescribe el modelo como es indicado en la referencia [1]

$$R_W^R(\theta)M_R\ddot{\xi} = J_1^\top \lambda, \quad (26)$$

$$E^\top M_r \ddot{\varphi} = E^\top J_2^\top \lambda + E^\top \tau_\varphi, \quad (27)$$

con

$$E = -J_2^{-1} J_1,$$

Sumando ecuaciones (26) y (27) el modelo se reduce a lo siguiente

$$R_W^R(\theta)M_R\ddot{\xi} + E^\top M_r \ddot{\varphi} = E^\top \tau_\varphi \quad (28)$$

El modelo esta expresado en términos de las aceleraciones generalizadas y la aceleración angular de las ruedas. En lo sub-siguiente se describe la metodología que el modelo únicamente dependa de las velocidades y aceleraciones generalizadas. Se deriva la expresión (12) respecto al tiempo obteniendo lo siguiente

$$ER_W^R(\theta)\dot{\xi} + ER_W^R(\theta)\ddot{\xi} = \ddot{\varphi}. \quad (29)$$

Para determinar $\dot{R}_W^R(\theta)$ se emplea la definición 2.1

$$\dot{R}_W^R(\theta) = \begin{bmatrix} 0 & \dot{\theta} & 0 \\ -\dot{\theta} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} R_W^R(\theta) \quad (30)$$

Sustituyendo (29) en (28) y multiplicando por $R_R^W(\theta)$ se obtiene

$$\bar{M}(\xi)\ddot{\xi} + \bar{C}(\xi, \dot{\xi})\dot{\xi} = R_R^W(\theta)E^\top \tau_\varphi \quad (31)$$

con

$$\bar{M}(\xi) = \begin{bmatrix} m_{11} & m_{12} & 0 \\ m_{12} & m_{11} & 0 \\ 0 & 0 & m_{33} \end{bmatrix} \quad (32)$$

$$m_{11} = m + 2m_R + \frac{2I_y \cos(\theta)^2}{r^2} \quad (33)$$

$$m_{12} = \frac{I_y \sin(2\theta)}{r^2} \quad (34)$$

$$m_{33} = I_R + 2I_x + 2l_1^2 m_R + \frac{2I_y l_1^2}{r^2} \quad (35)$$

$$\bar{C}(\xi, \dot{\xi}) = \begin{bmatrix} c_{11} & c_{12} & 0 \\ c_{21} & -c_{11} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (36)$$

$$c_{11} = -\frac{\dot{\theta} I_y \sin(2\theta)}{r^2} \quad (37)$$

$$c_{12} = \frac{2\dot{\theta} I_y \cos(\theta)^2}{r^2} \quad (38)$$

$$c_{21} = -\frac{2\dot{\theta} I_y \sin(\theta)^2}{r^2} \quad (39)$$

Para analizar la dinámica del motor se considera un motor de corriente continua con inductancia despreciable en cada

una de las ruedas. El diagrama del motor se presenta en la Figura 5. Las ecuaciones que describen el comportamiento del motor fueron obtenidas de [10], [11]. Sea v el voltaje de armadura del motor ([V]), i_a es la corriente en el motor ([A]), R_a es la resistencia de armadura ([Ω]), k_a la constante de torque ($\frac{A}{Nm}$]), k_b la constante de fuerza contraelectromotriz ($\frac{Vs}{rad}$]), J_m la inercia de la flecha ([Kg m²]), $f_m(q_m)$ la fricción en el eje del motor en [Nm], r_e es la relación de reducción de engranes, φ_i es el angulo en la rueda i ([rad]) y τ_{φ_i} es el torque en la rueda i ([Nm]). Las ecuaciones que describen el comportamiento del motor son

$$v = i_a R_a + e_b \quad (40)$$

$$e_b = \dot{q}_{mi} k_b \quad (41)$$

$$\dot{q}_{mi} = r_e \dot{\varphi}_i \quad (42)$$

$$\tau_m = i_a k_a \quad (43)$$

$$J_m \ddot{q}_{mi} = \tau_m - f_m(\dot{q}_{mi}) - \frac{\tau_{\varphi_i}}{r_e} \quad (44)$$

$$\ddot{q}_{mi} = r_e \ddot{\varphi}_i \quad (45)$$

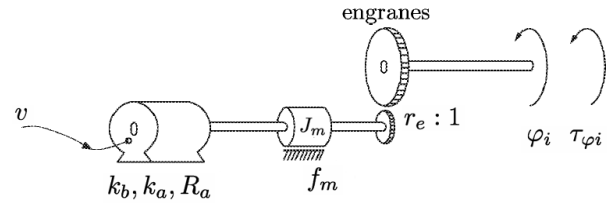


Figura 5. Diagrama de un motor de corriente directa

Despejando i_a de (40) y sustituyendo (42) se tiene

$$i_a = \frac{1}{R_a}(v - k_b r_e \dot{\varphi}_i), \quad (46)$$

y sustituyendo i_a en la ecuación (43) se obtiene

$$\tau_m = \frac{k_a}{R_a}(v - k_b r_e \dot{\varphi}_i). \quad (47)$$

Sustituyendo las ecuaciones (47), (45) en (44) se llega a

$$J_m r_e \ddot{\varphi}_i = \frac{k_a}{R_a}(v - k_b r_e \dot{\varphi}_i) - f_m(\dot{q}_{mi}) - \frac{\tau_{\varphi_i}}{r_e}. \quad (48)$$

Conociendo que cada uno de los motores que mueven las ruedas del robot tienen las mismas características, se puede escribir el modelo en forma matricial como se muestra a continuación

$$J_m r_e \ddot{\varphi} = \frac{k_a}{R_a}(\mathbf{u} - k_b r_e \dot{\varphi}) - \bar{\mathbf{f}}_m - \frac{1}{r_e} \tau_\varphi \quad (49)$$

$$\mathbf{u} = [v_1 \ v_2]^\top$$

$$\bar{\mathbf{f}}_m = [f_m(\dot{q}_{m1}) \ f_m(\dot{q}_{m2})]^\top$$

donde v_1, v_2 son los voltajes aplicados a los motores de las ruedas. Para obtener el modelo dinámico del robot tomando en cuenta la dinámica del motor se realiza el siguiente procedimiento. Sea k_v el coeficiente de fricción viscosa en el eje

de motor. Sustituyendo (49), (12) y (29) en (31) se obtiene la siguiente expresión

$$M_v \ddot{\xi} + C_v(\xi) \dot{\xi} + D_v(\xi) \xi = R_R^W(\theta) E^T \frac{k_a}{k_b} \mathbf{u} \quad (50)$$

con

$$\begin{aligned} M_v(\xi) &= \frac{1}{r_e} \bar{M} + J_m r_e R_R^W(\theta) E^T E R_W^R(\theta) \\ C_v(\xi, \dot{\xi}) &= \frac{1}{r_e} \bar{C}(\xi, \dot{\xi}) + J_m r_e \dot{\theta} R_R^W(\theta) E^T E B R_W^R(\theta) \\ D_v(\xi) &= r_e \left(k_v + \frac{k_a k_b}{R_a} \right) R_R^W(\theta) E^T E R_W^R(\theta) \\ B &= \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{aligned}$$

La representación en espacio de estados es la siguiente

$$\begin{aligned} \frac{d}{dt} \xi &= \dot{\xi}, \\ \frac{d}{dt} \dot{\xi} &= M_v^{-1}(\xi) \left(R_R^W(\theta) E^T \frac{k_a}{k_b} \mathbf{u} - C_v(\xi) \dot{\xi} - D_v(\xi) \xi \right). \end{aligned} \quad (51)$$

El espacio de estados tiene como señal de entrada \mathbf{u} los voltajes de armadura que se aplican a ambos motores de las ruedas. Las matrices $M_v(\xi)$ y $C_v(\xi, \dot{\xi})$ cumplen con la propiedad de antisimetría

$$\chi^T \left(\dot{M}_v(\xi) - \frac{1}{2} C_v(\xi, \dot{\xi}) \right) \chi = 0. \quad (52)$$

La matriz D_v es simétrica ($D_v = D_v^T$) y definida positiva ($\chi^T D_v \chi > 0$) contiene el coeficiente de fricción y de fuerza contraelectromotriz, constante de torque y relación de engranes. En dicha matriz se encuentran todos los coeficientes de los componentes que transforman la energía (eléctrica a mecánica, mecánica a calor y eléctrica a calor).

VI. CONCLUSIONES

El objetivo del artículo es encontrar un modelo electro-mecánico para un robot móvil diferencial. La señal de entrada es el voltaje de armadura que se aplica a cada motor CD. El modelo contiene los parámetros eléctricos y mecánicos del robot. Se despreció la inductancia del motor porque suele ser de un valor que puede ser despreciable. La inclusión de los parámetros eléctricos no afectan las propiedades conocidas para la matriz de inercia y la matriz de centrifugas y Coriolis. Se añade una tercera matriz que contiene los fenómenos de disipación y transformación de la energía. Este modelo puede ayudar a diseñar controladores para robots sin servomotores en los actuadores que utilizan. Como trabajo futuro se pretende desarrollar controladores con el modelo presentado y con lo cual se puede demostrar estabilidad considerando los parámetros del motor.

AGRADECIMIENTOS

Este proyecto ha sido parcialmente financiado PFCE 2017 y TecNM.

REFERENCIAS

- [1] G. Campion, G. Bastin, and B. D'Andréa-Novel, "Structural Properties and Classification of Kinematic and Dynamic Models of Wheeled Mobile Robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 47–62, 1996.
- [2] R. S. Ortigoza, M. M. Aranda, G. S. Ortigoza, V. M. H. Guzmán, M. A. M. Vilchis, G. S. González, J. C. H. Lozada, and M. O. Carbajal, "Wheeled Mobile Robots : A Review," *IEEE Latin America Transactions*, vol. 10, no. 6, pp. 2209–2217, 2012.
- [3] J. Batlle and A. Barjau, "Holonomy in mobile robots," *Robotics and Autonomous Systems*, vol. 57, pp. 433–440, apr 2009.
- [4] E. Bugarin and A. Gutierrez, "Modelado Cinemático de un Robot Móvil Omnidireccional no Estándar," in *Congreso Latinoamericano de Control Automático*, (Lima, Perú), 2009.
- [5] E. Bugarin, A. Aguilar-Bustos, and O. A. Borrego-Ramirez, "Validación Experimental del Modelo Cinemático de un Robot Móvil Omnidireccional de 4 Ruedas," in *Congreso Internacional de Robótica y Computación*, pp. 24–29, 2013.
- [6] D. Devon and T. Bretl, "Kinematic and Dynamic Control of a Wheeled Mobile Robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (San Diego, California), pp. 4065–4070, 2007.
- [7] M. W. Spong, S. Hutchinson, and V. M., "Robot Modeling and Control," *Control*, vol. 141, no. 1, p. 419, 2006.
- [8] S. G. Tzafestas, *Introduction to Mobile Robot Control*. Atenas, Grecia: Elsevier Insights, 2013.
- [9] A. Isidori, *Nonlinear Control Systems*. Springer, 1995.
- [10] R. Kelly, V. Santibáñez, and A. Loria, *Control of Robot Manipulator in Joint Space*. Springer, 2005.
- [11] R. Kelly and V. Santibáñez, *Control de movimiento de robots manipuladores*. Pearson Prentice Hall, 2003.

Sistema de Visión para Parametrizar la Trayectoria de Extremidades en Marcha Bípeda

Luis Alberto Pacheco Vega

División de Estudios de Posgrados e Investigación
Instituto Tecnológico de La Paz
La Paz, B.C. S., México
e-mail: pachecomsc16@itlp.edu.mx

José Luis Gómez Torres

División de Estudios de Posgrados e Investigación
Instituto Tecnológico de La Paz
La Paz, B.C. S., México
e-mail: jgomez@itlp.edu.mx

Israel Marcos Santillán Mendez

División de Estudios de Posgrados e Investigación
Instituto Tecnológico de La Paz
La Paz, B.C. S., México
e-mail: israel.santillan@itlp.edu.mx

Saúl Martínez Díaz

División de Estudios de Posgrados e Investigación
Instituto Tecnológico de La Paz
La Paz, B.C. S., México
e-mail: smdiaz06@itlp.edu.mx

Resumen—El presente trabajo consiste en el diseño de un sistema de Visión Artificial para detectar e identificar marcadores, ubicados en las articulaciones de una persona con el fin de parametrizar su comportamiento durante la marcha y obtener patrones con miras a ser implementados en el control de un robot bípedo, buscando que los movimientos sean similares a los del ser humano. Para tal efecto el sistema utiliza etiquetas que proveen una rápida colocación además de un sistema de visión que convierte los puntos detectados en las tres cámaras a coordenadas tridimensionales.

I. INTRODUCCIÓN

Durante años el campo de la robótica se ha centrado en el diseño y desarrollo de autómatas. En la búsqueda de desarrollar robots con la capacidad de realizar tareas de manera autónoma y segura, capaz de asistir a los seres humanos en diferentes ámbitos, derivando en diversos problemas, uno de ellos la bipedestación.

La bipedestación o bipedismo es la capacidad de caminar con postura erguida sobre dos extremidades inferiores. Emulando esta característica en robots con una severa complejidad de inestabilidad al quedar soportado en una sola pierna durante ciertos periodos de tiempo.

Una de las áreas de interés es la asistencia médica a personas que presentan algún tipo de discapacidad motriz. Actualmente existen diferentes dispositivos para que estas personas puedan trasladarse de un lugar a otro, como son las silla de ruedas, muletas, bastones, etc. sin embargo, estas tienen una movilidad limitada en terrenos irregulares o en terrenos regulares con obstáculos como pudiera ser las escaleras.

Algo importante a considerar es la forma en que el sistema interactúa con la persona al realizar la parametrización. Para

tal fin se cuentan con dos técnicas la medición directa y la medición por imágenes.

I-A. Medición Directa

Goniómetro

Un goniómetro es un nombre especial dado al medidor mecánico o eléctrico que se puede conectar para medir un ángulo de articulación [5]. Un brazo del goniómetro está unido a un segmento de la extremidad, el otro al segmento de la extremidad adyacente, y el eje del goniómetro está alineado con el eje de la articulación.

Ventajas:

1. Los valores de salida pueden ser inmediatamente registrados o enviados a una computadora.

Desventajas:

1. Se proporcionan datos angulares relativos, no ángulos absolutos, lo que limita severamente el valor de evaluación de los datos.
2. Pueden requerir un tiempo excesivo para ajustarse y alinearse, y la alineación sobre la grasa y el tejido muscular puede variar a lo largo del movimiento.
3. Si hay necesidad de medir varias articulaciones, el movimiento puede ser obstaculizado por las correas y los cables.

Acelerómetro

Es un dispositivo que mide la aceleración. La mayoría de los acelerómetros no son más que transductores de fuerza diseñados para medir las fuerzas de reacción asociadas con una aceleración dada [5].

Ventajas:

1. Los valores de salida al igual que el goniómetro pueden ser inmediatamente registrados o enviados a una computadora.

Desventajas:

1. La aceleración es relativa a su posición en el segmento de la extremidad.
2. El costo de los acelerómetros puede ser excesivo si se usa un gran número; también el costo de la grabadora o del convertidor analógico a digital puede ser alto.
3. Si se usa un gran número de ellos, pueden obstaculizar el movimiento.
4. Muchos tipos de acelerómetros son bastante sensibles a los golpes.

I-B. Medición por Imágenes

Debido a la complejidad de la mayoría de los movimientos, un sistema de imágenes es el único capaz de capturar todos los datos. Dada la tarea adicional de describir una actividad dinámica, donde el reto mayor es tener que capturar datos durante un periodo prolongado de tiempo. Esto requiere tomar muchas imágenes a intervalos regulares durante la marcha.

Ventajas:

1. Todos los puntos se presentan en un mismo sistema de referencia respecto a la cámara.
2. El sistema no está limitado en cuanto a la cantidad de marcadores utilizados.
3. No existe obstrucción al movimiento para la mayoría de los sistemas que usan marcadores reflectantes livianos, y el tiempo para colocar los marcadores es mínimo.

Desventajas:

1. El tiempo de procesamiento puede ser un problema, por lo que se debe considerar el número de imágenes por segundo capaz de capturar y la resolución.
2. Los marcadores pueden ser obstruidos por otras extremidades como los brazos.

II. MARCHA HUMANA

La marcha humana es un modo de locomoción bípeda con actividad alternada de los miembros inferiores, que se caracteriza por una sucesión de doble apoyo y de apoyo unipodal, es decir que durante la marcha al menos uno de los miembros no deja nunca el suelo [1].

El ciclo de la marcha comienza cuando el pie contacta con el suelo y termina con el siguiente contacto con el suelo del mismo pie. Los dos mayores componentes del ciclo de la marcha son: la fase de apoyo y la fase de balanceo. Una pierna está en fase de apoyo cuando está en contacto con el suelo y está en fase de balanceo cuando no contacta con el suelo.

La cantidad relativa de tiempo transcurrido durante cada fase del ciclo de la marcha, a una velocidad normal como se muestra en la figura 1:

1. Fase de apoyo: 60
2. Fase de balanceo: 40
3. Doble apoyo: 20

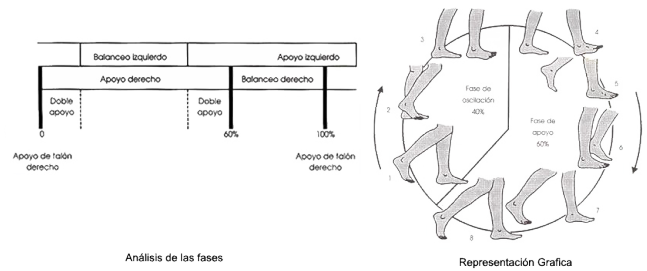


Figura 1. Fases de la caminata.

Si se tiene en cuenta la duración de cada una de las fases, es posible observar que sumando los ciclos que se están produciendo de manera simultánea en ambos miembros inferiores, en algún momento ambos pies se encuentran en contacto con el suelo. A este periodo se le denomina fase de doble apoyo.

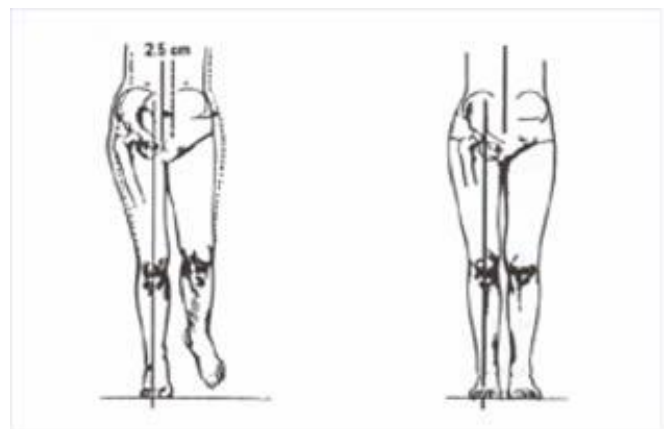


Figura 2. Desplazamiento (vista frontal).

Centro de gravedad: Se consigue un mínimo gasto de energía, cuando el cuerpo se mueve en línea recta sin que el centro de gravedad se desvíe.

Desplazamiento vertical: con un rango de movimiento de aproximado de 5 cm.

Desplazamiento horizontal: con un rango aproximado de movimiento de 5 cm. representado en la figura 2.

III. VISIÓN ARTIFICIAL

La visión artificial es una rama de la inteligencia artificial que tiene por objetivo modelar matemáticamente los procesos de percepción visual en los seres vivos y generar programas que permitan simular estas capacidades visuales por computadora. La visión artificial permite la detección automática de la estructura y propiedades de objetos en 3 dimensiones a partir una o varias imágenes bidimensionales. Las imágenes pueden ser monocromáticas o a color; capturadas por una o varias cámaras, y cada cámara puede ser estacionaria o móvil. La estructura y propiedades del mundo tridimensional

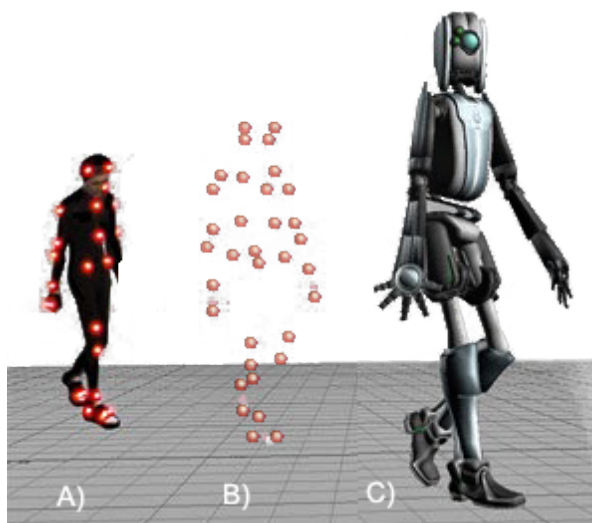


Figura 3. Etapas del modelado: A) Uso de marcadores. B) Detección de marcadores. C) Implementación

que se intentan deducir en la visión artificial incluyen no sólo propiedades geométricas (tamaños, formas, localización de objetos, etc.), sino también propiedades del material (sus colores, sus texturas, la composición, etc.) y la luminosidad u oscuridad de las superficies [2].

Utilizando un sistema de visión artificial es posible generar una representación digital de un objeto del mundo real como se muestra en la figura 3.

IV. ARQUITECTURA DEL SISTEMA

El sistema propuesto para la parametrización consta de marcadores circulares adhesivos de 19 mm de diámetro de un color específico (rojo), posicionados en las articulaciones determinantes en la marcha humana.

Los puntos de colocación para los marcadores propuestos pueden observarse en la figura 4 detectado con ello las coordenadas de los marcadores a intervalos de 200 ms, con la esquina superior derecha de la imagen como su origen. Posteriormente las coordenadas se trasladan a otro sistema coordenado, con origen en la intersección formada por el piso (caminadora) y el eje vertical del cuerpo, como se observa en la figura 5.

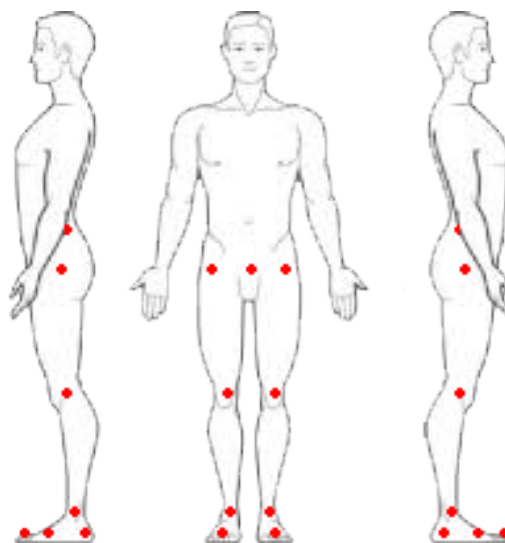


Figura 4. Posición de marcadores.

El sistema de Visión propuesto consta de tres cámaras web marca Logitech modelo C525 HD (1280 x 720 px) a 30 fps. Se propone realizar la caminata sobre una banda eléctrica a una velocidad promedio de 1.38 m/s, con la finalidad de mantener los marcadores en un área acotada del campo visual de las cámaras, fijadas a tres metros de distancia y distribuidas de la siguiente manera: una para el plano frontal y dos para captar los planos laterales de la persona mostrado en la figura 6.



Figura 5. Origen de coordenadas en el sistema.

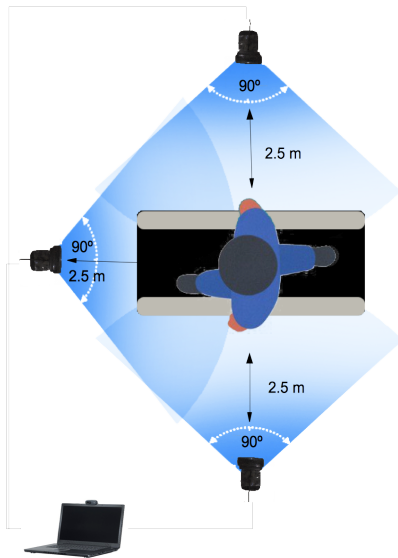


Figura 6. Esquema de captura.

El algoritmo representado en el diagrama de flujo de la Figura 7 fué programado en C++ con el uso de la librería OpenCV para la captura de video y la obtención de coordenadas en 3D de cada articulación. La detección de marcadores (que se realiza de forma paralela en las tres cámaras) se muestra en el Algoritmo 1, buscando sincronizar así la captura de las imágenes.

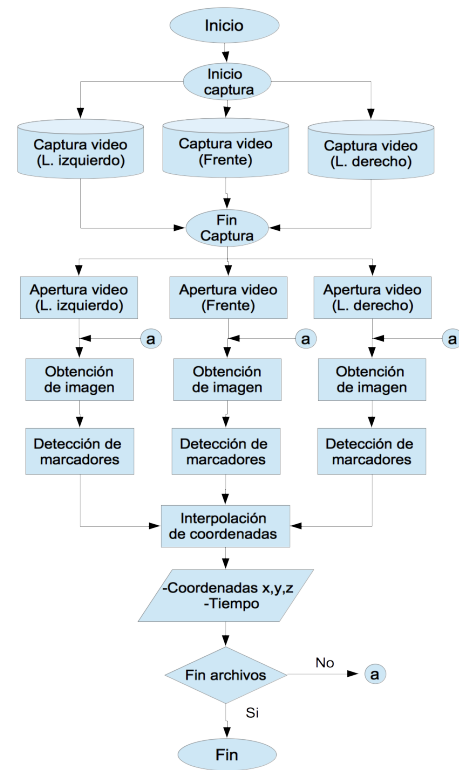


Figura 7. Diagrama de flujo.

Algoritmo 1 .- Detección de marcadores

1. Inversión de los colores de la imagen, el color rojo se convierte en cyan facilitando el rango de selección.
2. Conversión de la imagen del modelo RGB al modelo HSV.
3. Binarización de la imagen con la selección de color de los marcadores (Cyan: HL=70, HH=95, SL=90, SH=255, VL=113, VH=255).
4. Selección de las figuras circulares con área igual a 10.
5. Obtención de las cordenadas de los centroides de cada marcador.

V. RESULTADOS

Al termino y prueba del sistema se obtuvieron imágenes con reconocimiento de 100 % en ambientes controlados donde la intensidad lumínica es uniforme. Siendo afectado en condiciones de poca luz o altos contrastes de luz ambiental, como se muestra en la figura 8 donde se aplica el proceso para seleccionar en condiciones de alto contraste de luz, obteniendo un reconocimiento de marcaores del 85 % y alcanzando el 95 % al compensar la saturación de color(SL=65) .

Obteniendo posteriormente las siluetas redondas y rojas como se muestra en la figura 9, en la cual ademas de los puntos

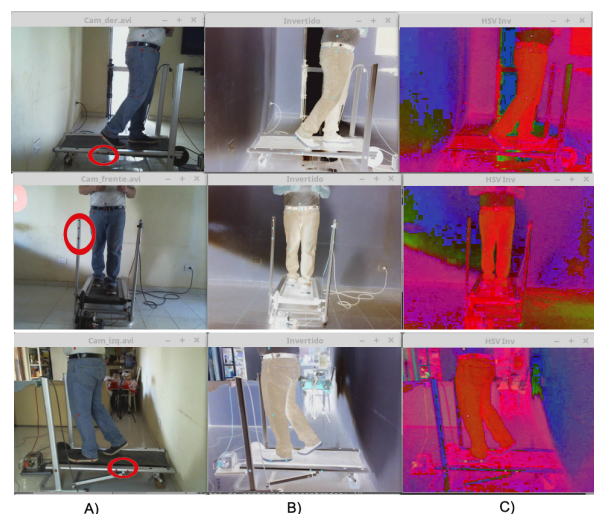


Figura 8. Tratamiento de la imagen: A) Imagen original con puntos de referencia señalados, B) Imagen despues de invertir los colores, C) Conversión de RGB a HSV.

de los marcadores se muestran dos puntos de referencia fijos a 10 cm entre ellos utilizada en la conversión de coordenadas para obtener distancias normalizadas entre imagenes, obteniendo una interpolación mas precisa de las capturas, resultando coordenadas 3D.



Figura 9. Identificación de marcadores: A) Imagen binaria con la detección del color rojo y señalando los puntos de referencia B) Imagen con los centroides de los marcadores y sus coordenadas 2D.

Las coordenadas 3D se forman utilizando como X y Y las coordenadas de la parte frontal y Z se obtiene al sumar los incrementos las X obtenidas de la cámara según el lado que corresponda, por ejemplo para la rodilla de la izquierda se tomaría X y Y del marcador de la rodilla izquierda con la cámara frontal y para Z los incrementos de X obtenidos entre cada imagen del marcador de la rodilla tomada con la cámara de lado izquierdo.

VI. CONCLUSIONES

- El algoritmo cumple con su función de detectar los marcadores, considerando la intensidad de luz ambiental y la velocidad de marcha normal en promedio 1.83 m/s.
- Las pruebas se realizaron en personas de 23 - 27 años en buen estado de salud, como una referencia intermedia a futuro.

VI-A. Trabajo Futuro

- Se buscara optimizar los tiempos de respuesta a fin de poder ser implementado y obtener la parametrización necesaria en tiempo real.
- En trabajo futuro se propone la implementación de redes neuronales (ANFIS) para obtener el modelo de la trayectoria de cada articulación parametrizada.
- Proponiendo realizarlas de igual forma en menores y adultos con el fin de estimar los cambios que sufre el cuerpo humano a través del tiempo a fin de percibir o pronosticar anomalías en pacientes dando con esto un oportuno tratamiento de ser requerido.

AGRADECIMIENTOS

Se agradece al Tecnológico Nacional de México, al Instituto Tecnológico de La Paz y al CONACyT por el apoyo recibido y las facilidades otorgadas para el desarrollo de este trabajo.

REFERENCIAS

- [1] Mayco Ramos, *Desglose de la biomecánica de la marcha humana y algunas patologías*. Santo Domingo, República Dominicana 2016.
- [2] Jorge Estarita, Andrés Jiménez, Jaime Brochero, Hugo Escobar, Silvia Moreno, *Sistema de Reconocimiento de objetos en tiempo real*. Barranquilla, Colombia 2017.
- [3] Richard Hartley, Andrew Zisserman, *Multiple View Geometry in Computer Vision*, 2da ed. Cambridge, United Kingdom 2003.
- [4] Juan Humberto Sossa Azuela, *Visión artificial: Rasgos descriptores para el reconocimiento de objetos*, 2da ed. 2013.
- [5] David A Winter, *Biomechanics and Motor Control of Human Movement*, 4ta ed. Waterloo, Ontario, Canada, 2009.
- [6] Christopher L. Vaughan, Brian L. Davis, Jeremy C. O'Connor, *Dynamics of Human Gait*, 2nd ed. Cape Town, South Africa, 1992.
- [7] Cai, Ziyun; Long, Yang; Shao, Ling; *Adaptive RGB Image Recognition by Visual-Depth Embedding*, Vol. 27 Num. 5. (IEEE Transactions on Image Processing), 2018.
- [8] Carranza, Cesar; Llamocca, Daniel; Pattichis, Marios; *Fast 2D Convolutions and Cross-Correlations Using Scalable Architectures*, Vol. 26 Num. 5. (IEEE Transactions on Image Processing), 2018.
- [9] Lv, Xiaoqi; Jiang, Guangqi; Gu, Yu; *Recognition of a Virtual Scene via Simulated Prosthetic Vision*, Vol. 5. Irlanda (Frontiers in Bioengineering and Biotechnology), 2017.
- [10] Zhang, Yigong; Su, Yingna; Yang, Jian; *When Dijkstra Meets Vanishing Point: A Stereo Vision Approach for Road Detection*, Vol. 27 Num. 5. (IEEE Transactions on Image Processing), 2018.
- [11] Minjie Cai ; Kris M. Kitani ; Yoichi Sato; *An Ego-Vision System for Hand Grasp Analysis*, Vol. 47. (IEEE Transactions on Human-Machine Systems), 2017.
- [12] A. I. Kapandji, *Fisiología Articular*, 6ta ed. Tomo 2. París, Francia, 2012.
- [13] Carlos Alberto Guevara Guillen, Juan Antonio Martínez Chavelas, *Diseño de Sistema de Locomoción Bípida Basada en Marcha Humana*, Instituto Tecnológico de La Paz, La Paz, B. C. S., México, 2016.
- [14] Miguel Angel Saucedo Romero, *Valoración de la Marcha Humana*, Universidad Nacional Autónoma de México, CD. de México, México, 2009.

Análisis dinámico y simulación de un mecanismo para rehabilitación de miembro inferior.

Muñoz-Reina José Saúl.
Centro de Innovación y Desarrollo
Tecnológico en Cómputo
del Instituto Politécnico Nacional.
Email: antares_ars@hotmail.com

Villarreal-Cervantes Miguel Gabriel.
Centro de Innovación y Desarrollo
Tecnológico en Cómputo
del Instituto Politécnico Nacional.
Email: mvillarrealc@ipn.mx

Corona-Ramírez Leonel Germán.
Unidad Profesional Interdisciplinaria
en Ingeniería y Tecnologías Avanzadas
del Instituto Politécnico Nacional.
Email: lcoronaramirez@hotmail.com

Resumen—La rehabilitación proporcionada por mecanismos ayuda a disminuir los tiempos de recuperación y resulta ser una herramienta muy útil al fisioterapeuta. Estos sistemas otorgan al paciente una rutina de entrenamiento adecuada con movimientos repetitivos que preparan a la extremidad afectada a tener un buen desempeño en un trabajo activo. En este trabajo se presenta el modelado cinemático y dinámico de un mecanismo de 8 eslabones, con el cual es posible obtener trayectorias similares a las presentes en el tobillo durante la marcha, y servirá como medio para rehabilitar los miembros inferiores.

I. INTRODUCCIÓN

En México viven cerca de 16 millones de personas que presentan dificultades para realizar sus actividades como son caminar y subir o bajar escaleras. La discapacidad motora es una de las discapacidades más frecuentes a nivel nacional y a nivel mundial [1].

Las patologías más comunes que se presentan en los miembros inferiores son fracturas, artrosis e insuficiencia venosa, debidas a factores como son accidentes de tránsito, lesiones deportivas, envejecimiento, enfermedades degenerativas y estilo de vida.

Una alternativa a la recuperación motriz es incluir al paciente en un programa de rehabilitación a partir del diagnóstico de un fisioterapeuta. En una terapia convencional el fisioterapeuta apoya al paciente a realizar movimientos corporales repetitivos, con el fin de mejorar la función músculo-esquelética del miembro afectado.

La rehabilitación proporcionada por sistemas mecánicos especializados resultan ser una herramienta de apoyo al fisioterapeuta en la rutina de entrenamiento, debido a que ejecutan movimientos repetitivos similares a los presentes en un trabajo activo. Usualmente estos sistemas son empleados en la etapa de rehabilitación pasiva, es decir, en la etapa en la que el paciente realiza movimientos no voluntarios, sin presentar algún tipo de resistencia en la zona a tratar [2].

Los sistemas de rehabilitación de miembros inferiores se pueden clasificar de acuerdo a su principio de funcionamiento [3], como caminadoras entrenadoras de marcha, sistemas con pie-plato-base, formadores de marcha, sistemas de rehabilitación de tobillo y formadores de marcha estacionarios, siendo estos últimos los sistemas de menor costo debido a que cuentan con pocos grados de libertad y no requieren sistemas

complejos de control, pero se ven limitados a realizar rutinas de ejercicio simples .

Con el fin de diseñar un sistema de rehabilitación que otorgue una rutina de entrenamiento a bajo costo se han propuesto mecanismos formadores de marcha estacionarios de un grado de libertad, como los propuestos por E. E. Rosero [4] para rehabilitación de rodilla a través de un mecanismo de balancín-biela-corredera y el sistema de C. H. Guzman [5] para la rehabilitación de tobillo a través de un sistema de corredera actuada. Sin embargo estos sistemas se ven limitados a realizar rutinas simples de flexión y extensión de un área en específico.

Con el objetivo de otorgarle al paciente una rutina que prepare a la extremidad afectada a un mejor trabajo activo, se han propuesto mecanismos que ayudan a simular el movimiento de la extremidad inferior durante la marcha, como el mecanismo de cuatro eslabones de Z. Ji [6], los mecanismos de 6 y 10 eslabones propuestos por B. Y. Tsuge [7][8] y el mecanismo Stephenson III de seis barras de Y. Shao [9]. Estos mecanismos se diseñan para ejercicios en el plano parasagital y se obtienen a través de la síntesis dimensional del mecanismo, para simular la trayectoria presente en el tobillo durante la marcha.

Desde un punto de vista de diseño mecatrónico, para el diseño de un mecanismo no solo basta con enfocarse en la síntesis dimensional, sino también es necesario encontrar el compromiso entre el sistema de control y el diseño mecánico, ya que a partir de un diseño mecánico adecuado es posible establecer leyes de control poco complejas que otorguen un rendimiento adecuado al sistema [10].

En este trabajo se presenta el modelado cinemático y dinámico para un mecanismo de 8 eslabones [11], el cual se tomará como modelo conceptual para realizar un sistema de rehabilitación para miembros inferiores. La importancia de obtener el modelado cinemático y dinámico radica en que a partir de éstos es posible realizar un diseño concurrente, para satisfacer ciertos requerimientos, como son, la síntesis dimensional del mecanismo, su morfología estructural y el sistema de control.

II. ANÁLISIS CINEMÁTICO.

En este trabajo se presentan las ecuaciones cinemáticas que se requerirán en el proceso de análisis dinámico, las cuales

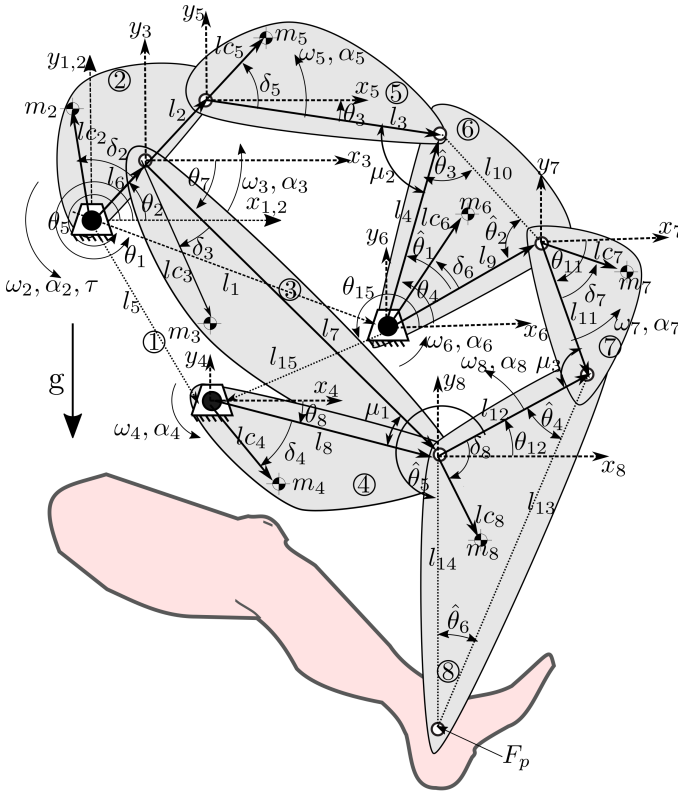


Figura 1. Mecanismo de 8 eslabones.

se obtuvieron a partir del método de análisis vectorial para el mecanismo de 8 eslabones [12] (Ver Figura 1).

El análisis vectorial del mecanismo de la Figura 1 consiste en seccionarlo en 3 submecanismos más simples, resultando 2 submecanismos de 4 barras conformados por $M_1 = \{\vec{l}_1, \vec{l}_2, \vec{l}_3, \vec{l}_4\}$ y $M_2 = \{\vec{l}_5, \vec{l}_6, \vec{l}_7, \vec{l}_8\}$ y un submecanismo de 5 barras conformado por $M_3 = \{\vec{l}_8, \vec{l}_9, \vec{l}_{11}, \vec{l}_{12}, \vec{l}_{15}\}$. Por lo tanto las ecuaciones de lazo vectorial que describen a los 3 submecanismos son:

Submecanismo M_1 :

$$l_2 e^{j\theta_2} + l_3 e^{j\theta_3} - l_4 e^{j\theta_4} - l_1 e^{j\theta_1} = 0 \quad (1)$$

Submecanismo M_2 :

$$l_6 e^{j\theta_6} + l_7 e^{j\theta_7} - l_5 e^{j\theta_5} - l_8 e^{j\theta_8} = 0 \quad (2)$$

Submecanismo M_3 :

$$l_9 e^{j\theta_9} + l_{11} e^{j\theta_{11}} - l_{12} e^{j\theta_{12}} - l_8 e^{j\theta_8} - l_{15} e^{j\theta_{15}} = 0 \quad (3)$$

Del mecanismo de la Figura 1 se desconocen las orientaciones $\theta_3, \theta_4, \theta_6, \theta_7, \theta_8, \theta_9, \theta_{11}$ y θ_{12} . Estas incógnitas se pueden obtener a partir de las ecuaciones de lazo vectorial (1)-(3) separando parte real y parte imaginaria y resolviendo el sistema de ecuaciones resultantes. Las ecuaciones obtenidas de este proceso de análisis son las siguientes:

$$\theta_6 = \theta_2 \quad (4)$$

$$\theta_9 = \theta_4 - \hat{\theta}_1 \quad (5)$$

$$\theta_{j+k} = 2 \operatorname{atan2} \left(\frac{-\hat{B} + (-1)^{k+1} \sqrt{\hat{B}^2 + \hat{A}^2 - \hat{C}^2}}{\hat{C} - \hat{A}} \right) \quad (6)$$

$\forall j = 0, 4 \wedge k = 3, 4$

$$\theta_{o+11} = 2 \operatorname{atan2} \left(\frac{-\tilde{B} + (-1)^o \sqrt{\tilde{B}^2 + \tilde{A}^2 - \tilde{C}^2}}{\tilde{C} - \tilde{A}} \right) \quad (7)$$

$\forall o = 0, 1$

donde:

$$\hat{A} = (-1)^k 2l_{j+1}l_{j+k} \cos \theta_{j+1} + (-1)^{k+1} 2l_{j+2}l_{j+k} \cos \theta_2$$

$$\hat{B} = (-1)^k 2l_{j+1}l_{j+k} \sin \theta_{j+1} + (-1)^{k+1} 2l_{j+2}l_{j+k} \sin \theta_2$$

$$\hat{C} = l_{j+1}^2 + l_{j+2}^2 + (-1)^{k+1} l_{j+3}^2 + (-1)^k l_{j+4}^2 - 2l_{j+1}l_{j+2} \cos(\theta_{j+1} - \theta_2)$$

$$\tilde{A} = (-1)^{o+1} 2l_{o+11}l_{8(1-o)+15o} \cos \theta_{8(1-o)+15o} + (-1)^o 2l_9 l_{o+11} \cos \theta_9$$

$$+ (-1)^{o+1} 2l_{o+11}l_{8o+15(1-o)} \cos \theta_{8o+15(1-o)}$$

$$\tilde{B} = (-1)^{o+1} 2l_{o+11}l_{8(1-o)+15o} \sin \theta_{8(1-o)+15o} + (-1)^o 2l_9 l_{o+11} \sin \theta_9$$

$$+ (-1)^{o+1} 2l_{o+11}l_{8o+15(1-o)} \cos \theta_{8o+15(1-o)}$$

$$\tilde{C} = l_8^2 + l_9^2 + l_{o+11}^2 + l_{15}^2 - l_{12-o}^2 + (-1)^{o+1} 2l_8 l_{9(1-o)+15o} \cos(\theta_8 - \theta_{9(1-o)+15o}) + (-1)^o 2l_8 l_{9o+15(1-o)} \cos(\theta_8 - \theta_{9o+15(1-o)}) - 2l_9 l_{15} \cos(\theta_9 - \theta_{15})$$

Un punto importante del análisis de posición, es determinar la posición del punto donde se genera la trayectoria deseada (punto F_p), para esto es necesario conocer la orientación vector de posición \vec{l}_{14} , la cual está determinada por:

$$\theta_{14} = \theta_{12} + \hat{\theta}_5 \quad (8)$$

En el análisis de velocidades del mecanismo de 8 eslabones, se desconocen las velocidades angulares $\omega_3, \omega_4, \omega_5, \omega_6, \omega_7$ y ω_8 . Para obtener estas incógnitas se deriva con respecto al tiempo las ecuaciones de lazo vectorial (1)-(3), y separando parte real y parte imaginaria se obtiene un sistema de ecuaciones, que al resolverlo resultan (9)-(12).

$$\omega_3 = -\frac{l_6 \omega_2 \operatorname{sen}(\theta_2 - \theta_8)}{l_7 \operatorname{sen}(\theta_7 - \theta_8)}; \omega_4 = -\frac{l_6 \omega_2 \operatorname{sen}(\theta_2 - \theta_7)}{l_6 \operatorname{sen}(\theta_7 - \theta_8)} \quad (9)$$

$$\omega_5 = -\frac{l_2 \omega_2 \operatorname{sen}(\theta_2 - \theta_4)}{l_3 \operatorname{sen}(\theta_3 - \theta_4)}; \omega_6 = -\frac{l_2 \omega_2 \operatorname{sen}(\theta_2 - \theta_3)}{l_4 \operatorname{sen}(\theta_3 - \theta_4)} \quad (10)$$

$$\omega_7 = -\frac{l_9 \omega_6 \operatorname{sen}(\theta_9 - \theta_{12}) - l_8 \omega_4 \operatorname{sen}(\theta_8 - \theta_{12})}{l_{11} \operatorname{sen}(\theta_{11} - \theta_{12})} \quad (11)$$

$$\omega_8 = -\frac{l_9 \omega_6 \operatorname{sen}(\theta_9 - \theta_{11}) - l_8 \omega_4 \operatorname{sen}(\theta_8 - \theta_{11})}{l_{12} \operatorname{sen}(\theta_{11} - \theta_{12})} \quad (12)$$

Para el análisis de aceleraciones del mecanismo, se desconocen las aceleraciones angulares $\alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7$ y α_8 . Estas incógnitas se obtienen al realizar la segunda derivada con respecto al tiempo de las ecuaciones de lazo vectorial (1)-(3), tras separar parte real y parte imaginaria se obtiene un sistema de ecuaciones, que al resolverlo resultan (13)-(18).

$$\alpha_3 = -\frac{l_6 \cos(\theta_2 - \theta_8) \omega_2^2 + l_7 \cos(\theta_7 - \theta_8) \omega_3^2}{l_7 \operatorname{sen}(\theta_7 - \theta_8)} - \frac{-l_8 \omega_4^2 + \alpha_2 l_6 \operatorname{sen}(\theta_2 - \theta_8)}{l_7 \operatorname{sen}(\theta_7 - \theta_8)} \quad (13)$$

$$\alpha_4 = -\frac{l_6 \cos(\theta_2 - \theta_7) \omega_2^2 - l_8 \cos(\theta_7 - \theta_8) \omega_4^2}{l_8 \sin(\theta_7 - \theta_8)} - \frac{l_7 \omega_3^2 + \alpha_2 l_6 \sin(\theta_2 - \theta_7)}{l_8 \sin(\theta_7 - \theta_8)} \quad (14)$$

$$\alpha_5 = -\frac{l_2 \cos(\theta_2 - \theta_4) \omega_2^2 + l_3 \cos(\theta_3 - \theta_4) \omega_5^2}{l_3 \sin(\theta_3 - \theta_4)} - \frac{-l_4 \omega_6^2 + \alpha_2 l_2 \sin(\theta_2 - \theta_4)}{l_3 \sin(\theta_3 - \theta_4)} \quad (15)$$

$$\alpha_6 = -\frac{l_2 \cos(\theta_2 - \theta_3) \omega_2^2 - l_4 \cos(\theta_3 - \theta_4) \omega_6^2}{l_4 \sin(\theta_3 - \theta_4)} - \frac{l_3 \omega_5^2 + \alpha_2 l_2 \sin(\theta_2 - \theta_3)}{l_4 \sin(\theta_3 - \theta_4)} \quad (16)$$

$$\alpha_7 = \frac{l_8 \cos(\theta_8 - \theta_{12}) \omega_4^2 - l_9 \cos(\theta_9 - \theta_{12}) \omega_6^2}{l_{11} \sin(\theta_{11} - \theta_{12})} + \frac{\alpha_4 l_8 \sin(\theta_8 - \theta_{12}) - \alpha_6 l_9 \sin(\theta_9 - \theta_{12})}{l_{11} \sin(\theta_{11} - \theta_{12})} + \frac{l_{12} \omega_8^2 - l_{11} \cos(\theta_{11} - \theta_{12}) \omega_7^2}{l_{11} \sin(\theta_{11} - \theta_{12})} \quad (17)$$

$$\alpha_8 = \frac{l_8 \cos(\theta_8 - \theta_{11}) \omega_4^2 - l_9 \cos(\theta_9 - \theta_{11}) \omega_6^2}{l_{12} \sin(\theta_{11} - \theta_{12})} + \frac{\alpha_4 l_8 \sin(\theta_8 - \theta_{11}) - \alpha_6 l_9 \sin(\theta_9 - \theta_{11})}{l_{12} \sin(\theta_{11} - \theta_{12})} + \frac{-l_{11} \omega_7^2 + l_{12} \cos(\theta_{11} - \theta_{12}) \omega_8^2}{l_{12} \sin(\theta_{11} - \theta_{12})} \quad (18)$$

III. ANÁLISIS DINÁMICO

III-A. Análisis de aceleraciones lineales.

En esta sección se obtendrán las aceleraciones lineales que actúan en el centro de masa de los eslabones móviles, para obtenerlas, deben de ser calculadas a partir del sistema de coordenadas base (x_1, y_1) hacia el centro de masa del eslabón en estudio.

Los vectores de posición de los centros de masa se muestran en la Figura 2, los cuales se definen por medio de la longitud del vector de posición l_{c_i} y su orientación δ_i respecto al eje de referencia propio de cada eslabón (x_i, y_i) para todo $i = \{2, 3, \dots, 8\}$.

Para obtener el vector de aceleración en el centro de masa definido desde el sistema de coordenadas base, se calcula el vector de aceleración medido del eje de referencia propio del eslabón (x_i, y_i) hacia su centro de masa, posteriormente se calcula el vector de aceleraciones medido desde el eje de referencia (x_1, y_1) hacia el eje de referencia del eslabón (x_i, y_i) , obtenidos ambos vectores de aceleración se realiza la suma vectorial de ambos para obtener las aceleraciones requeridas.

En las ecuaciones (19)-(25) se describen los vectores de aceleración de los centros de masa, expresados por la notación $a_{G_{ij}}$, en donde i es el número del eslabón en estudio y j es la componente del vector de aceleración en el eje x ó y .

Eslabón 2:

$$\begin{aligned} a_{G_{2x}} &= -l_2(\sin(\theta_2 + \delta_2)\alpha_2 + \cos(\theta_2 + \delta_2)\omega_2^2) \\ a_{G_{2y}} &= l_2(\cos(\theta_2 + \delta_2)\alpha_2 - \sin(\theta_2 + \delta_2)\omega_2^2) \end{aligned} \quad (19)$$

Eslabón 3:

$$\begin{aligned} a_{G_{3x}} &= -l_6(\sin\theta_2\alpha_2 + \cos\theta_2\omega_2^2) - \\ &lc_3(\sin(\theta_7 + \delta_3)\alpha_3 + \cos(\theta_7 + \delta_3)\omega_3^2) \\ a_{G_{3y}} &= l_6(\cos\theta_2\alpha_2 - \sin\theta_2\omega_2^2) + \\ &lc_3(\cos(\theta_7 + \delta_3)\alpha_3 - \sin(\theta_7 + \delta_3)\omega_3^2) \end{aligned} \quad (20)$$

Eslabón 4:

$$\begin{aligned} a_{G_{4x}} &= -lc_4(\sin(\theta_8 + \delta_4)\alpha_4 + \cos(\theta_8 + \delta_4)\omega_4^2) \\ a_{G_{4y}} &= lc_4(\cos(\theta_8 + \delta_4)\alpha_4 - \sin(\theta_8 + \delta_4)\omega_4^2) \end{aligned} \quad (21)$$

Eslabón 5:

$$\begin{aligned} a_{G_{5x}} &= -l_2(\sin\theta_2\alpha_2 + \cos\theta_2\omega_2^2) - \\ &lc_5(\sin(\theta_3 + \delta_5)\alpha_5 + \cos(\theta_3 + \delta_5)\omega_5^2) \\ a_{G_{5y}} &= l_2(\cos\theta_2\alpha_2 - \sin\theta_2\omega_2^2) + \\ &lc_5(\cos(\theta_3 + \delta_5)\alpha_5 - \sin(\theta_3 + \delta_5)\omega_5^2) \end{aligned} \quad (22)$$

Eslabón 6:

$$\begin{aligned} a_{G_{6x}} &= -lc_6(\sin(\theta_9 + \delta_6)\alpha_6 + \cos(\theta_9 + \delta_6)\omega_6^2) \\ a_{G_{6y}} &= lc_6(\cos(\theta_9 + \delta_6)\alpha_6 - \sin(\theta_9 + \delta_6)\omega_6^2) \end{aligned} \quad (23)$$

Eslabón 7:

$$\begin{aligned} a_{G_{7x}} &= -l_9(\sin\theta_9\alpha_6 + \cos\theta_9\omega_6^2) - \\ &lc_7(\sin(\theta_{11} + \delta_7)\alpha_7 + \cos(\theta_{11} + \delta_7)\omega_7^2) \\ a_{G_{7y}} &= l_9(\cos\theta_9\alpha_6 - \sin\theta_9\omega_6^2) + \\ &lc_7(\cos(\theta_{11} + \delta_7)\alpha_7 - \sin(\theta_{11} + \delta_7)\omega_7^2) \end{aligned} \quad (24)$$

Eslabón 8:

$$\begin{aligned} a_{G_{8x}} &= -l_8(\sin\theta_8\alpha_4 + \cos\theta_8\omega_4^2) \\ &- lc_8(\sin(\theta_{12} + \delta_8)\alpha_8 + \cos(\theta_{12} + \delta_8)\omega_8^2) \\ a_{G_{8y}} &= l_8(\cos\theta_8\alpha_4 - \sin\theta_8\omega_4^2) \\ &+ lc_8(\cos(\theta_{12} + \delta_8)\alpha_8 - \sin(\theta_{12} + \delta_8)\omega_8^2) \end{aligned} \quad (25)$$

III-B. Análisis por el método de Newton.

En este apartado se analizará la dinámica del mecanismo de 8 eslabones de la Figura 1 por el método Newton para el análisis de fuerzas y pares como se describe en [13], se emplea este método debido a que proporciona información sobre pares y fuerzas en las juntas de pasador de los eslabones, y a través de estos datos es posible realizar un análisis de esfuerzos, con el fin de evitar fallas en los eslabones.

Para un problema dinámico bidimensional por el método de Newton, se deben de satisfacer las ecuaciones (26), en donde F_j es la fuerza ejercida en las juntas de pasador de los eslabones, M_i es el momento torsional medidos desde el centro de masa del eslabón, m_i es la masa del eslabón, $a_{G_{ij}}$ es la aceleración lineal del centro de masa del eslabón medido desde el centro de coordenadas base, I_{G_i} es el tensor de inercia sobre el eje z medido desde el centro de masa del eslabón, α_i es la aceleración angular y g es la aceleración producida por la gravedad de la Tierra.

$$\sum F_x = m_i \cdot a_{G_{ix}} \quad \sum F_y = m_i \cdot (a_{G_{iy}} + g) \quad \sum M_i = I_{G_i} \cdot \alpha_i \quad (26)$$

Para el análisis de fuerzas y pares se emplea el diagrama de cuerpo libre de la Figura 2, en el cual se realiza el desacoplamiento de los elementos y se analizan las fuerzas de forma específica en cada eslabón. Por lo tanto las ecuaciones de fuerzas y momentos que definen al mecanismo en estudio son las siguientes:

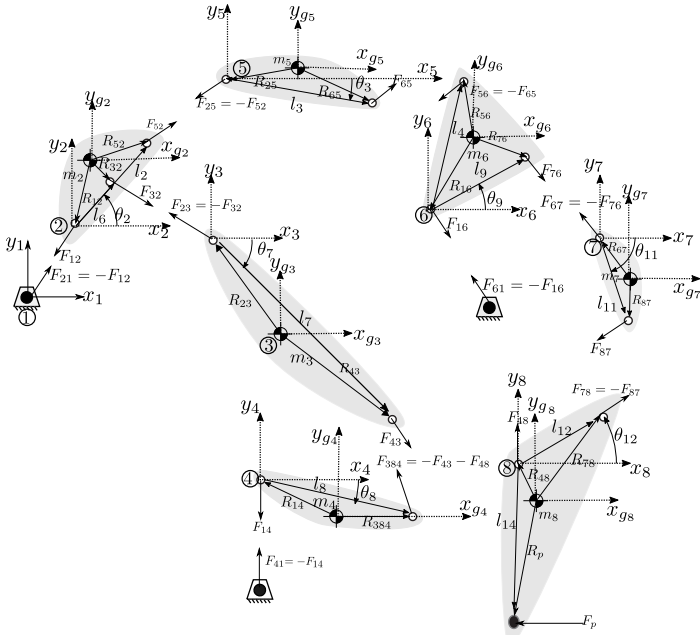


Figura 2. Mecanismo desacoplado para análisis dinámico.

Eslabón 2:

$$\begin{aligned}
 F_{12_x} + F_{32_x} + F_{52_x} &= m_2 a_{G_{2x}} \\
 F_{12_y} + F_{32_y} + F_{52_y} &= m_2 a_{G_{2y}} + m_2 g \\
 \tau + (R_{12_x} F_{12_y} - R_{12_y} F_{12_x}) + (R_{32_x} F_{32_y} - R_{32_y} F_{32_x}) + \\
 (R_{52_x} F_{52_y} - R_{52_y} F_{52_x}) &= I_{2_G} \alpha_2
 \end{aligned} \quad (27)$$

Eslabón 3:

$$\begin{aligned}
 F_{43_x} - F_{32_x} &= m_3 a_{G_{3x}} \\
 F_{43_y} - F_{32_y} &= m_3 a_{G_{3y}} + m_3 g \\
 (R_{43_x} F_{43_y} - R_{43_y} F_{43_x}) - (R_{23_x} F_{32_y} - R_{23_y} F_{32_x}) &= I_{3_G} \alpha_7
 \end{aligned} \quad (28)$$

Eslabón 4:

$$\begin{aligned}
 F_{14_x} - F_{43_x} - F_{48_x} &= m_4 a_{G_{4x}} \\
 F_{14_y} - F_{43_y} - F_{48_y} &= m_4 a_{G_{4y}} + m_4 g \\
 (R_{14_x} F_{14_y} - R_{14_y} F_{14_x}) - [R_{384_x} (F_{43_y} + F_{48_y}) - \\
 R_{384_y} (F_{43_x} + F_{48_x})] &= I_{4_G} \alpha_8
 \end{aligned} \quad (29)$$

Eslabón 5:

$$\begin{aligned}
 F_{65_x} - F_{52_x} &= m_5 a_{G_{5x}} \\
 F_{65_y} - F_{52_y} &= m_5 a_{G_{5y}} + m_5 g \\
 (R_{65_x} F_{65_y} - R_{65_y} F_{65_x}) - (R_{25_x} F_{52_y} - R_{25_y} F_{52_x}) &= I_{5_G} \alpha_3
 \end{aligned} \quad (30)$$

Eslabón 6:

$$\begin{aligned}
 F_{16_x} + F_{76_x} - F_{65_x} &= m_6 a_{G_{6x}} \\
 F_{16_y} + F_{76_y} - F_{65_y} &= m_6 a_{G_{6y}} + m_6 g \\
 (R_{16_x} F_{16_y} - R_{16_y} F_{16_x}) + (R_{76_x} F_{76_y} - R_{76_y} F_{76_x}) - \\
 (R_{56_x} F_{65_y} - R_{56_y} F_{65_x}) &= I_{6_G} \alpha_4
 \end{aligned} \quad (31)$$

Eslabón 7:

$$\begin{aligned}
 F_{87_x} - F_{76_x} &= m_7 a_{G_{7x}} \\
 F_{87_y} - F_{76_y} &= m_7 a_{G_{7y}} + m_7 g \\
 (R_{87_x} F_{87_y} - R_{87_y} F_{87_x}) - (R_{67_x} F_{76_y} - R_{67_y} F_{76_x}) &= I_{7_G} \alpha_{11}
 \end{aligned} \quad (32)$$

Eslabón 8:

$$\begin{aligned}
 F_{p_x} + F_{48_x} - F_{87_x} &= m_8 a_{G_{8x}} \\
 F_{p_y} + F_{48_y} - F_{87_y} &= m_8 a_{G_{8y}} + m_8 g \\
 (R_{p_x} F_{p_y} - R_{p_y} F_{p_x}) + (R_{48_x} F_{48_y} - R_{48_y} F_{48_x}) - \\
 (R_{78_x} F_{87_y} - R_{78_y} F_{87_x}) &= I_{8_G} \alpha_{12}
 \end{aligned} \quad (33)$$

En donde, los vectores de posición denotados por R_{kij} están definidos por las ecuaciones (34)-(40) considerando a i como el número del eslabón al que pertenece el vector, k es el número del eslabón con el que se conecta la junta de pasador y j es la componente del vector sobre el eje x o y .

Eslabón 2:

$$\begin{aligned}
 R_{12_x} &= -l_2 \cos(\theta_2 + \delta_2) \\
 R_{12_y} &= -l_2 \sin(\theta_2 + \delta_2) \\
 R_{32_x} &= l_6 \cos \theta_2 - l_2 \cos(\theta_2 + \delta_2) \\
 R_{32_y} &= l_6 \sin \theta_2 - l_2 \sin(\theta_2 + \delta_2) \\
 R_{52_x} &= l_2 \cos \theta_2 - l_2 \cos(\theta_2 + \delta_2) \\
 R_{52_y} &= l_2 \sin \theta_2 - l_2 \sin(\theta_2 + \delta_2)
 \end{aligned} \quad (34)$$

Eslabón 3:

$$\begin{aligned}
 R_{23_x} &= -l_3 \cos(\theta_7 + \delta_3) \\
 R_{23_y} &= -l_3 \sin(\theta_7 + \delta_3) \\
 R_{43_x} &= l_7 \cos \theta_7 - l_3 \cos(\theta_7 + \delta_3) \\
 R_{43_y} &= l_7 \sin \theta_7 - l_3 \sin(\theta_7 + \delta_3)
 \end{aligned} \quad (35)$$

Eslabón 4:

$$\begin{aligned}
 R_{14_x} &= -l_4 \cos(\theta_8 + \delta_4) \\
 R_{14_y} &= -l_4 \sin(\theta_8 + \delta_4) \\
 R_{384_x} &= l_8 \cos \theta_8 - l_4 \cos(\theta_8 + \delta_4) \\
 R_{384_y} &= l_8 \sin \theta_8 - l_4 \sin(\theta_8 + \delta_4)
 \end{aligned} \quad (36)$$

Eslabón 5:

$$\begin{aligned}
 R_{25_x} &= -l_5 \cos(\theta_3 + \delta_5) \\
 R_{25_y} &= -l_5 \sin(\theta_3 + \delta_5) \\
 R_{65_x} &= l_3 \cos \theta_3 - l_5 \cos(\theta_3 + \delta_5) \\
 R_{65_y} &= l_3 \sin \theta_3 - l_5 \sin(\theta_3 + \delta_5)
 \end{aligned} \quad (37)$$

Eslabón 6:

$$\begin{aligned}
 R_{16_x} &= -l_6 \cos(\theta_9 + \delta_6) \\
 R_{16_y} &= -l_6 \sin(\theta_9 + \delta_6) \\
 R_{56_x} &= l_4 \cos \theta_4 - l_6 \cos(\theta_9 + \delta_6) \\
 R_{56_y} &= l_4 \sin \theta_4 - l_6 \sin(\theta_9 + \delta_6) \\
 R_{76_x} &= l_9 \cos \theta_9 - l_6 \cos(\theta_9 + \delta_6) \\
 R_{76_y} &= l_9 \sin \theta_9 - l_6 \sin(\theta_9 + \delta_6)
 \end{aligned} \quad (38)$$

Eslabón 7:

$$\begin{aligned}
 R_{67_x} &= -l_7 \cos(\theta_{11} + \delta_7) \\
 R_{67_y} &= -l_7 \sin(\theta_{11} + \delta_7) \\
 R_{87_x} &= l_{11} \cos \theta_{11} - l_7 \cos(\theta_{11} + \delta_7) \\
 R_{87_y} &= l_{11} \sin \theta_{11} - l_7 \sin(\theta_{11} + \delta_7)
 \end{aligned} \quad (39)$$

Eslabón 8:

$$\begin{aligned}
 R_{48_x} &= -l_8 \cos(\theta_{12} + \delta_8) \\
 R_{48_y} &= -l_8 \sin(\theta_{12} + \delta_8) \\
 R_{78_x} &= l_{12} \cos \theta_{12} - l_8 \cos(\theta_{12} + \delta_8) \\
 R_{78_y} &= l_{12} \sin \theta_{12} - l_8 \sin(\theta_{12} + \delta_8) \\
 R_{p_x} &= l_{14} \cos \theta_{14} - l_8 \cos(\theta_{12} + \delta_8) \\
 R_{p_y} &= l_{14} \sin \theta_{14} - l_8 \sin(\theta_{12} + \delta_8)
 \end{aligned} \quad (40)$$

Agrupando las ecuaciones (27)-(33) es posible obtener el modelo dinámico directo, y expresar la evolución del sistema en función de las fuerzas y los pares que intervienen. Para este caso el modelo dinámico directo está expresado como un sistema de ecuaciones lineal de la forma $AX + B = 0$ siendo X el vector de incógnitas como se muestra en (41).

$$X = [F_{12_x} \quad F_{12_y} \quad F_{32_x} \quad F_{32_y} \quad F_{52_x} \quad F_{52_y} \quad F_{43_x} \quad \dots \quad F_{43_y} \quad F_{14_x} \quad F_{14_y} \quad F_{65_x} \quad F_{65_y} \quad F_{16_x} \quad F_{16_y} \quad \dots \quad F_{76_x} \quad F_{76_y} \quad F_{87_x} \quad F_{87_y} \quad F_{48_x} \quad F_{48_y} \quad \tau]^T \quad (41)$$

Como se observa en (41), las variables que se obtienen por el modelado dinámico por el método de Newton son el par motor τ y las fuerzas internas de los eslabones. Es importante mencionar que la fuerza en el punto F_p no es una incógnita del sistema ya que es una fuerza que se conoce. Para este caso en particular representa la fuerza ejercida en el punto F_p por el miembro inferior de una persona.

IV. RESULTADOS.

El proceso de validación del modelo obtenido se divide en dos experimentos, en los cuales se realiza un análisis comparativo entre los resultados obtenidos por el modelo analítico y los obtenidos por un programa de diseño en CAD. Para cada experimento se consideran distintos parámetros estructurales y algunas condiciones generales como se mencionan a continuación.

Las condiciones generales son:

- La velocidad angular de la manivela ω_2 es constante y definida como $\omega_2 = 2 \cdot \pi \text{ rad/s}$, por lo tanto la aceleración angular se define como $\alpha_2 = 0 \text{ rad/s}^2$ y la posición angular como $\theta_2 = 2 \cdot \pi \cdot t \text{ rad}$, considerando a t como el tiempo.
- Para el análisis comparativo se emplea el programa Matlab® para la simulación con el modelado analítico y el programa Solidworks® para simular el diseño en CAD a través de su herramienta de análisis de movimiento.
- El tiempo de simulación para ambos programas es de $t \in [0, 1]s$ y se considera una frecuencia de muestreo de $1KHz$.
- Para el experimento 1 se considera una fuerza en el efector constante en el eje y y definida por $F_{p_x} = 0 \text{ N}$, $F_{p_y} = -100 \text{ N}$, y para el experimento 2 se considera una fuerza con componentes en el eje x y y definida por $F_{p_x} = 5 \text{ N}$, $F_{p_y} = -98,1 \text{ N}$

Como condiciones particulares se consideran para cada experimento los parámetros cinemáticos mostrados en la Tabla I y Tabla II y como parámetros dinámicos los datos de la Tabla III y Tabla IV, ambos parámetros cinemáticos y dinámicos representan a los mecanismos que se muestran en la Figura 3.

En las Figura 4 se muestran las gráficas comparativas del par motor obtenidas por la simulación del modelo analítico y la simulación en Solidworks®. En estas gráficas es posible realizar una comparativa visual de resultados, en la cual se observa que las curvas presentan trayectorias iguales. Por lo tanto es posible concluir que el modelo analítico es correcto,

debido a que la gráfica del par motor, se considera el resultado más importante, ya que es resultado de la combinación de las fuerzas y momentos que actúan sobre todo el mecanismo, y en dado caso que alguna fuerza o momento fuera incorrecto se observaría un diferencia significativa entre ambas gráficas.

Tabla I
PARÁMETROS CINEMÁTICOS DE SIMULACIÓN PARA EL EXPERIMENTO 1

| Parámetro | Valor | Parámetro | Valor |
|------------------|------------|------------------|------------|
| l_1 | 0.2000m | l_2 | 0.0800m |
| l_3 | 0.1684m | l_4 | 0.1344m |
| l_5 | 0.1726m | l_6 | 0.0235m |
| l_7 | 0.0933m | l_8 | 0.1997m |
| l_9 | 0.1861m | l_{11} | 0.1867m |
| l_{12} | 0.1788m | l_{14} | 0.2000m |
| θ_1 | 5.7711rad | θ_5 | 3.3934 rad |
| $\hat{\theta}_1$ | 1.0434 rad | $\hat{\theta}_5$ | 5.1757 rad |

Tabla II
PARÁMETROS CINEMÁTICOS DE SIMULACIÓN PARA EL EXPERIMENTO 2

| Parámetro | Valor | Parámetro | Valor |
|------------------|------------|------------------|------------|
| l_1 | 0.6524 m | l_2 | 0.2603 m |
| l_3 | 0.4346 m | l_4 | 0.5519 m |
| l_5 | 0.7499 m | l_6 | 0.1649 m |
| l_7 | 0.4873 m | l_8 | 0.5178 m |
| l_9 | 0.4212 m | l_{11} | 0.5186 m |
| l_{12} | 0.5714 m | l_{14} | 0.2402 m |
| θ_1 | 1.0887 rad | θ_5 | 2.4037 rad |
| $\hat{\theta}_1$ | 6.0517 rad | $\hat{\theta}_5$ | 0.1620 rad |

Tabla III
PARÁMETROS DINÁMICOS DE SIMULACIÓN PARA EL EXPERIMENTO 1

| Eslabón | Masa [kg] | I_G [$kg \cdot m^2$] | l_c [m] | δ [rad] |
|---------|-----------|--------------------------|-----------|----------------|
| 2 | 0.0207 | 1.4280E-5 | 0.0407 | 0.0614 |
| 3 | 0.0246 | 2.1370E-5 | 0.0467 | -0.0495 |
| 4 | 0.0667 | 2.0168E-4 | 0.1000 | 0.0691 |
| 5 | 0.0436 | 1.0564E-4 | 0.0842 | 0.0304 |
| 6 | 0.1574 | 4.9156E-4 | 0.0937 | 0.4142 |
| 7 | 0.0481 | 1.4116E-4 | 0.0933 | 0.0271 |
| 8 | 0.2186 | 9.0526E-4 | 0.1078 | -0.5931 |

Tabla IV
PARÁMETROS DINÁMICOS DE SIMULACIÓN PARA EL EXPERIMENTO 2

| Eslabón | Masa [kg] | I_G [$kg \cdot m^2$] | l_c [m] | δ [rad] |
|---------|-----------|--------------------------|-----------|----------------|
| 2 | 0.1617 | 0.0017 | 0.1066 | -0.0010 |
| 3 | 0.2706 | 0.0075 | 0.2633 | 0.0009 |
| 4 | 3.9712 | 0.1608 | 0.2963 | 0.0459 |
| 5 | 0.0074 | 0.0001 | 0.2174 | 3.6152E-5 |
| 6 | 1.2906 | 0.0847 | 0.1503 | -0.4391 |
| 7 | 0.2626 | 0.0071 | 0.2585 | -0.0002 |
| 8 | 0.6418 | 0.0407 | 0.1248 | 0.1300 |

Otra forma de concluir que el modelo es correcto es por medio de cuantificar el parentesco de los resultados obtenidos por las simulaciones, para ésto se calcula el error entre datos a través de la raíz del error cuadrático medio (RMSE) y la desviación estándar del error (STD). Los resultados de este proceso para cada experimento se muestran en la Tabla V y Tabla VI.

De acuerdo a los resultados obtenidos, se concluye que el modelo matemático es correcto ya que los resultados de RMSE y STD del error no son mayores a valores de $8.22E-2$ lo que significa que ambos resultados convergen a la misma solución.

Se atribuye que el error presente en las mediciones se debe, a que el programa Solidworks® emplea un método numérico GSTIFF para el análisis del mecanismo.

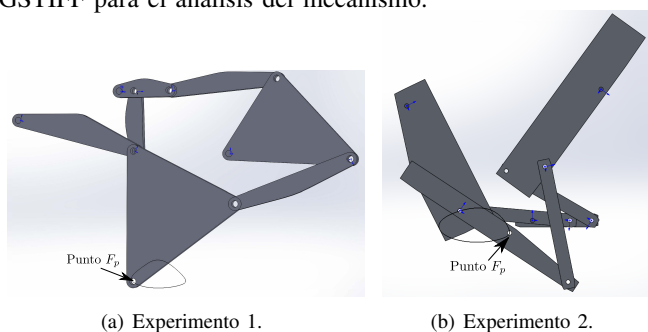


Figura 3. Representación en CAD del mecanismo para los experimentos.

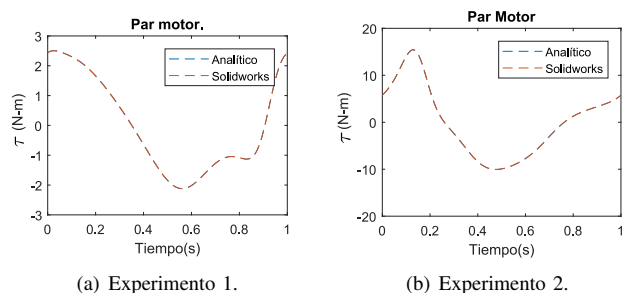


Figura 4. Comportamiento del par motor.

Tabla V
ESTADÍSTICA DESCRIPTIVA DEL EXPERIMENTO 1

| Variable | RMSE | STD | Variable | RMSE | STD |
|-----------|----------|----------|-----------|----------|----------|
| τ | 4.77E-03 | 4.11E-03 | F_{65x} | 1.76E-02 | 1.69E-02 |
| F_{12x} | 8.16E-02 | 7.70E-02 | F_{65y} | 9.13E-03 | 8.76E-03 |
| F_{12y} | 4.76E-02 | 4.03E-02 | F_{16x} | 4.77E-02 | 4.44E-02 |
| F_{32x} | 7.57E-02 | 7.24E-02 | F_{16y} | 4.58E-02 | 3.80E-02 |
| F_{32y} | 4.28E-02 | 3.62E-02 | F_{76x} | 3.50E-02 | 3.28E-02 |
| F_{52x} | 1.76E-02 | 1.69E-02 | F_{76y} | 4.22E-02 | 3.53E-02 |
| F_{52y} | 8.95E-03 | 8.59E-03 | F_{87x} | 3.51E-02 | 3.29E-02 |
| F_{43x} | 7.57E-02 | 7.24E-02 | F_{87y} | 4.23E-02 | 3.53E-02 |
| F_{43y} | 4.29E-02 | 3.62E-02 | F_{48x} | 3.54E-02 | 3.31E-02 |
| F_{14x} | 8.06E-02 | 8.01E-02 | F_{48y} | 4.19E-02 | 3.51E-02 |
| F_{14y} | 1.69E-02 | 1.69E-02 | | | |

Tabla VI
ESTADÍSTICA DESCRIPTIVA DEL EXPERIMENTO 2.

| Variable | RMSE | STD | Variable | RMSE | STD |
|-----------|----------|----------|-----------|----------|----------|
| τ | 4.22E-02 | 4.18E-02 | F_{65x} | 1.38E-01 | 1.37E-01 |
| F_{12x} | 1.03E-01 | 1.02E-01 | F_{65y} | 3.77E-02 | 3.66E-02 |
| F_{12y} | 5.99E-02 | 5.82E-02 | F_{16x} | 1.09E-01 | 1.08E-01 |
| F_{32x} | 4.57E-02 | 4.33E-02 | F_{16y} | 8.22E-02 | 8.07E-02 |
| F_{32y} | 2.77E-02 | 2.73E-02 | F_{76x} | 4.84E-02 | 4.81E-02 |
| F_{52x} | 1.38E-01 | 1.37E-01 | F_{76y} | 5.33E-02 | 5.25E-02 |
| F_{52y} | 4.17E-02 | 4.07E-02 | F_{87x} | 4.57E-02 | 4.54E-02 |
| F_{43x} | 4.76E-02 | 4.48E-02 | F_{87y} | 3.78E-02 | 3.68E-02 |
| F_{43y} | 2.85E-02 | 2.83E-02 | F_{48x} | 4.06E-02 | 4.05E-02 |
| F_{14x} | 5.71E-02 | 5.24E-02 | F_{48y} | 3.12E-02 | 2.92E-02 |
| F_{14y} | 4.77E-02 | 3.99E-02 | | | |

V. CONCLUSIONES.

En el presente trabajo se presenta el modelo cinemático y dinámico para un mecanismo de 8 eslabones, y se ha verificado que el modelo matemático es correcto por medio

de simulaciones, y a partir del análisis de resultados de forma estadística y gráfica.

Como trabajo futuro con los modelos cinemáticos y dinámicos obtenidos se diseñará un sistema de rehabilitación a partir del planteamiento de un problema de optimización, ya que por medio de esta metodología de diseño es posible realizar de forma concurrente la síntesis de mecanismos y la minimización del par motor con la finalidad de facilitar el control.

Una de las ventajas significativas que nos ofrece el método de análisis de Newton es que otorga información sobre las fuerzas en las juntas de pasador del mecanismo. Con estos datos es posible incluir restricciones de esfuerzos en el problema de optimización, con la finalidad de obtener diseños de eslabones que no fallen a esfuerzos de tracción o compresión.

AGRADECIMIENTOS

Este trabajo fue apoyado por la Secretaría de Investigación y Posgrado del Instituto Politécnico Nacional(SIP-IPN) bajo los proyectos SIP-20181312 y SIP-20180196

REFERENCIAS

- [1] O.M.S., “Informe mundial sobre la discapacidad,” http://www.who.int/disabilities/world_report/2011/summary_es.pdf?ua=1, 2011, [Online; accessed 19-December-2017].
- [2] L. Brosseau, S. Milne, G. Wells, P. Tugwell, V. Robinson, L. Casimiro, L. Pelland, M.-J. Noel, J. Davis, and H. Drouin, “Efficacy of continuous passive motion following total knee arthroplasty: a metaanalysis.” *The journal of rheumatology*, vol. 31, no. 11, pp. 2251–2264, 2004.
- [3] I. Díaz, J. J. Gil, and E. Sánchez, “Lower-limb robotic rehabilitation: literature review and challenges,” *Journal of Robotics*, vol. 2011, 2011.
- [4] E. E. Rosero, R. Martínez, E. Galvis *et al.*, “diseño y construcción de una máquina de movimiento pasivo continuo para la terapia de rodilla,” *Ingeniería y Competitividad*, vol. 3, no. 2, pp. 56–64, 2011.
- [5] C. H. Guzmán Valdivia, J. L. Carrera Escobedo, A. Blanco Ortega, M. A. Oliver Salazar, and F. A. Gómez Becerra, “Diseño y control de un sistema interactivo para la rehabilitación de tobillo: Tobibot,” *Ingeniería mecánica, tecnología y desarrollo*, vol. 5, no. 1, pp. 255–264, 2014.
- [6] Z. Ji and Y. Manna, “Synthesis of a pattern generation mechanism for gait rehabilitation,” *Journal of Medical Devices*, vol. 2, no. 3, p. 031004, 2008.
- [7] B. Y. Tsuge, M. M. Plecnik, and J. M. McCarthy, “Homotopy directed optimization to design a six-bar linkage for a lower limb with a natural ankle trajectory,” *Journal of Mechanisms and Robotics*, vol. 8, no. 6, p. 061009, 2016.
- [8] B. Y. Tsuge and J. M. McCarthy, “Synthesis of a 10-bar linkage to guide the gait cycle of the human leg,” in *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2015, pp. 1–7.
- [9] Y. Shao, Z. Xiang, H. Liu, and L. Li, “Conceptual design and dimensional synthesis of cam-linkage mechanisms for gait rehabilitation,” *Mechanism and Machine Theory*, vol. 104, pp. 31–42, 2016.
- [10] Q. Li, W. Zhang, and L. Chen, “Design for control-a concurrent engineering approach for mechatronic systems design,” *IEEE/ASME transactions on mechatronics*, vol. 6, no. 2, pp. 161–169, 2001.
- [11] J. Pantoja-García, M. Villarreal-Cervantes, J. González-Robles, and G. S. Cervantes, “Síntesis óptima de un mecanismo para la marcha bípeda utilizando evolución diferencial,” *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería*, vol. 33, no. 1, pp. 138 – 153, 2017.
- [12] J. Muñoz-Reina, L. Corona-Ramírez, and M. Villarreal-Cervantes, “Análisis cinemático y simulación de un mecanismo para rehabilitación de miembro inferior,” *XXIII Congreso Internacional Anual de la SOMIM(Diseño Mecánico)*, 2017.
- [13] R. L. Norton, *Diseño de maquinaria, Síntesis y análisis de máquinas y mecanismos*, 2016.

Desarrollo de una GUI para la sintonización de ganancias y simulación de esquemas de control adaptable

Ricardo Falcón

Tecnológico Nacional de México,
Instituto Tecnológico de la Laguna,
Blvd. Revolución y Av. Instituto
Tecnológico de la Laguna, S/N,
Torreón, Coahuila, México.
Email: ricardofnp@gmail.com

Víctor Santibáñez

Tecnológico Nacional de México,
Instituto Tecnológico de la Laguna,
Blvd. Revolución y Av. Instituto
Tecnológico de la Laguna, S/N,
Torreón, Coahuila, México
Email: vsantiba@itlaguna.edu.mx

Javier Ollervides

Tecnológico Nacional de México,
Instituto Tecnológico de la Laguna
Blvd. Revolución y Av. Instituto
Tecnológico de la Laguna, S/N,
Torreón, Coahuila, México.

Email: ejollervidesv@correo.itlaguna.edu.mx

Resumen—En este artículo se presenta un ambiente computacional desarrollado en MATLAB® y Simulink que permite evaluar y visualizar el desempeño de diversas leyes de control para varios sistemas mecatrónicos. Debido a que la sintonización en este tipo de controladores suele resultar complicada, el poder modificar diferentes parámetros de la ley de control y visualizar el cambio provocado en el sistema en cuestión resulta de gran utilidad. En este artículo se propone una Interfaz Gráfica de Usuario (GUI) hecha en MATLAB®, la cual muestra a manera de gráficas bidimensionales los resultados de diversas variables del sistema en cuestión. Se utilizan cinco controladores diferentes, todos ellos adaptables, para el control de robots manipuladores completamente actuados de dos g.d.l. A partir de los resultados de dichas gráficas, se pueden obtener interesantes conclusiones que ayuden a sintonizar adecuadamente el sistema.

I. INTRODUCCIÓN

Una interfaz de usuario (UI por sus siglas en inglés) es una forma gráfica de mostrar, en una o más ventanas, qué componentes contienen los controladores, que permiten al usuario desarrollar actividades interactivas. El usuario no tiene que crear un `script`, ni teclear comandos en la ventana de comandos para llevar a cabo las actividades. A diferencia de programar un `script` o introducir comandos en la ventana de comandos, el usuario de la GUI no tiene necesidad de entender los detalles de cómo se desarrollan las tareas.

Los componentes de una UI pueden incluir menús, barras de herramientas, “push buttons”, “list boxes” y controles deslizantes, entre otros. Las UIs creadas utilizando herramientas de MATLAB® pueden llevar a cabo cualquier tipo de cómputo, leer y escribir en archivos de datos, comunicarse con otras UIs, y mostrar datos como tablas o gráficas.

Una UI de MATLAB® es una ventana “figure” a la cual se le añaden componentes utilizados por el usuario. Estos componentes se pueden seleccionar, modificar su tamaño, y modificar su posición como sea mejor. Utilizando “callbacks” se puede hacer que estos componentes hagan lo que se desea cuando el usuario cliquee o manipula los componentes al teclear.

El presente artículo está organizado de la siguiente manera. En la sección II se explican dos metodologías diferentes para la elaboración de una interfaz gráfica de usuario (GUI), donde se describen algunas propiedades fundamentales para el desarrollo de ésta, así como los componentes y las propiedades más importantes a tomar en cuenta. En la sección III se explica cómo utilizar la GUI para sintonizar las ganancias de manera heurística y obtener así el mejor desempeño de las leyes de control. Finalmente, en la sección IV se presentan resultados de los errores de posición de estas leyes de control con las ganancias obtenidas gracias a la GUI.

II. DESARROLLO DE UNA GUI

Se pueden crear UIs de MATLAB® de dos maneras diferentes:

- Crear la UI utilizando GUIDE con el comando `guide`. Mediante este enfoque se empieza con una figura a la que se le ingresan componentes a partir de un editor de gráficos. GUIDE crea un código que contiene los “callbacks” para la UI y sus componentes. GUIDE guarda la ventana “figure” (como un archivo `.fig`) y el archivo del código. La GUI se puede correr a partir de cualquiera de los dos archivos mencionados.
- Crear la UI mediante programación. Con este enfoque se puede crear un archivo de código que define las propiedades y comportamientos de todos los componentes. Cuando el usuario ejecuta el archivo, éste crea la figura, los componentes de ésta, y maneja las interacciones con el usuario. Típicamente, la figura no es guardada entre sesiones debido a que el código en el archivo crea una nueva figura cada vez que es compilado.

Los archivos generados por ambos enfoques son diferentes. Los archivos de UIs creadas mediante programación, generalmente son más largos, debido a que explícitamente definen cada propiedad de la figura y de sus controles, así como sus “callbacks”. Las UIs creadas utilizando GUIDE definen la mayoría de las propiedades dentro de la figura por sí sola.

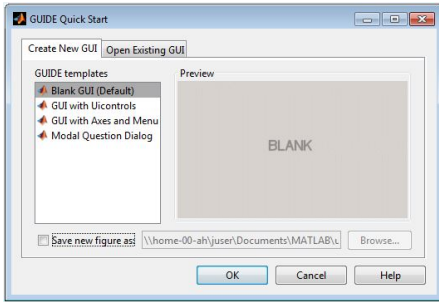


Figura 1: Cuadro de diálogo del GUIDE Quick Start.

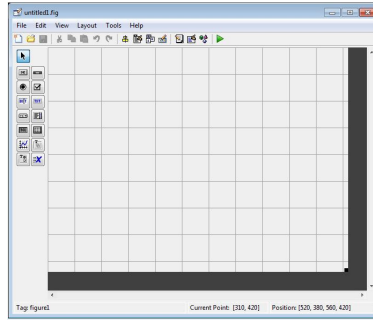


Figura 2: Entorno de desarrollo de GUI.

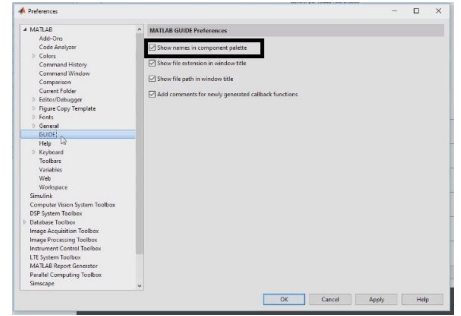


Figura 3: Ventana de «Preferencias».

Una UI puede ser creada utilizando GUIDE y después modificarla mediante programación. Sin embargo, no se puede crear una UI por programación y después modificarla con GUIDE.

II-A. Abrir una UI nueva en el editor GUIDE

1. Inicializar GUIDE tecleando el comando `guide` en la entrada de MATLAB®.
2. En el cuadro de diálogo del GUIDE Quick Start (Figura 1), seleccionar la plantilla **Blank GUI (Default)**, y entonces presionar **OK**.
3. Para desplegar los nombres de los componentes en el entorno de desarrollo GUI (Figura 2) en la paleta de componentes, seguir los siguientes pasos:
 - a) Seleccionar **File > Preferences > GUIDE**.
 - b) Seleccionar **Show names in component palette** en la ventana de «Preferencias» (Figura 3).
 - c) Hacer clic en **OK** (ver Figura 4).

II-B. Texto estático

En la UI, el texto estático cumple la función de una etiqueta que muestra información e instrucciones de cada controlador. Al igual que con los demás componentes, se abre el inspector de propiedades, y en la propiedad “String” hacer clic en el

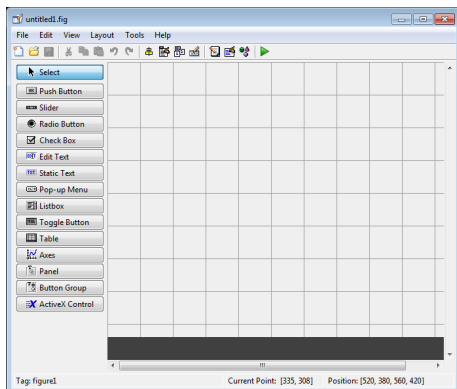


Figura 4: Entorno de desarrollo de GUI con los nombres de los íconos en la paleta de componentes.

icono con tres líneas horizontales (ver Figura 5). En este icono se pueden agregar cantidades de texto cuya longitud excede a la de un renglón.

II-C. Configurar el tamaño de la ventana en el GUIDE

Se configura el tamaño de la ventana de la UI ajustando el área cuadrículada en el “Layout Editor”. Haga clic en la esquina inferior derecha y arrástrelo hasta que la cuadrícula sea aproximadamente 3 pulgadas de alto y 4 pulgadas de ancho (ver Figura 6). Si es necesario, haga la cuadrícula más grande.

II-D. Uso de los componentes

Para la interfaz se utilizaron lo que son “Push Buttons”, “Radio Buttons”, “Edit Texts”, “Static Texts”, y “Panels”.

Al dar doble clic en los componentes, seleccione el valor en la propiedad “String”, y reemplace el texto con la frase deseada.

Haciendo clic fuera de la propiedad “String” o haciendo “Enter”, éste cambiará, tomando el nombre especificado.

II-E. Agregar código

Para agregar código y programar las funciones deseadas al hacer uso de cada componente, hay que ir a cada uno de los componentes y hacer clic derecho en ellos al mover el cursor sobre la opción **View Callbacks > Callback**. Para saber cómo hacer referencia a cada propiedad de cada componente de la UI, abriendo el inspector de propiedades, se puede ver el valor de la propiedad “Tag”, el cual puede ser editado, como el diseñador de la UI lo desee (ver Figura 9).

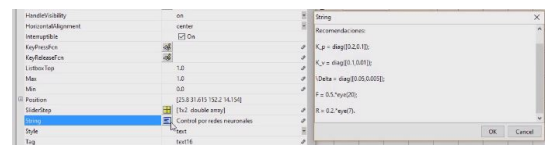


Figura 5: Propiedad “String” para agregar texto estático.

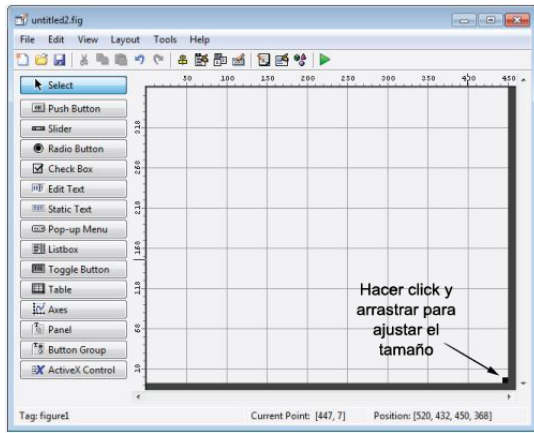


Figura 6: Área cuadriculada en el “Layout Editor”.

II-F. Editor de MATLAB®

Al hacer clic en el “callback” de un componente, se puede agregar código que se ejecutará cuando se activen los eventos indicados.

El código que se agregó a cada radiobutton muestra los componentes relacionados con el controlador seleccionado y oculta los componentes que no tienen relevancia para este controlador.

Después de borrar un componente en la GUIDE, todos los “callbacks” que tenía permanecen en el código fuente. Si se tiene la seguridad de que ningún otro componente utiliza dichos “callbacks”, entonces estos “callbacks” pueden ser removidos del código fuente manualmente. Para más detalles consultar [MATLAB®, 2015].

Después del trabajo de diseño de la interfaz y la programación de los componentes, se obtiene una interfaz como se ve en la Figura 10.

Al compilar el archivo .fig, la GUI se verá como en la Figura 11. Los campos “Stop Time” y “Fixed Step” permiten modificar los tiempos de simulación y el paso de integración de cada archivo de Simulink.

Para obtener las simulaciones para un controlador dado hay que seleccionar el controlador deseado desde el conjunto de los “Radio Buttons”, como se muestra en la Figura 12 .

Después, se da clic en la opción de «Cargar parámetros» para rellenar las ganancias, las condiciones iniciales, los puntos

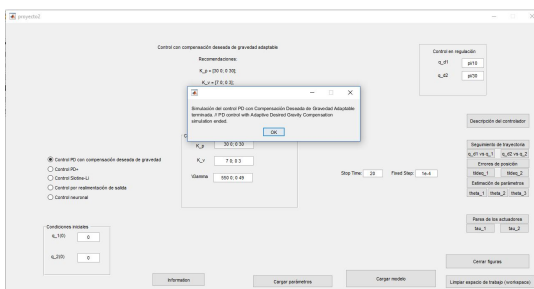


Figura 7: Selección de un controlador a simular en la GUI.

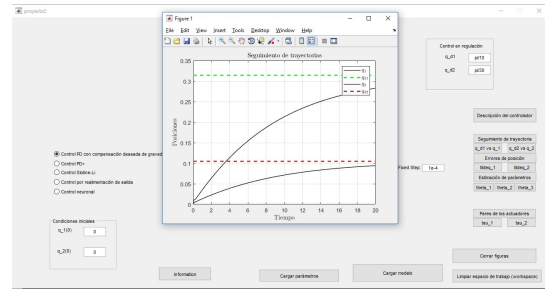


Figura 8: Resultados obtenidos a través de la GUI.

de posiciones deseadas (para el controlador de regulación) y los parámetros “Stop Time” y “Fixed Step” de Simulink.

Al presionar el botón «Cargar modelo» da inicio la simulación del controlador seleccionado con las ganancias y los parámetros ingresados en los campos de texto estático. Al finalizar la simulación aparece una ventana emergente con la leyenda “Simulación del control [nombre del controlador seleccionado] terminada. // [controlador seleccionado en inglés] simulation ended.” (ver Figura 7).

Al terminar la simulación, se puede seleccionar la gráfica a mostrar a partir del conjunto de botones en el lado derecho de la interfaz, y la gráfica deseada se muestra en una ventana “figure” de MATLAB® (ver Figura 8).

III. SINTONIZACIÓN DE CONTROLADORES ADAPTABLES

Después del trabajo de diseñar la GUI, se procede a la sintonización de las ganancias de las diferentes leyes de control para las que la GUI fue diseñada. Aquí se puede seleccionar uno de los cinco controladores analizados en este trabajo, los cuales son:

- Controlador PD con compensación deseada de gravedad adaptable [Kelly *et al.*, 2006, Chap. 15].
- Controlador PD+ adaptable [Lozano y Taoutaou, 2002].
- Controlador con compensación adaptable [Kelly *et al.*, 2006, Chap. 16].
- Controlador con realimentación de salida adaptable [Moreno-Valenzuela *et al.*, 2010].
- Controlador por redes neuronales adaptable [Puga-Guzmán *et al.*, 2014].

III-A. Controlador PD con compensación deseada de gravedad

Se empezó el trabajo con la ley de control PD con compensación deseada de gravedad adaptable, cuya ley de control

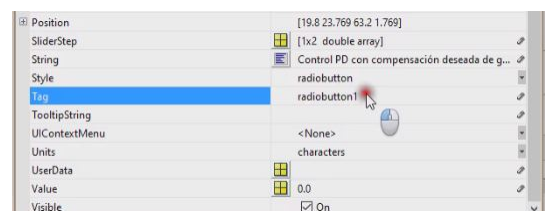


Figura 9: Propiedad “tag” para un componente.

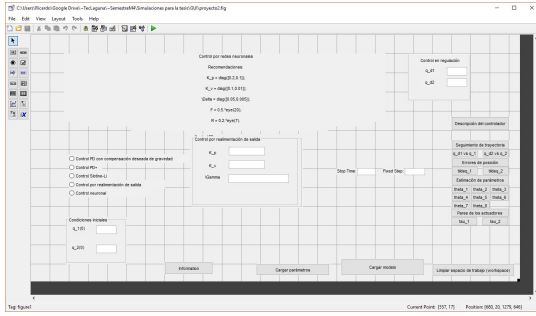


Figura 10: Versión final de la GUI.

está dada por:

$$\tau = K_p \tilde{q} - K_v \dot{q} + g(q_d) \quad (1)$$

donde se impone la restricción de que el vector q_d de posiciones articulares deseadas sea constante. Sin embargo, al ser una ley de control adaptable, la ecuación (1) es reescrita como

$$\tau = K_p \tilde{q} - K_v \dot{q} + \Phi_g(q_d) \hat{\theta} + g_0(q_d) \quad (2)$$

donde τ es el vector de los pares en los actuadores para llevar al robot manipulador de dos g.d.l. a las posiciones deseadas dadas por el vector q_d ; la matriz K_p y la matriz K_v representan a las matrices de ganancias proporcionales y derivativas, respectivamente; el vector q representa a las posiciones articulares y el vector \dot{q} representa a las velocidades articulares; $\Phi_g(q_d)$ es la matriz de regresión utilizada para los parámetros a adaptar y $\hat{\theta}$ es el vector de parámetros adaptados.

La estructura de los sistemas de control adaptables para robots manipuladores está definida por una ley adaptable comúnmente utilizada en los sistemas adaptables continuos [Kelly *et al.*, 2006], dada por:

$$\hat{\theta}(t) = \Gamma \int_0^t \Psi(s, q, \dot{q}, \ddot{q}, q_d, \dot{q}_d, \ddot{q}_d) ds + \hat{\theta}(0) \quad (3)$$

donde $\Gamma = \Gamma^T \in \mathbb{R}^{m \times m}$ representa a la ganancia de adaptación, $\hat{\theta}(0)$ son los parámetros iniciales de diseño y $\Psi \in \mathbb{R}^m$ es una función vectorial a determinar; por lo que la ley adaptable queda definida por [Falcón *et al.*, 2017]:

$$\hat{\theta}(t) = \Gamma \Phi_g^T(q_d) \int_0^t [\gamma \tanh(\tilde{q}) - \dot{q}] ds + \hat{\theta}(0) \quad (4)$$

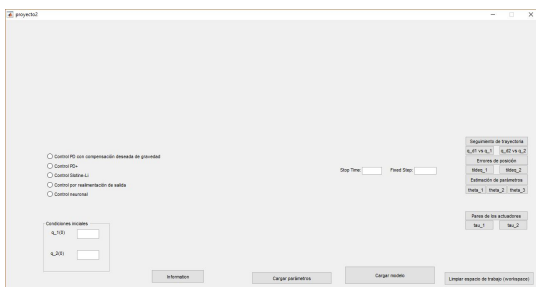


Figura 11: Aspecto de la GUI al ejecutarse.

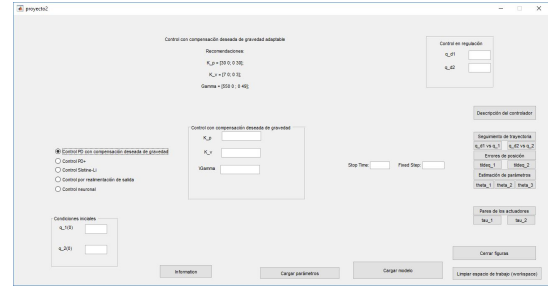


Figura 12: Selección del controlador PD con compensación deseada de gravedad adaptable para simular en la GUI.

Las ganancias seleccionadas para este controlador con la GUI quedaron definidas por

$$K_p = \text{diag}\{30, 30\} \text{ N} \cdot \text{m/rad},$$

$$K_v = \text{diag}\{7, 3\} \text{ N} \cdot \text{m} \cdot \text{s/rad}.$$

y la matriz de ganancias de los parámetros adaptables definida como $\Gamma = \text{diag}\{550, 49\}$.

A partir de la simulación de este controlador se obtuvo la gráfica de los errores de posición (Figura 13) y las adaptaciones paramétricas (Figura 14) llevadas a cabo por la ecuación (4) para un tiempo de 100 s y un paso de integración de 1×10^{-4} s, con $q_d = [\frac{\pi}{10} \quad \frac{\pi}{30}]^T$ rad y condiciones iniciales como $q(0) = [0 \quad 0]^T$ rad.

III-B. Controlador PD+ adaptable

El controlador PD+ para el seguimiento de posiciones deseadas está dado por la siguiente ecuación:

$$\tau = K_p \tilde{q} + K_v \dot{q} + M(q) \ddot{q}_d + C(q, \dot{q}) \dot{q}_d + g(q) \quad (5)$$

La versión adaptable de este controlador está dado por:

$$\tau = K_p \tilde{q} + K_v \dot{q} + \Phi \theta + M_0(q) \ddot{q}_d + C_0(q, \dot{q}) \dot{q}_d + g_0(q) \quad (6)$$

donde $K_p, K_v \in \mathbb{R}^{n \times n}$; y M_0, C_0 y g_0 representan las partes conocidas de la matriz de inercias, la matriz de Coriolis, y el vector de pares gravitacionales sin los parámetros a adaptar por la ley adaptable. La ley adaptable está dada por:

$$\hat{\theta}(t) = \Gamma \int_0^t \Phi^T [\gamma \tanh(\tilde{q}(\sigma)) + \dot{q}(\sigma)] d\sigma + \hat{\theta}(0). \quad (7)$$

Las mejores ganancias encontradas a prueba y error mediante el uso de la GUI fueron

$$K_p = \text{diag}\{2000, 1000\} \text{ N} \cdot \text{m/rad},$$

$$K_v = \text{diag}\{150, 15\} \text{ N} \cdot \text{m} \cdot \text{s/rad},$$

y la matriz de ganancias de los parámetros adaptables como $\Gamma = \text{diag}\{10, 0.19, 0.09\}$.

Utilizando la GUI se obtienen gráficas de los errores de posición articulares (Figura 15) y las adaptaciones paramétricas (Figura 16), en un tiempo de 50 s, un paso de integración de 1×10^{-4} s, y trayectorias a seguir dadas por

$$\begin{bmatrix} q_{d1}(t) \\ q_{d2}(t) \end{bmatrix} = \begin{bmatrix} 45(1 - e^{-2t^3}) + 10(1 - e^{-2t^3}) \sin(4t) \\ 60(1 - e^{-\frac{9}{5}t^3}) + 125(1 - e^{-\frac{9}{5}t^3}) \sin(3t) \end{bmatrix},$$

con condiciones iniciales $\mathbf{q}(0) = [0 \ 0]^T \in \mathbb{R}^2$.

III-C. Controlador con compensación adaptable (Slotine–Li)

El controlador Slotine–Li se utiliza para el seguimiento de posiciones deseadas en robots manipuladores de n g.d.l., y su ley de control está dada por:

$$\boldsymbol{\tau} = K_p \tilde{\mathbf{q}} + K_v \dot{\tilde{\mathbf{q}}} + M(\mathbf{q}) [\ddot{\mathbf{q}}_d + \Lambda \dot{\tilde{\mathbf{q}}}] + C(\mathbf{q}, \dot{\mathbf{q}}) [\dot{\mathbf{q}}_d + \Lambda \tilde{\mathbf{q}}] + \mathbf{g}(\mathbf{q}) \quad (8)$$

donde $K_p, K_v \in \mathbb{R}^{n \times n}$ son matrices simétricas definidas positivas, y $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$ representan a las posiciones, velocidades, y aceleraciones articulares, respectivamente, y la matriz Λ es el resultado del producto de dos matrices simétricas definidas positivas $\Lambda = K_v^{-1} K_p$. La versión adaptable queda definida como

$$\boldsymbol{\tau} = K_p \tilde{\mathbf{q}} + K_v \dot{\tilde{\mathbf{q}}} + \Phi \hat{\boldsymbol{\theta}} + M_0 [\ddot{\mathbf{q}}_d + \Lambda \dot{\tilde{\mathbf{q}}}] + C_0 [\dot{\mathbf{q}}_d + \Lambda \tilde{\mathbf{q}}] + \mathbf{g}_0(\mathbf{q}) \quad (9)$$

donde M_0, C_0 y \mathbf{g}_0 representan las partes conocidas de la matriz $M(\mathbf{q})$, $C(\mathbf{q}, \dot{\mathbf{q}})$ y el vector $\mathbf{g}_0(\mathbf{q})$ sin los términos que involucran a los parámetros dinámicos a estimar por la ley adaptable. Ésta está dada por

$$\hat{\boldsymbol{\theta}}(t) = \Gamma \int_0^t \Phi^T [\dot{\tilde{\mathbf{q}}} + \Lambda \tilde{\mathbf{q}}] d\sigma + \hat{\boldsymbol{\theta}}(0) \quad (10)$$

donde la matriz de ganancias adaptables $\Gamma \in \mathbb{R}^{m \times m}$ es simétrica y definida positiva.

Las ganancias optimizadas resultantes con la GUI para la matriz K_p y K_v quedan como

$$\begin{aligned} K_p &= \text{diag}\{200, 150\} \text{ Nm/rad}, \\ K_v &= \text{diag}\{3, 3\} \text{ Nms/rad} \end{aligned}$$

y la matriz de ganancias de los parámetros dinámicos queda como $\Gamma = \text{diag}\{1.6, 0.004, 0.004\} \in \mathbb{R}^{3 \times 3}$. Se utilizó este controlador para las siguientes trayectorias deseadas

$$\begin{aligned} q_{d1}(t) &= \frac{\pi}{4} [1 - e^{-2t^3}] + \frac{\pi}{9} [1 - e^{-2t^3}] \sin(4t) \text{ rad}, \\ q_{d2}(t) &= \frac{\pi}{3} [1 - e^{-2t^3}] + \frac{\pi}{6} [1 - e^{-2t^3}] \sin(3t) \text{ rad}. \end{aligned}$$

Se utilizan condiciones iniciales $\mathbf{q}(0) = \dot{\mathbf{q}}(0) = \ddot{\mathbf{q}}(0) = [0 \ 0]^T \in \mathbb{R}^2$. Se obtuvieron resultados del desempeño de este controlador en simulación utilizando la GUI, teniendo la gráfica de los errores de posición articulares (Figura 17) y los valores de las estimaciones paramétricas (Figura 18), para un tiempo de simulación de 50 s y un paso de integración dado por 1×10^{-4} s.

III-D. Controlador con realimentación de salida adaptable

En este caso, se realizó una ley de control de voltaje \mathbf{u} proporcional a los torques $\boldsymbol{\tau}$ como $\mathbf{u}(t) = K^{-1} \boldsymbol{\tau}$, donde esta ley está dada por

$$\mathbf{u} = K_p \tanh(\sigma \tilde{\mathbf{q}}) + K_v \tanh(\tilde{\boldsymbol{\vartheta}}) + \Phi \hat{\boldsymbol{\theta}}, \quad (11)$$

donde $\tilde{\mathbf{q}}(t)$ representa el error de posiciones articulares, las matrices K_p y K_v son matrices diagonales simétricas y definidas positivas, y el vector de señales $\tilde{\boldsymbol{\vartheta}}(t)$ viene de un filtro

no lineal utilizado para estimar los errores de las velocidades articulares. Se utilizó una ley adaptable dada por

$$\hat{\boldsymbol{\theta}}(t) = -\Gamma_a \int_0^t \Phi^T [\dot{\tilde{\mathbf{q}}} - \epsilon \tanh(\sigma \tilde{\mathbf{q}})] d\tau \quad (12)$$

Con el modelo del robot manipulador y la GUI, se realizaron simulaciones que permitieron seleccionar las ganancias proporcionales y derivativas como

$$\begin{aligned} K_p &= \text{diag}\{0.4, 0.4\} \text{ Nm/rad}, \\ K_v &= \text{diag}\{7, 10\} \text{ Nms/rad}, \end{aligned}$$

y la matriz de ganancias de los parámetros adaptables como $\Gamma_a = \text{diag}\{0.1, 0.02, 0.025, 0.002, 0.03, 0.04, 0.01, 0.01\}$.

Se obtuvieron resultados en simulación y con condiciones iniciales dadas por $q_1(0) = 0$, $q_2(0) = \pi/4$, $\dot{q}_1(0) = 0$ y $\dot{q}_2(0) = 0$, y con las ganancias mencionadas anteriormente, se tiene la gráfica de los errores de posición articulares (Figura 19) y la gráfica del vector de parámetros dinámicos (Figura 20).

III-E. Controlador por redes neuronales adaptable

Este controlador por redes neuronales utiliza la siguiente ley de control

$$\boldsymbol{\tau} = \hat{W}^T \boldsymbol{\sigma}(\hat{V}^T \mathbf{x}_d) + K_p \tanh(\gamma \tilde{\mathbf{q}}) + K_d \dot{\tilde{\mathbf{q}}} + \Delta \text{sign}(\mathbf{r}), \quad (13)$$

donde K_p, K_d y Δ son matrices diagonales definidas positivas, γ es una constante escalar positiva, \hat{V} y \hat{W} son las matrices de pesos de entrada y salida estimados, respectivamente, $\mathbf{x}_d \in \mathbb{R}^{N+1}$ es el vector de señales de entrada y $\mathbf{r} = \dot{\tilde{\mathbf{q}}} + \alpha \tanh(\gamma \tilde{\mathbf{q}})$.

$$\begin{aligned} \hat{V} &= R \mathbf{x}_d \mathbf{r}^T \hat{W}^T \hat{\boldsymbol{\sigma}}, \\ \hat{W} &= F \hat{\boldsymbol{\sigma}} \mathbf{r}^T - F \hat{\boldsymbol{\sigma}} \hat{V}^T \mathbf{x}_d \mathbf{r}^T, \end{aligned}$$

donde $R \in \mathbb{R}^{[N+1] \times [N+1]}$ y $F \in \mathbb{R}^{L \times L}$ son definidas positivas, $\hat{\boldsymbol{\sigma}} = \boldsymbol{\sigma}(\hat{V}^T \mathbf{x}_d)$ y $\hat{\boldsymbol{\sigma}} = \frac{\partial \boldsymbol{\sigma}(\mathbf{x})}{\partial \mathbf{x}}$ con $\mathbf{x} = \hat{V}^T \mathbf{x}_d$.

Las ganancias resultantes para este controlador están dadas por [Falcón, 2017]

$$\begin{aligned} K_p &= \text{diag}\{40, 3\} \text{ Nm/rad}, \\ K_d &= \text{diag}\{32, 3.2\} \text{ Nms/rad}, \\ \Delta &= \text{diag}\{0.05, 0.005\}, \end{aligned}$$

y con $\alpha = 1$ y $\gamma = 5$. Además $R = 0.2 \mathbb{I}_7$ y $F = 0.5 \mathbb{I}_{20}$, donde \mathbb{I}_n es la matriz identidad de dimensiones $n \times n$. Las trayectorias a seguir en este caso fueron

$$\begin{aligned} q_{d1}(t) &= \cos(2t) - 1 \text{ rad}, \\ q_{d2}(t) &= \cos(4t) - 1 \text{ rad}. \end{aligned}$$

Se utilizaron condiciones iniciales dadas por $q_1(0) = \pi/4$ rad, $q_2(0) = -\pi/8$ rad, $\dot{q}_1(0) = \dot{q}_2(0) = 0$ rad/s. A partir de aquí se obtuvieron la gráfica de los errores de posiciones articulares (Figura 21), y las gráficas de las estimaciones neuronales de salida (Figuras 22 y 23).

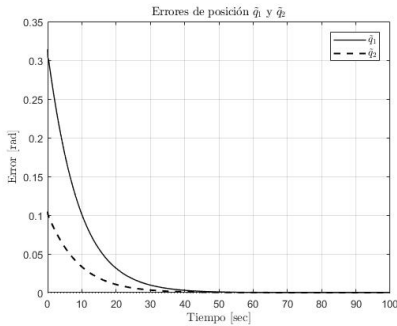


Figura 13: Errores articulares para el robot de 2 g.d.l.

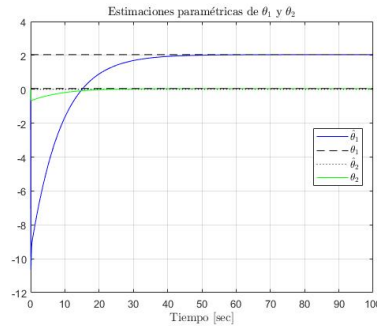


Figura 14: Estimaciones paramétricas en el controlador PD con compensación deseada de gravedad.

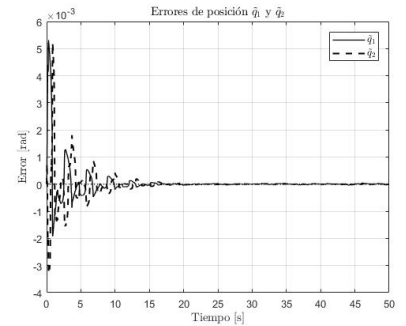


Figura 15: Errores de seguimiento de posiciones deseadas para el controlador PD+.

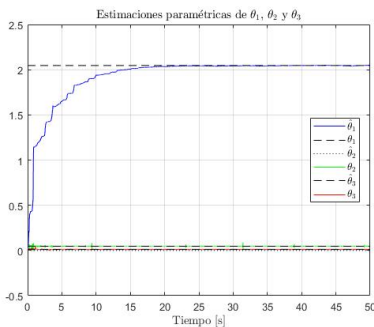


Figura 16: Estimaciones paramétricas en el controlador PD+ adaptable.

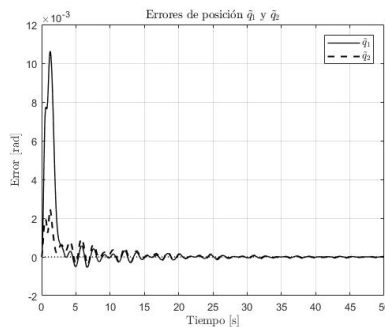


Figura 17: Errores de seguimiento de posiciones deseadas para el controlador con compensación adaptable (Slotine-Li).

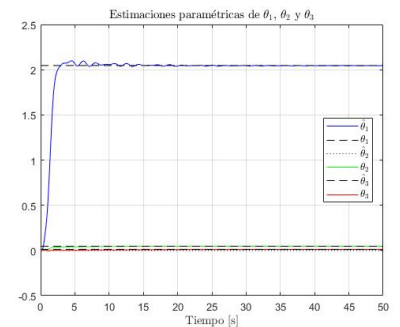


Figura 18: Estimaciones paramétricas en el controlador con compensación adaptable (Slotine-Li).

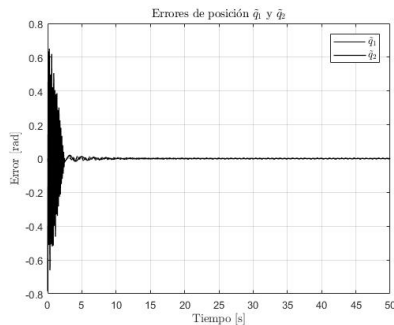


Figura 19: Errores de seguimiento de posiciones deseadas para el controlador con realimentación de salida adaptable.

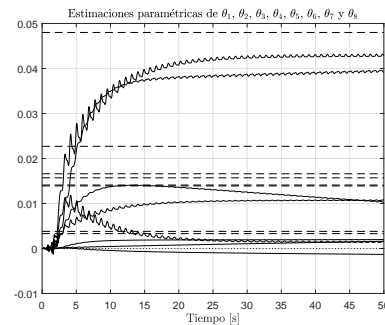


Figura 20: Estimados paramétricos con el controlador con realimentación de salida adaptable.

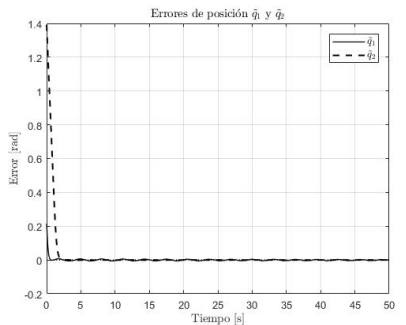


Figura 21: Errores de seguimiento de posiciones deseadas con el controlador por redes neuronales adaptable.

IV. CONCLUSIONES

En este artículo se describe el trabajo realizado para la realización de una Interfaz Gráfica de Usuario (GUI), utilizando el ambiente de programación MATLAB[®] junto con Simulink, y usando herramientas de programación orientada a objetos y

lenguaje M. Después de estudiar cada controlador, se pueden buscar las mejores ganancias para su mejor desempeño de una manera eficaz.

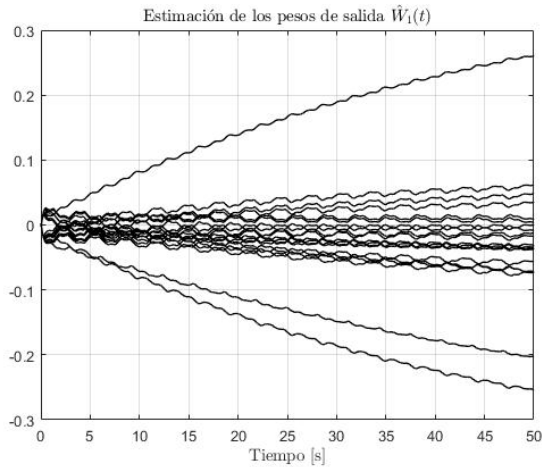


Figura 22: Evolución temporal de los pesos de salida $\widehat{W}_1(t) \in \mathbb{R}^{20 \times 1}$ para la articulación q_1 .

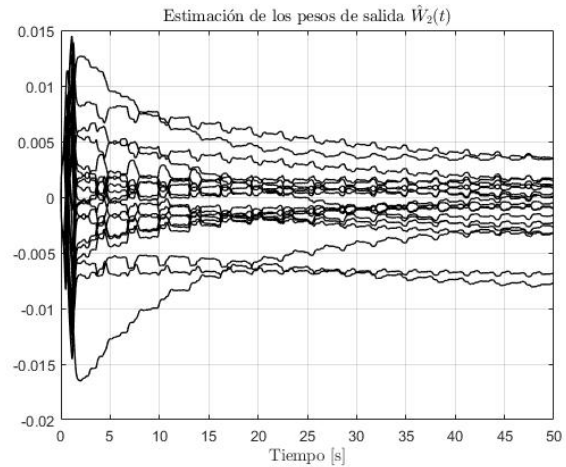


Figura 23: Evolución temporal de los pesos de salida $\widehat{W}_2(t) \in \mathbb{R}^{20 \times 1}$ para la articulación q_2

AGRADECIMIENTOS

Los autores expresan su agradecimiento al TecNM y al CONACyT 134534.

REFERENCIAS

- [Falcón, 2017] Falcón, R. (2017). Aplicación del Principio de Invariancia de LaSalle en control adaptable de sistemas robóticos manipuladores autónomos y no autónomos.
- [Falcón *et al.*, 2017] Falcón, R., Santibáñez, V., Ollervides, J., y Sandoval, J. (2017). Análisis de estabilidad y convergencia global de un esquema adaptable en regulación utilizando el principio de invariancia de LaSalle. *Congreso Internacional de Robótica y Computación CIRC 2017*, pp. 61 – 67.
- [Kelly *et al.*, 2006] Kelly, R., Santibáñez, V., y Loria, A. (2006). *Control of Robot Manipulators in Joint Space*. Advanced Textbooks in Control and Signal Processing. Springer London.
- [Lozano y Taoutaou, 2002] Lozano, R. y Taoutaou, D. (2002). *Commande adaptative et applications*. Hermes Science.
- [MATLAB®, 2015] MATLAB® (2015). *Creating Graphical User Interfaces*. MathWorks.
- [Moreno-Valenzuela *et al.*, 2010] Moreno-Valenzuela, J., Santibáñez, V., Orozco-Manríquez, E., y González-Hernández, L. (2010). Theory and experiments of global adaptive output feedback tracking control of manipulators. *IET Control Theory Applications*, 4(9):1639–1654.
- [Puga-Guzmán *et al.*, 2014] Puga-Guzmán, S., Moreno-Valenzuela, J., y Santibáñez, V. (2014). Adaptive neural network motion control of manipulators with experimental evaluations. *The Scientific World Journal*, p. 13.

Diseño de Librería en Matlab con Enfoque Didáctico para el Robot Móvil Diferencial Khepera IV.

Francesco José García Luna
Departamento de Industrial y Manufactura
Universidad Autónoma de Ciudad Juárez
Ciudad Juárez, Chihuahua
Email: francesco.garcia@uacj.mx

David Luviano Cruz
Departamento de Industrial y Manufactura
Universidad Autónoma de Ciudad Juárez
Ciudad Juárez, Chihuahua
Email: david.luviano@uacj.mx

Luis Ricardo Vidal Portilla
Departamento de Industrial y Manufactura
Universidad Autónoma de Ciudad Juárez
Ciudad Juárez, Chihuahua
Email: lvidal@uacj.mx

Resumen—En el presente artículo se muestran las especificaciones técnicas del robot Khepera en su cuarta iteración (Ver sección II). En la sección III se muestra el modelo odométrico utilizado en las funciones de control diseñadas del robot. Dichas funciones se muestran con detalle en la sección IV, en donde se especifica los argumentos de entrada de la función, así como la salida esperada. Por último en la sección V se muestran resultados experimentales de adquisición de datos de los sensores infrarrojos y ultrasónicos.

I. INTRODUCCIÓN

En el sector educativo de nivel licenciatura y superior, sobre todo en carreras de ingeniería afines a Mecatrónica es común encontrarse con distintos tipos de plataformas robóticas. Dichas plataformas son ampliamente utilizadas en prácticas de clases de nivel avanzado, en proyectos de titulación, o inclusive por los mismos docentes-investigadores para sus propios productos.

Las plataformas robóticas muy comúnmente cuentan con sus propias instrucciones para comunicación y programación, sin embargo, estas instrucciones pueden en ocasiones ser complejas o complicadas, o no estar en el lenguaje de programación deseado. Por esta razón, diversas personas, entendiendo el funcionamiento básico de redes de comunicación y la naturaleza de los sensores y actuadores involucrados, tienden a diseñar y programar sus propias subrutinas para comunicarse o controlar el robot en el lenguaje en el que más sean hábiles.

Tal es el caso de [1], quienes crearon un toolbox para Matlab llamado SpaceDyn, enfocado en soluciones numéricas para los campos de robótica e ingeniería espacial. Dicha librería contenía funciones que permiten el estudio y simulación de la cinemática y dinámica de sistemas multi-cuerpos articulados con base móvil con o sin gravedad.

Por otro lado [2], comentan que existen herramientas para el modelado y control de sistemas robóticos tipo serie y paralelos, pero no para sistemas híbridos, por lo que desarrollaron un toolbox para este tipo de sistemas. Otro ejemplo de equipos que se han enfocado en desarrollar herramientas para casos particulares de robots personalizados [3].

Se han diseñado herramientas para aplicaciones de competencia [4], y para robots industriales [5] y [6], las cuales simplificaban operaciones como configuración, control y administración de robots.

Entre otros tipos de aplicaciones, se han diseñado plataformas de experimentación completas con librerías propias para ciertos tipos de experimentos con robots móviles [7].

Existen otro tipo de librerías cuyo objetivo es disminuir el tiempo de la etapa de modelación cinemática y dinámica de los robots. Por lo que [8] diseñaron una herramienta que dados los parámetros de Denavit-Hartenberg y utilizado Euler-Lagrange son capaces de generar las ecuaciones requeridas para controlar un robot.

En particular, en Matlab, la librería más conocida para aplicaciones de robótica es la llamada Robotics Toolbox”, desarrollada por Peter Corke [9]. Esta ha ido aumentando sus funciones a través de los años.

El objetivo de este artículo es presentar una alternativa para el control y la adquisición de datos de los sensores del robot móvil diferencial Khepera IV en un lenguaje de alto nivel.

El artículo está dividido en 5 secciones. En la sección I se plantea el contexto de las librerías con enfoque didáctico para plataformas robóticas. En la sección II se muestran las especificaciones del robot utilizado, así como algunas de sus características principales. El modelo odométrico para la estimación de la pose del robot en función del desplazamiento de cada llanta se muestra en la sección III. La contribución

principal se muestra en la sección IV en donde se especifica el nombre de la función, sus argumentos de entrada, de salida y una breve descripción. Los resultados de los experimentos se muestran en la sección V y por último, las conclusiones en la sección VI.

II. ESPECIFICACIONES

Desarrollado por K-Team, el Khepera IV (mostrado en la fig. 1) ha pasado por cuatro iteraciones (descritas en el cuadro I). Siendo la cuarta iteración la versión 3.0. Este robot es del tipo móvil diferencial con restricciones de no-holonomía. Cuenta con dos ruedas actuadas y una rueda caster (no actuada).

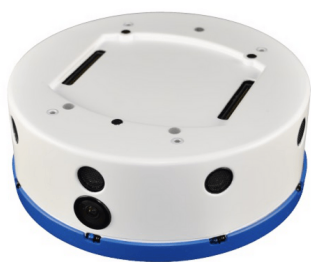


Figura 1. Khepera IV desarrollado por K-Team

Cuadro I
HISTORIAL DE REVISIONES [10]

| Revisión | Fecha | Descripción |
|----------|----------|--|
| 1.0 | 14/03/14 | Creación Inicial |
| 2.0 | 3/02/15 | Adaptado para el Gumstix Overo FireSTORM-P |
| 2.1 | 15/02/15 | Micrófono mejorado |
| 3.0 | 7/08/16 | Adaptado para el Gumstix Overo FireSTORM-Y |

Entre sus características se encuentran:

- Switch de encendido/apagado.
- LED de estatus.
- Conector MinUSB-B.
- Conector USB-A (500 mAh).
- Conector para cable de alimentación (9V, 1.5A, 0.65mm positivo al centro).
- Boton de reinicio.
- 12 sensores infrarrojos (4 por debajo y 8 en los costados).
- 5 sensores ultrasónicos.
- Cámara RGB.
- 3 LED RGB.
- Bluetooth y WiFi.
- IMU de 6 ejes (acelerómetro y giroscopio).

Se puede establecer la comunicación con el robot mediante tres diferentes protocolos, teniendo en cuenta las diferentes ventajas y desventajas del uso de cada uno:

- USB serial (115,200 Baudios, sin paridad y sin control de flujo).
- Bluetooth.
- WiFi.

II-A. Sensores

Los sensores infrarrojos colocados alrededor del robot, le permiten detectar obstáculos que se encuentren entre 2mm y 250mm, dependiendo de la calibración, las condiciones del ambiente y el color del obstáculo. También pueden ser utilizados para medir la luz ambiental. Estos sensores se encuentran con una separación de 45° entre ellos (mostrado en la fig 2).

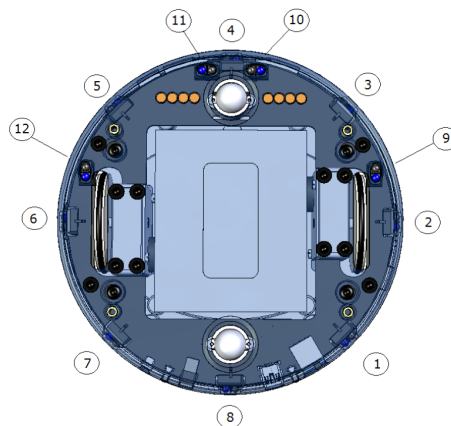


Figura 2. Localización de los sensores infrarrojos desde la vista inferior, del 1 al 8 se encuentran alrededor del robot y del 9 al 12 se encuentran por debajo.

Los sensores ultrasónicos están colocados, dos a cada lado del robot y uno al frente (mostrados en la fig. 3). Estos sensores le permiten al robot detectar obstáculos que se encuentren entre 25cm y 250cm. Al igual que los sensores infrarrojos, estos se encuentran con una separación de 45° entre ellos.

Cabe destacar que debido al principio de operación de los sensores ultrasónicos, no es posible detectar objetos a menos de 25cm. Por lo que es buena idea utilizar una técnica conocida como fusión sensorial para hacer redundante las mediciones con distintos tipos de sensores (en este caso, infrarrojos + ultrasónicos).

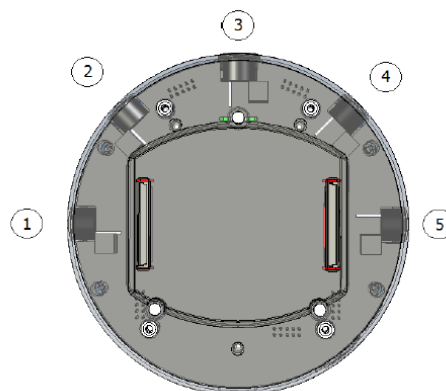


Figura 3. Localización de los sensores ultrasónicos desde la vista superior.

Cuenta una cámara RGB al frente capaz de tomar fotos o videos con un tamaño máximo de 752x480 pixeles y un procesador ARM Cortex-A8 a 800 MHz integrado capaz de procesar las imágenes. Dicha cámara tiene una distancia focal de 2.1mm. Y su campo de visión es de 131° de forma horizontal y 101° de forma vertical.

La IMU de 6 ejes es una LSM330DLC que incluye un acelerómetro y un giroscopio. Se encuentra justo sobre el eje de giro en el baricentro del robot, sobre la tarjeta electrónica.

En cuanto al acelerómetro, el eje X positivo se encuentra hacia adelante, el eje Z positivo se encuentra hacia arriba y por convención dextrogiro, el eje Y positivo se encuentra hacia la izquierda (mostrado en la fig. 4). Este regresa información de 12 bits con un rango de $\pm 2g$. Esto significa que cuando mide 1g, regresará un valor de 16,384. El acelerómetro esta configurado para actualizar los datos cada 10Hz (100ms).

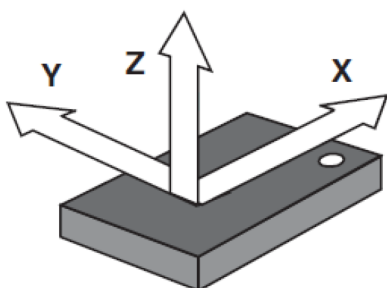


Figura 4. Localización del marco referencial del acelerómetro

Por otro lado, el giroscopio esta configurado a $\pm 2000dps$ con una frecuencia de 95Hz. Se utiliza la ec. 1 para convertir la información cruda en grados/segundo. El eje X gira en el plano YZ, el eje Y gira en el plano XZ, y por último, el eje Z gira en el plano XY (mostrado en la fig. 5).

$$\frac{deg}{sec} = data_{raw} * 0,66 \quad (1)$$

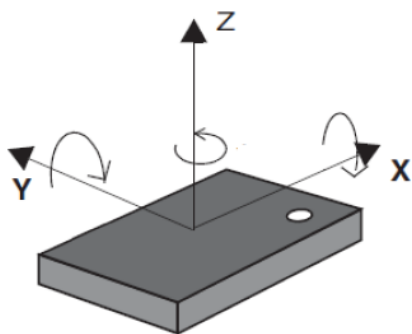


Figura 5. Localización de los ejes de giro detectables desde la vista superior

Además, cada rueda de 42mm de diámetro cuenta con encoder de 19,456 pulsos por vuelta. Y tiene una circunferencia de 131.94mm, lo que significa que genera 1 pulso cada 0.006782mm (6.7818 μ m).

III. MODELO ODOMÉTRICO DE ROBOTS MÓVILES DIFERENCIALES

En los robots diferenciales es normal utilizar odometría para el cálculo del posicionamiento, siempre considerando que no existe derrape. En este tipo de robots, el movimiento se consigue con dos ruedas actuadas y una o mas ruedas de apoyo (mostrado en la fig. 6).

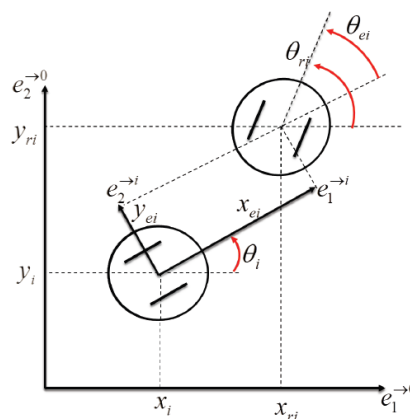


Figura 6. Esquema del modelo de un robot diferencial

El movimiento se genera considerando combinando la velocidad y dirección de cada una de las ruedas (mostrado en la fig. 7). Si se desea ir hacia adelante (considerando que los motores no están posicionados de forma encontrada) ambos motores deben de ir en la misma dirección; si se desea girar hacia algún lado, ambas llantas deben de tener diferente velocidad.

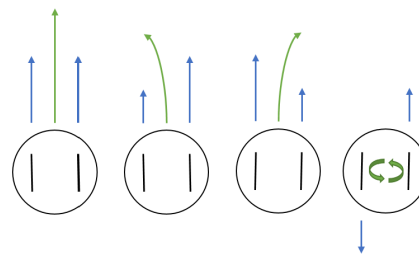


Figura 7. Movimientos posibles de un robot diferencial en función de la dirección y velocidad de cada rueda.

El desplazamiento del robot según [11] esta dado por la ec. 2:

$$D = \frac{D_1 + D_2}{2} \quad (2)$$

Donde D es el desplazamiento del centroide del robot, D_1 y D_2 es el desplazamiento individual de cada llanta.

Mientras que la orientación del robot esta descrita por la ecuación 3:

$$\theta = \frac{D_1 - D_2}{d} \quad (3)$$

Donde d es la distancia entre llantas.

Se puede determinar el desplazamiento lineal de la llanta si se considera el radio efectivo y el encoder (descrito en la ec. 4).

$$D_i = \frac{2\pi N_i}{C_i} R_{ei} \quad (4)$$

Donde:

- R_{ei} : radio efectivo de la llanta i (considerando compliancia).
- R_{ni} : radio nominal de la llanta i .
- N_i : Cantidad de pasos recorridos del encoder.
- C_i : Cantidad de pasos totales del encoder por revolución de la llanta.
- $R_{ei} = R_{ni}$ en el caso particular del Khepera IV.

Mientras que para la dirección la ec. 3 se puede expresar en función del encoder como:

$$\theta = \frac{2\pi R_e}{Cd} (N_1 - N_2) \quad (5)$$

Por último, utilizando la fig. 6 y las ecuaciones 4 y 5 se puede estimar la posición del robot como:

$$\begin{aligned} x &= D \cos(\theta) \\ y &= D \sin(\theta) \end{aligned} \quad (6)$$

IV. CONTRIBUCIÓN PRINCIPAL

Considerando [10], los sensores del robot móvil diferencial y la naturaleza de los datos, se programó una serie de funciones en Matlab para controlar y obtener datos del robot. El toolbox se presenta como una alternativa para controlar al robot en modo esclavo.

Para lograr esto, primeramente hay que configurar el robot para que acepte comandos por puerto serial. Se recomienda utilizar un sistema operativo basado en UNIX para facilitar la conexión por bluetooth-serial. En el caso particular se utilizó el paquete Minicom para establecer la conexión bluetooth-serial con el robot.

El protocolo de comunicación del robot funciona del modo comando-respuesta. Todos los comandos son enviados en código ASCII. En caso de que se requiera hacer la conexión en otro sistema operativo, se tiene que cambiar la dirección serial a la correspondiente.

El toolbox actualmente cuenta en su versión 1.0 con 17 funciones:

1. `data = khepera_ALValues(s)`:
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Argumento(s) de salida:
 - arreglo de tamaño 12x1 (data)
 - Notas
 - Entrega el valor de la incidencia de rayos infrarrojos
 - 0 → 1024 (IR saturado - IR carente)
2. `s = khepera_connect`:
 - Argumento(s) de salida:
 - objeto serial (s)
 - Notas
 - Conecta el robot en modo esclavo en la dirección serial `/dev/rfcomm0` a 115,200 Baudios
3. `khepera_disconnect(s)`:
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Notas
 - Detiene las ruedas del robot y se desconecta del puerto serial
4. `data = khepera_encodersValues(s)`:
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Argumento(s) de salida:
 - arreglo de tamaño 2x1 (data)
 - Notas
 - posición absoluta del encoder de cada llanta
5. `data = khepera_getBatteryStatus(s)`:
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Argumento(s) de salida:
 - celda con 7 parámetros de solo lectura (data)
 - Notas
 - Contiene información sobre el voltaje, corriente, corriente promedio, capacidad absoluta disponible (mAh), capacidad absoluta relativa (%), Temperatura y estatus del conector AC (conectado o no conectado)
6. `khepera_getFirmware(s)`:
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Notas
 - Da información en la ventana de comandos sobre la versión actual del firmware del robot
7. `data = khepera_IRValues(s)`:
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Argumento(s) de salida:
 - arreglo de tamaño 12x1 (data)
 - Notas
 - 0 → 1024 (Obstáculo alejado - Obstáculo cerca), lectura comenzando desde el sensor ubicado atrás a la izquierda y ordenado en sentido horario. Los últimos 4 valores corresponden a los sensores ubicados en la parte inferior de izquierda a derecha.
8. `[raw_data, mean_data] = khepera_readAccelalues(s)`:
 - Argumento(s) de entrada:
 - objeto serial (s)

- Argumento(s) de salida:
 - arreglo de 10x3 (raw_data)
 - arreglo de 1x3 (mean_data)
 - Notas
 - raw_data contiene 10 datos del historial de aceleración inercial sin procesar de los tres ejes cartesianos $[x, y, z]$
 - mean_data contiene el promedio de 10 mediciones de aceleración inercial de cada eje.
- 9. [raw_data, mean_data] = khepera_readGyroValues(s):
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Argumento(s) de salida:
 - arreglo de 10x3 (raw_data)
 - arreglo de 1x3 (mean_data)
 - Notas
 - raw_data contiene 10 datos del historial del giroscopio sin procesar sobre los tres ejes cartesianos $[x, y, z]$
 - mean_data contiene el promedio de 10 mediciones del giroscopio sobre cada eje.
- 10. data = khepera_readSpeed(s):
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Argumento(s) de salida:
 - arreglo de 2x1
 - Notas
 - Contiene información sobre la velocidad lineal (m/s) de cada rueda
- 11. khepera_resetEncoders(s):
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Notas
 - Contiene información sobre la velocidad lineal (m/s) de cada rueda
- 12. khepera_RGB(s, led₁, led₂, led₃):
 - Argumento(s) de entrada:
 - objeto serial (s)
 - arreglo de 1x3 (led_i)
 - Notas
 - Los valores de saturación de cada canal están en el rango de 0-255
- 13. khepera_setGoal(s, motorTravel):
 - Argumento(s) de entrada:
 - objeto serial (s)
 - arreglo de 1x2 (motorTravel)
 - Notas
 - Ocasiona que el robot desplace la rueda(i) motorTrave(i) mm lineales
- 14. khepera_setSpeed(s, lspeed, rspeed):
 - Argumento(s) de entrada:
 - objeto serial (s)
 - escalar (lspeed y rspeed)
 - Notas
 - Ocasiona que la rueda(i) alcance la velocidad indicada a lazo cerrado utilizando el esquema de control PID interno
- 15. khepera_setSpeedOpens, ldir, lspeed, rdir, rspeed):
 - Argumento(s) de entrada:
 - objeto serial (s)
 - escalar (lspeed y rspeed)
 - booleano (ldir y rdir)
 - Notas
 - Manda una señal PWM (0-255) y dirección a cada rueda a lazo abierto
 - dir = verdadero: hacia adelante
 - dir = falso: hacia atrás
- 16. serial_out = khepera_talk(s, data):
 - Argumento(s) de entrada:
 - objeto serial (s)
 - cadena de caracteres (data)
 - Argumento(s) de salida:
 - cadena de caracteres (serial_out)
 - Notas
 - Es la base de todas las funciones, se envía en código ASCII el comando y se registra la respuesta
- 17. data = khepera_USValues(s):
 - Argumento(s) de entrada:
 - objeto serial (s)
 - Argumento(s) de salida:
 - arreglo de 5x1 (data)
 - Notas
 - Información acerca de los valores de los sensores ultrasónicos (en cm) de izquierda a derecha
 - un valor de 1000 significa que no se registra un obstáculo
 - un valor de 0 significa que existe un obstáculo a menos de 25cm
 - un valor entre 25 y 250 significa la distancia entre el sensor y el obstáculo en cm

V. RESULTADOS

La librería se utiliza actualmente en distintas clases de la carrera de Ingeniería Mecatrónica en la Universidad Autónoma de Ciudad Juárez, por ejemplo:

- Diseño Mecatrónico.
- Robótica.
- Control I y II.
- Seminario de Tesis y Trabajo de Tesis.

Se diseñaron dos experimentos:

1. Estimación de la distancia de obstáculos mediante la adquisición de datos ultrasónicos.

2. Adquisición de datos de los sensores infrarrojos y ultrasónicos para realizar fusión sensorial.

Los resultados del experimento 1 mostrados en la fig. 8, en donde se puede observar la distancia a los obstáculos en centímetros desde el robot cada 0.001 segundos (mostrados en la ec. 7).

Se utilizó la función $data = khepera_USValues(s)$, en donde:

- $data$: arreglo de tamaño 5x1 con información en cm de los sensores ultrasónicos.
- s : objeto serial que apunta a la dirección en la que está conectado el Khepera.

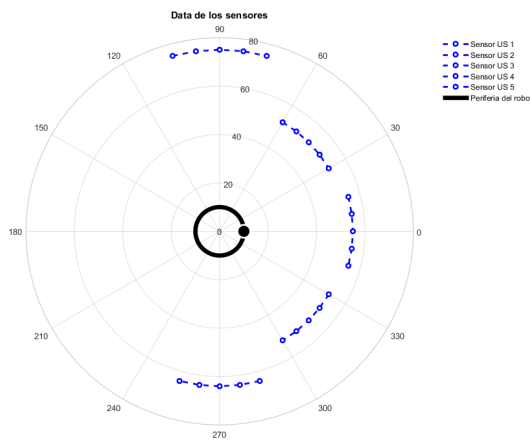


Figura 8. Experimento de adquisición de datos de los sensores ultrasónicos en cm.

$$\begin{bmatrix} \theta_1 & d_{u1} \\ \theta_2 & d_{u2} \\ \theta_3 & d_{u3} \\ \theta_4 & d_{u4} \\ \theta_5 & d_{u5} \end{bmatrix} = \begin{bmatrix} \pi/2 & 75 \\ \pi/4 & 52 \\ 0 & 55 \\ -\pi/4 & 52 \\ -\pi/2 & 64 \end{bmatrix} \quad (7)$$

Donde d_{ui} es la distancia en centímetros hacia un obstáculo desde el robot con un ángulo θ_i radianes en el plano XY.

En la fig. 9 se puede observar el resultado del experimento 2 de fusión sensorial en donde se obtiene en tiempo real información de los sensores infrarrojos al mismo tiempo que información de los sensores ultrasónicos.

Se utilizó la función $data = khepera_IRValues(s)$, en donde:

- $data$: arreglo de tamaño 8x1 con información en cm de los sensores infrarrojos.
- s : objeto serial que apunta a la dirección en la que está conectado el Khepera.

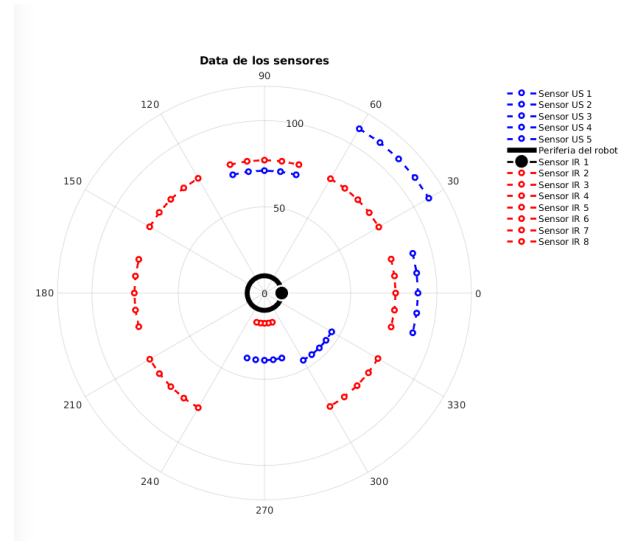


Figura 9. Experimento de fusión sensorial de datos de los sensores ultrasónicos e infrarrojos en cm.

Los dos experimentos fueron realizados en distintas situaciones, por lo que es de esperarse que ambas gráficas tengan información diferente.

VI. CONCLUSIÓN

Debido a la naturaleza de los sensores infrarrojos, es necesario que los experimentos para la detección de obstáculos se realice bajo condiciones de luz artificial únicamente.

Sin embargo, la utilización de este toolbox permite al usuario enfocarse en los resultados de experimentación y no en la parte de programación del robot. Ya que esta última suele ser la de mayor duración.

En la versión actual del toolbox, puede ser utilizado para aplicaciones de SLAM, regulación punto a punto, seguimiento de trayectorias, detección de inclinación en el plano de navegación, detección de incidencia infrarroja, entre otras.

En siguientes versiones del toolbox, se planea incluir funciones para graficar los datos de los sensores y algoritmos de control y seguimiento de trayectorias a lazo cerrado de uno o múltiples agentes.

RECONOCIMIENTO

La librería puede adquirirse enviando un correo a la dirección de correo electrónico francesco.garcia@uacj.mx o david.luviano@uacj.mx.

Los autores extienden su agradecimiento a la Secretaría de Educación Pública de México por financiar este trabajo a través del Acuerdo de Investigación 511-6 / 17-7605.

REFERENCIAS

- [1] K. Yoshida, "The spacedyn: A matlab toolbox for space and mobile robots," 1999.
- [2] H. Ozakyol, C. Karaman, and A. Bingül, "Robotics toolbox for kinematic analysis and design of hybrid multibody systems," 2017.
- [3] J. Nicho and B. Nowak, "A versatile toolbox for studying anthropomorphic robot hands," 2009.

- [4] J. Jang, C. K. Ahn, S. Han, and W. H. Kwon, "Rapid control prototyping for robot soccer system using simtool," 2006.
- [5] F. Chinello, S. Scheggi, F. Mordibi, and D. Prattichizzo, "Kct: A matlab toolbox for motion control of kuka robot manipulators," 2010.
- [6] E. Engels, "A versatile matlab toolbox for rapid-robot-prototyping of custom made industrial robots," 2016.
- [7] M. Kloetzer, S. Magdici, and A. Burlacu, "Experimental platform and matlab toolbox for planning mobile robots," 2012.
- [8] E. Dean-Leon, S. Nair, and A. Knoll, "User friendly matlab-toolbox for symbolic robot dynamic modeling used for control design," 2012.
- [9] P. Corke, *Robot Manipulator Capability in Matlab*. Mathworks, 2017.
- [10] K-Team, *Khepera IV User Manual*. K-Team, 2016.
- [11] M. A. G. Rodriguez. (2011, jan) Modelo odométrico diferencial de robots móviles. [Online]. Available: <http://www.tamps.cinvestav.mx/~mgomez/>

Diseño y construcción de un guante portable para medición de flexión-extensión de falanges

Ismael Hernández-Capuchin*[§], Cristina Ramírez-Fernández*[§], Edgar Pérez*[†], Erick Rodríguez*[†], Iván Rubio*[†]

* Tecnológico Nacional de México \ Instituto Tecnológico de Ensenada

Blvd. Tecnológico 150, Ex-ejido Chapultepec,

Ensenada, B.C., México 22780

Email: {ihernandez, cramirez, al14760571, al14760573, al14760574}@ite.edu.mx

[§]División de Estudios de Posgrado e Investigación

[†]Ingeniería Electrónica

Resumen—En este trabajo se presenta el diseño y construcción de un guante portable, basado en un sistema embebido de bajo consumo con sensores de flexión, para la medición de posiciones angulares de falanges. Esta clase de dispositivos son ampliamente utilizados en sistemas de terapia de rehabilitación para medir y dar seguimiento a personas con limitaciones en sus movimientos. La funcionalidad y precisión del sistema propuesto es ilustrado a través de las curvas de calibración de los sensores de flexión y resultados experimentales de las posiciones angulares obtenidas para realizar dos movimientos de cuatro actividades de la vida cotidiana.

I. INTRODUCCIÓN

De acuerdo a la Organización Mundial de la Salud (OMS), alrededor del mundo más de un billón de personas viven con alguna limitación funcional. En México los estudios demuestran que más del 17.7% de las personas sufren de una lesión física en sus manos o dedos causada de manera accidental [1]. Para disminuir las afectaciones en sus actividades de la vida diaria, los especialistas se apoyan en diversos dispositivos de asistencia que permiten medir y dar seguimiento a la rehabilitación [2].

En el tratamiento para la rehabilitación de mano, se han desarrollado guantes basados en diversas tecnologías; las más comunes son aquellas basadas en el uso de sensores de flexión, acelerómetros, sistemas asistidos por visión, entre otros [3], [4], [5]. Desde el punto de vista de instrumentación [6], los sistemas basados en visión requieren del uso de marcadores y su captación por parte de las cámaras; por lo tanto, una reconstrucción tridimensional del movimiento requiere la utilización de más de una cámara. En el caso de sistemas basados en acelerómetros, se requiere el diseño de filtros para rechazar las mediciones de ruido y vibraciones que no corresponden a los movimientos; además, es necesario satisfacer diversas condiciones de implementación [7]. Algunos de los guantes comerciales como Cyberglove III [8], 5DT Data glove [9], Power Glove, X-IST DataGlove, P5 Glove [10], han sido probados clínicamente con resultados exitosos. Sin embargo, su elevado costo y difícil acceso limita su uso por parte de los especialistas [11].

En este trabajo se propone un guante portable de bajo costo, diseñado y construido para asistir la rehabilitación ocupacional, y medir el progreso en la flexión y extensión

de los dedos de la mano. Mientras que la medición precisa de los movimientos fraccionados de los dedos se puede realizar de manera controlada en un laboratorio, aún no está muy claro como estas mediciones corresponden de manera útil en las actividades de la vida diaria. La capacidad de monitorear adecuadamente las funciones de la mano, el rango de movimiento (ROM, por sus siglas en inglés), así como también la fuerza y la habilidad para realizar las actividades de la vida diaria, son de suma importancia [12]. Tradicionalmente los especialistas realizan las mediciones de ROM usando goniómetros con resultados exitosos cuando el movimiento es pasivo. Sin embargo, estas mediciones de ROM son imposibles de evaluar en las tareas funcionales sin el uso de dispositivos de asistencia [1]. Por tanto, el desarrollo de dispositivos de asistencia efectivos y seguros para la rehabilitación de mano sigue siendo un problema abierto.

I-A. Contribuciones

La aportación de este trabajo corresponde al diseño, construcción y evaluación experimental de un prototipo de guante portable, para medición de posiciones angulares de las falanges, orientado al seguimiento en terapias de rehabilitación [13]. Las características del sistema propuesto se enuncian a continuación.

- Los movimientos analizados por el sistema son la extensión y la flexión de cada uno de los dedos de la mano derecha.
- Validación experimental de repetitividad y uniformidad de mediciones mediante la comparación de dos tipos de sensores de flexión.
- Implementación de dos sensores de flexión por dedo, que permiten una medición angular con mayor precisión.
- La portabilidad del sistema propuesto se basa en la inclusión de un mecanismo de almacenamiento de datos en una memoria microSD y una batería alcalina.
- Los datos son almacenados para su procesamiento fuera de línea, con el objetivo de ser incorporados en alguna evaluación posterior por parte de un especialista en rehabilitación.
- Se evalúan dos movimientos (mano con dedos extendidos y mano con dedos flexionados).

El presente documento está organizado como se describe a continuación. En la Sección II se presenta el diseño del sistema embebido y los aspectos funcionales del prototipo construido. Resultados experimentales de mediciones realizadas por el prototipo de guante, para realizar movimientos de flexión y prensión se describen en la Sección III. Finalmente, las conclusiones y el trabajo futuro se analizan en la Sección IV.

II. DISEÑO DEL GUANTE PORTABLE

El sistema embebido, que permite la adquisición y almacenamiento de los datos generados por el guante portable, se encuentra basado en la tarjeta Arduino Mega2560 (AVR ATmega2560). Este sistema utiliza sensores de flexión para medición de extensión y flexión de los dedos de la mano. El sensor de flexión que se utilizó es el *Bend* fabricado por la compañía Flexpoint; este elemento es una pieza flexible de plástico con un grosor típico de 0.012cm, recubierta por una tinta con una mezcla de carbón y polímero, la cual es un elemento de tipo resistivo. Durante el proceso de fabricación, cuando la tinta es colocada sobre la pieza plástica, se forman micro grietas en el material de recubrimiento lo que permite que la pieza sea flexible, lo suficientemente suave como para doblarla pero lo suficientemente dura como para que pueda regresar a su posición original dándole las propiedades de flexionarse sin perder su forma original. Además, esta propiedad le permite regresar a su valor original, con la misma pendiente en la curva de comportamiento del sensor, sin importar que sea doblado repetidas veces sobre el mismo radio de curvatura. Estas características, junto con la confiabilidad y repetitividad del sensor *Bend* lo hacen ideal para ser utilizado como sensor de flexión en el guante.

También se propone la utilización de una batería alcalina, y la implementación de un mecanismo para el almacenamiento de datos; lo que permite incorporar portabilidad en el sistema propuesto. Así, el procesamiento y análisis de los datos obtenidos por medio de los sensores de flexión, se lleva a cabo fuera de línea. Como resultado, se obtienen las curvas representativas de movimientos comunmente realizados de la vida diaria, calibrados con el guante desarrollado. Asimismo, se ha desarrollado un programa en Matlab que permite caracterizar las mediciones de los sensores de flexión, en ángulos articulares, mediante la interpolación aplicando un polinomio de ajuste de orden 5. La Figura 1 muestra una foto del prototipo construido. Para evitar la obstrucción de movimientos de los dedos, se ha colocado el sistema embebido, la batería, y la unidad de almacenamiento en el brazo.

El registro de las mediciones incluye una estampa de tiempo obtenida a través de una tarjeta de reloj basada en el integrado de bajo costo Maxim Integrated DS1307, el cual utiliza un oscilador de cristal externo a una frecuencia de 32.768kHz [14]. Este integrado también funciona como calendario y tiene la capacidad de generar estampas de tiempo con la precisión de segundos, minutos, horas, día, fecha, mes y año con ajustes en los meses, y año bisiesto. De esta manera, la evaluación y seguimiento de los movimientos, con respecto a tiempos y fecha, puede ser guardado en forma de bitácora.

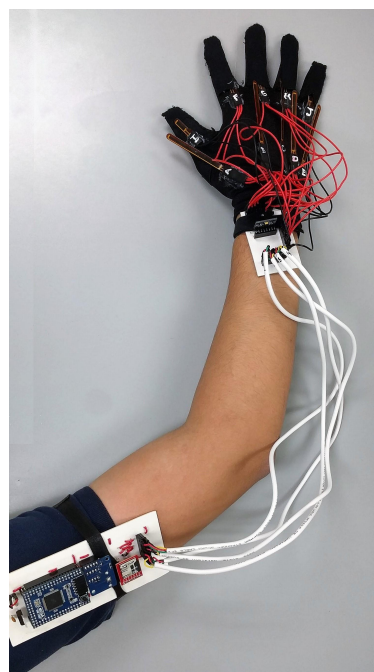


Figura 1: Vista superior del prototipo del guante portable.

También posee un circuito de sensado para detectar fallas en su alimentación y conmutar automáticamente a una alimentación de respaldo. La fuente de alimentación de respaldo es una batería de litio tipo botón de 3 volts, la cual le permite seguir funcionando y manteniendo la hora aunque el prototipo esté apagado. La adquisición de la fecha y hora se puede generar, para cada medición realizada con el guante, a través de una comunicación serial I2C.

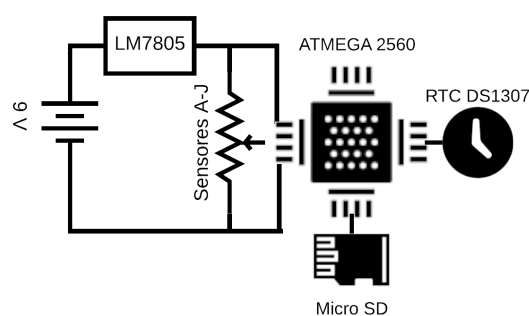


Figura 2: Esquema de conexiones del guante.

La creación de archivos de bitácora de las mediciones realizadas por el sistema, es implementada utilizando un conector para tarjeta microSD con un divisor de voltaje a 3.3 volts. La interfaz de comunicación utilizada es el protocolo SPI (Serial Peripheral Interface) para lectura/escritura de datos en una tarjeta de memoria microSDHC SanDisk C10 de 4 Gb.

Considerando la portabilidad del sistema, se incorpora una

batería alcalina de 9 volts y un regulador de voltaje LM7805 a 5 volts para alimentar la tarjeta del microcontrolador AT-Mega2560. El consumo aproximado del prototipo en funcionamiento es de 52 mA, por lo que es posible un uso continuo de operación de entre 7 y 9 horas. El esquema del diseño electrónico del sistema de adquisición de datos del guante se muestra en la Figura 2.

En este diseño, se ha considerado colocar dos sensores de flexión a lo largo de cada dedo. Esto permite una mejor descripción y sensibilidad en las mediciones de los movimientos realizados por el usuario. La Figura 3 describe la configuración de distribución de cada sensor sobre el dorso de la mano. En este caso, sólo se considera la mano derecha.

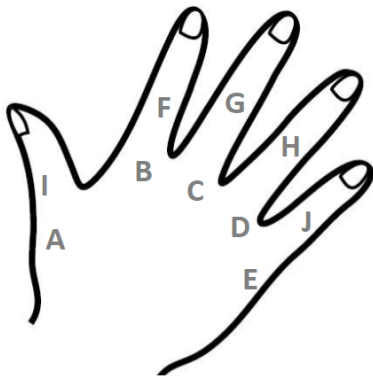


Figura 3: Distribución de los sensores de flexión con respecto a los dedos de la mano.

Para el diseño de este prototipo se analizaron dos opciones de sensores de flexión, los sensores resistivos patentados por *Spectra Symbol* (*Flex* [15]), y los sensores bidireccionales fabricados por la compañía *Flexpoint* (*Bend* [16]), ver Tabla I. Para realizar una discriminación entre ambos dispositivos, se propuso medir la respuesta en voltaje al flexionar un sensor cada dos grados. Se promediaron tres experimentos en tiempos distintos, cada uno con cinco mediciones. El promedio y la desviación estándar de estos tres experimentos, permitió estudiar el comportamiento de los sensores de tipo *Flex* y de tipo *Bend*, ver Figura 4. Nótese que la repetibilidad del sensor *Flex* es baja, dado que la medición promedio en voltaje a 0 grados se encuentra en el rango de la desviación estándar cuando el sensor se encuentra flexionado a 90 grados. Considerando los resultados obtenidos, se propone el uso de sensores bidireccionales tipo *Bend* para el diseño del guante.

De acuerdo a los tipos de movimientos y la proporción en longitud de las diferentes falanges de la mano, se han seleccionado tres dimensiones en los sensores con la siguiente distribución: sensores de 3" para medir el ángulo de flexión de las falanges proximales con respecto a su correspondiente metacarpo (sensores A, B, C, D, E); sensores de 2" para obtener el ángulo de flexión entre las falanges proximales y las falanges mediales, de los dedos índice, medio y anular (sensores F, G, H); y sensores de 1" para obtener la lectura de flexión de la falange distal con respecto a la falange proximal del dedo pulgar, y la falange proximal con respecto a la falange

Tabla I: Comparación entre los sensores de flexión tipo *Bend* y tipo *Flex* [6].

| Características | <i>Bend</i> | <i>Flex</i> |
|---------------------------------|----------------|-----------------|
| Ciclos de trabajo | > 1 millón | > 1 millón |
| Rango de temperatura | -35°C a 80°C | -35°C a 80°C |
| Resistencia de referencia | 100 a 500 KΩ | 25 KΩ |
| Tolerancia | NA | 30 % |
| Rango de resistencia en flexión | 1.5 KΩ - 40 KΩ | 45 KΩ - 125 KΩ |
| Consumo energético | NA | 0.5 W continuos |
| Histéresis | 7 % | NA |
| Resolución | < 1° | < 1° |
| Voltaje de operación | 5 V - 12 V | NA |

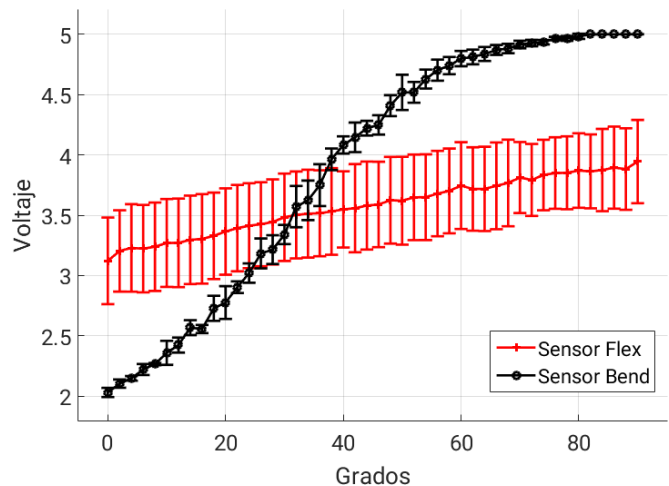


Figura 4: Comparación entre las curvas de calibración obtenidas para los sensores *Flex* de 3.75", y los sensores *Bend* de 3".

medial del dedo meñique (sensores I y J respectivamente) (ver Figura 3).

Cada sensor es resistivo y bidireccional, y puede modelarse como un divisor de voltaje. Así, la flexión en el sensor produce un cambio en su resistencia, y por ende una caída de voltaje a la salida de su configuración. Se realizaron 5 mediciones de la caída del voltaje con una resolución de dos grados de flexión angular. Las curvas características promedio de cada sensor, obtenidas en las mediciones, se presentan en la Figura 5. La Figura 5(a) corresponde a los sensores A, B, C, D, y E; la Figura 5(b) describe la curva de los sensores F, G, y H; y la Figura 5(c) muestra la curva del sensor I y J. Las mediciones obtenidas están comprendidas entre 1 y 5 Volts, correspondiente a un ángulo de flexión de cada sensor entre 0 y 90 grados, respectivamente. Nótese que los sensores de 2" presentan una menor dispersión entre sí.

Una vez obtenidas las mediciones promedio de cada sensor, se procedió a un ajuste de la curva tomando en cuenta un polinomio de grado 5 de la forma $\theta(V) = a_5V^5 + a_4V^4 + a_3V^3 + a_2V^2 + a_1V + a_0$. La Tabla II muestra el modelo característico de cada uno de los sensores.

Una vez terminado el proceso de caracterización cada sensor, se procede a realizar una validación experimental, a través mediciones y el registro de las posiciones articulares,

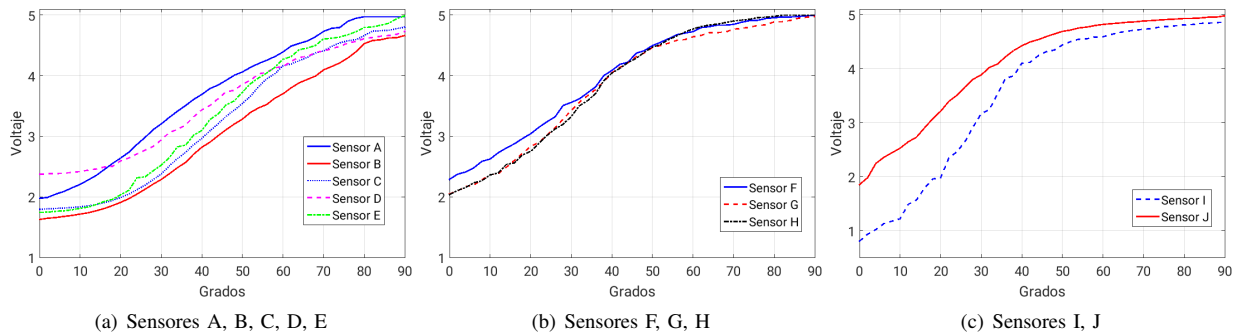


Figura 5: Curvas características obtenidas a partir de las mediciones de los sensores de flexión *Bend* utilizados en el guante. (a) sensores de 3", (b) sensores de 2", y (c) sensores de 1".

Tabla II: Modelo característico de cada sensor de acuerdo a las mediciones obtenidas.

| Sensor | a_5 | a_4 | a_3 | a_2 | a_1 | a_0 |
|--------|-------|---------|---------|----------|---------|----------|
| A | 2.14 | -37.85 | 265.12 | -914.98 | 1570.48 | -1063.16 |
| B | 3.02 | -50.28 | 328.22 | -1049.20 | 1659.14 | -1016.09 |
| C | 4.17 | -70.16 | 466.69 | -1530.85 | 2487.80 | -1577.17 |
| D | 10.47 | -190.73 | 1382.73 | -4981.17 | 8925.02 | -6343.00 |
| E | 2.01 | -34.49 | 234.08 | -785.71 | 1316.24 | -857.00 |
| F | 6.88 | -118.47 | 806.14 | -2711.83 | 4533.57 | -3018.20 |
| G | 1.81 | -28.35 | 177.93 | -565.47 | 931.52 | -630.97 |
| H | 5.08 | -85.22 | 563.72 | -1839.11 | 2979.75 | -1915.84 |
| I | 1.41 | -18.11 | 88.45 | -205.24 | 237.56 | -99.25 |
| J | 6.23 | -99.86 | 625.55 | -1912.54 | 2865.03 | -1682.48 |

para actividades comunes en la vida cotidiana.

III. RESULTADOS EXPERIMENTALES

Se realizaron experimentos para el prototipo de guante portable diseñado con la finalidad de validar su correcto funcionamiento. De acuerdo a [17], la prensión se refiere a *cada postura estática de la mano con la que un objeto se puede sostener de forma segura con una mano*. La prensión es parte fundamental de manipulación de los objetos porque requiere de una selección de comandos motrices que permitan realizar este movimiento. Por ejemplo, el movimiento necesario para prevenir la caída de los objetos implica la estabilización de la fuerza ejercida en las superficies al prensar (fuerzas de prensión) en relación de la desestabilización de las fuerzas tangenciales de las superficies (fuerzas de carga) [18].

Se realizaron mediciones de movimientos de prensión y extensión de falanges mediante el promediado de tres lecturas para cada experimento. Se han seleccionado tres actividades con movimiento de prensión, y un movimiento de extensión de todos los dedos de la mano para la calibrar el sistema, y generar una posición de referencia.

La Figura 6 muestra las mediciones de los sensores de flexión en forma de voltaje y su conversión a posiciones angulares para las 4 actividades realizadas. Esto es, mano con dedos extendidos, mano con dedos flexionados (prensión de botella, prensión de plumón y prensión de puño). La selección de estos movimientos coincide con lo reportado en otras investigaciones, véase [19], [20].

Observe que en la medición de calibración, en donde se encuentra la mano extendida, se obtienen dos mediciones de posición angular negativas. Esto ocurre cuando la extensión de las falanges no es completamente horizontal, lo cual puede ocurrir por la fisonomía de la mano de cada persona. Dado que los sensores de flexión son bidireccionales, es posible obtener estas mediciones y determinar cuando la mano extendida no se encuentra completamente alineada. Por otro lado, ofrece la posibilidad de generar una medición de ajuste para compensar esta desviación y considerarla de acuerdo a cada persona, como su plano de referencia relativo.

Nótese que en las mediciones de las actividades del movimiento de prensión, las posiciones angulares para el puño cerrado son mayores que en las actividades en la prensión de botella y del plumón. Lo anterior corresponde a lo esperado, considerando que este será el máximo valor de flexión que puede generar el sujeto de prueba en este caso. Para esta actividad, la menor posición angular ocurre en el sensor A (correspondiente al del dedo pulgar) dado que se coloca sobre las otras falanges y en consecuencia, tiene menor flexión.

Los resultados obtenidos para las actividades de prensión de una botella y de un plumón también son consistentes, dado que se observan posiciones angulares de menor flexión para el caso de la botella. Las mediciones de flexión para las actividades del puño y el plumón son uniformes; sin embargo, para el caso de la botella, se observan posiciones angulares negativas en el dedo índice (sensor B) y el meñique (sensor E). Esto ocurre por la manera en que el sujeto de prueba sostiene el objeto. En este caso, se detectó una tendencia a extender estos dos dedos al sujetar objetos de mayor dimensión. Se observa consistencia entre esta actividad y las mediciones de mano con dedos extendidos utilizada para calibración, ya que son los mismos sensores que detectan una posición angular negativa.

IV. CONCLUSIÓN

En este trabajo se presenta el diseño y construcción de un guante portable de bajo costo para la medición de flexión y extensión de los dedos de la mano. Para este diseño, se ha caracterizado y seleccionado el uso de sensores de flexión tipo *Bend* considerando una alta repetibilidad, de acuerdo a los resultados experimentales obtenidos. El diseño se encuentra

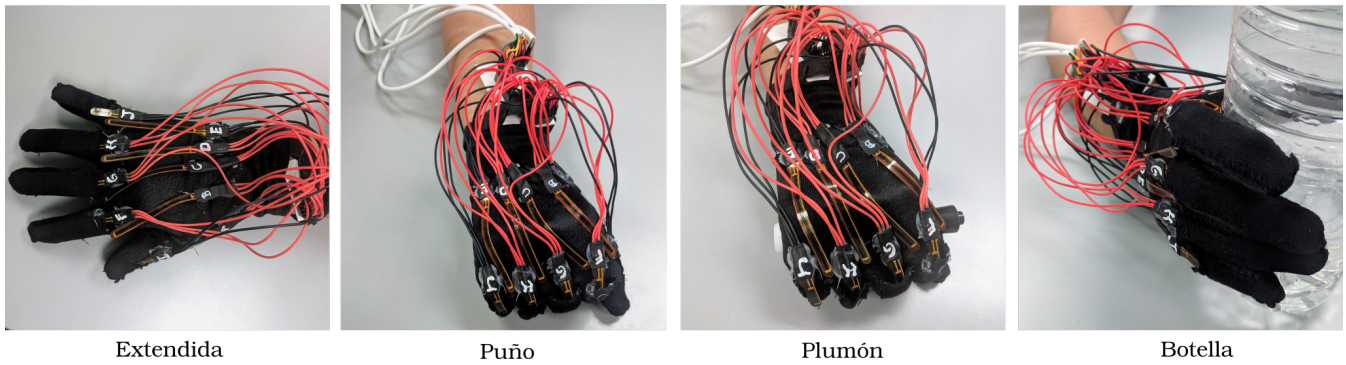


Figura 6: Actividades de la vida cotidiana propuestas para representar los movimientos de extensión y flexión.

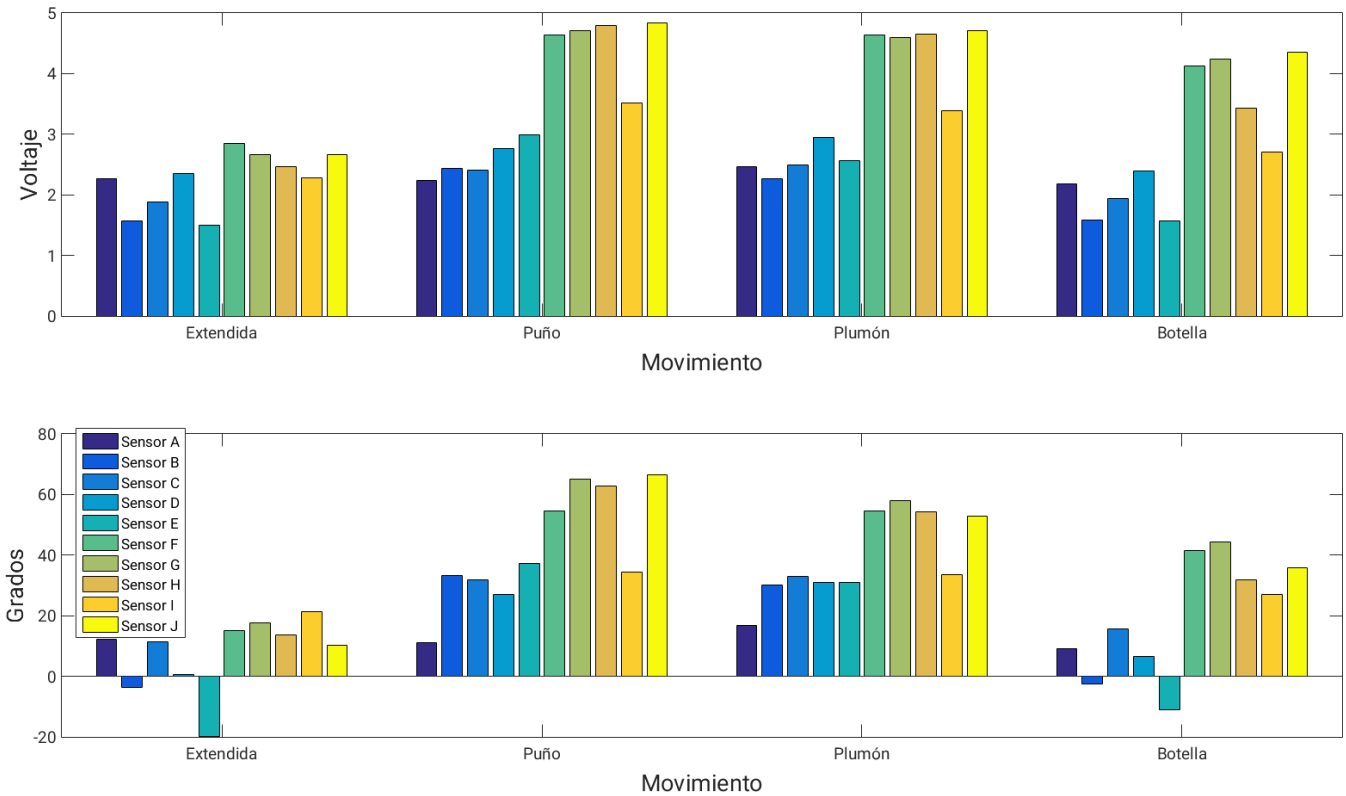


Figura 7: Mediciones obtenidas por el guante para las actividades de prueba propuestas. Superior: lecturas de voltaje de los sensores de flexión *Bend* de acuerdo a la distribución definida en la Figura 3; inferior: posición angular calculada a partir de las mediciones de los sensores de acuerdo a las curvas características interpoladas por el polinomio de ajuste de curva.

basado en un sistema embebido con el microcontrolador ATmega2560, así como un medio de almacenamiento de tecnología tipo microSD y la capacidad de generar mediciones en tiempo real utilizando un reloj externo. Lo anterior, ha permitido construir un dispositivo de bajo consumo capaz de medir posiciones angulares con alta precisión en movimientos de las articulaciones entre las falanges de cada dedo. De esta manera, se ha validado el sistema propuesto con el registro de las mediciones sobre 4 actividades básicas y comunes en la vida cotidiana, lo cual ha permitido su análisis y caracterización.

Como trabajo futuro, se propone modificar el diseño del sistema embebido del guante portable para lograr una mayor integración de los componentes en un espacio menor. Por ejemplo, componentes para la conexión inalámbrica y una IMU (unidad de medición inercial), que permitirán visualizar y analizar el comportamiento de los movimientos de la mano en el espacio tridimensional. De esta manera, es posible extender su aplicación como un sistema de registro, para el seguimiento en terapias de rehabilitación, en casos donde existan lesiones en las articulaciones interfalangianas de la mano.

AGRADECIMIENTOS

Los autores desean agradecer el apoyo del Tecnológico Nacional de México y al Instituto Tecnológico de Ensenada a través del proyecto ENS-PYR-2017-013; al Dr. Eddie Clemente y a la Dra. Marlen Meza-Sánchez por su aportación en el desarrollo del presente trabajo.

REFERENCIAS

- [1] WHO, *Rehabilitation in Health Systems*, 2017 (Accesado en Febrero de 2018). [Online]. Available: <http://apps.who.int/iris/bitstream/10665/254506/1/9789241549974-eng.pdf?ua=1>
- [2] W. H. Organization, *WHO Global Disability Action Plan 2014-2021- Better Health all People with Disability*, 2015.
- [3] Y. Zheng, Y. Peng, G. Wang, X. Liu, X. Dong, and J. Wang, "Development and evaluation of a sensor glove for hand function assessment and preliminary attempts at assessing hand coordination," *Measurement*, vol. 93, pp. 1 – 12, 2016.
- [4] G. Saggio, "A novel array of flex sensors for a goniometric glove," *Sensors and Actuators A: Physical*, vol. 205, pp. 119 – 125, 2014.
- [5] A. Asokan, A. J. Pothan, and R. K. Vijayaraj, "ARMatron: A wearable gesture recognition glove: For control of robotic devices in disaster management and human Rehabilitation," in *2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA)*, Dec 2016, pp. 1–5.
- [6] A. Rashid and O. Hasan, "Wearable technologies for hand joints monitoring for rehabilitation: A survey," *Microelectronics Journal*, pp.–, 2018.
- [7] Z. Kowalczukowalczuk and T. Merta, "Evaluation of position estimation based on accelerometer data," in *2015 10th International Workshop on Robot Motion and Control (RoMoCo)*, July 2015, pp. 246–251.
- [8] "CyberGlove," 1999 (Accesado en Marzo de 2018). [Online]. Available: <http://www.cyberglovesystems.com/>
- [9] "5DT Data Glove," p. (Accesado en Marzo de 2018), 1999 (Accesado en Marzo de 2018). [Online]. Available: <http://www.5dt.com/data-gloves/>
- [10] R. Gentner and J. Classen, "Development and evaluation of a low-cost sensor glove for assessment of human finger movements in neurophysiological settings." *Journal of neuroscience methods*, vol. 178, no. 1, pp. 138–47, mar 2009. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/19056422>
- [11] M. I. Tiwana, S. J. Redmond, and N. H. Lovell, "Sensors and Actuators A : Physical A review of tactile sensing technologies with applications in biomedical engineering," *Sensors & Actuators: A. Physical*, vol. 179, pp. 17–31, 2012. [Online]. Available: <http://dx.doi.org/10.1016/j.sna.2012.02.051>
- [12] L. K. Simone, N. Sundarajan, X. Luo, Y. Jia, and D. G. Kamper, "A low cost instrumented glove for extended monitoring and functional hand assessment," *Journal of Neuroscience Methods*, vol. 160, pp. 335–348, 2007.
- [13] N. L. Ifejika-Jones and A. M. Barrett, "Rehabilitation-emerging technologies, innovative therapies, and future objectives." *Neurotherapeutics*, vol. 8, no. 3, pp. 452–462, jul 2011. [Online]. Available: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3148149{\&}tool=pmcentrez{\&}rendertype=abstract>
- [14] Maxim-Integrated, "64x8, Serial, I²C Real-Time Clock," <https://datasheets.maximintegrated.com/en/ds/DS1307.pdf>, 2015 (Accesado en Marzo 2018).
- [15] S. symbol. (2018) Flexsensor Datasheet. [Online]. Available: <http://www.spectrasymbol.com/wp-content/uploads/2016/12/FLEX-SENSOR-DATA-SHEET-v2014-Rev-A.pdf>
- [16] F. S. S. Inc. *Mechanical Datasheet – Electrical Datasheet*, year = 2018, url = <http://www.flexpoint.com/media-resources/datasheet-downloads/>, urldate = 2018-02-08.
- [17] A. Tozeren, *Human Body Dynamics*. New York: Springer-Verlag, 2000. [Online]. Available: <http://link.springer.com/10.1007/b97432>
- [18] J. Flanagan and R. Johansson, "Hand Movements," *Encyclopedia of the Human Brain*, vol. 2, pp. 399–414, 2002.
- [19] Y. Zheng, Y. Peng, G. Wang, X. Liu, X. Dong, and J. Wang, "Development and evaluation of a sensor glove for hand function assessment and preliminary attempts at assessing hand coordination," *Measurement: Journal of the International Measurement Confederation*, vol. 93, pp. 1–12, 2016. [Online]. Available: <http://dx.doi.org/10.1016/j.measurement.2016.06.059>
- [20] M. Borghetti, E. Sardini, and M. Serpelloni, "Sensorized Glove for Measuring Hand Finger Flexion for Rehabilitation Purposes," *Ieee Transactions on Instrumentation and Measurement*, vol. 62, no. 12, pp. 3308–3314, 2013.

XIGUAR: Exoesqueleto para rehabilitación en terapia espejo

Fernando Muñoz
Gabinete de Tecnología Médica. Universidad
Nacional de San Juan – CONICET
San Juan, Argentina
fmunoz@gateme.unsj.edu.ar

Virginia Alfaro
Gabinete de Tecnología Médica. Universidad
Nacional de San Juan San Juan, Argentina
exohand4@gmail.com

Germán Grupico
Gabinete de Tecnología Médica. Universidad
Nacional de San Juan San Juan, Argentina
exohand4@gmail.com

Diego Moll
Gabinete de Tecnología Médica. Universidad
Nacional de San Juan San Juan, Argentina
exohand4@gmail.com

Marí Riquelme
Gabinete de Tecnología Médica. Universidad
Nacional de San Juan San Juan, Argentina
exohand4@gmail.com

Natalia López
Gabinete de Tecnología Médica. Universidad
Nacional de San Juan – CONICET
San Juan, Argentina
nlopez@gateme.unsj.edu.ar

Abstract—This work proposes a device that intended to complement the rehabilitation of patients who suffered a stroke with hemiplegia of the upper limb. The device consists of a 3D printed exoskeleton for the hand that is powered by servomotors, controlled by muscle signals (EMG) of the non-parasitic hand taken with a Myoware sensor and an Arduino board. The use of mirror neurostimulatory therapies is proposed under the hypothesis that proprioceptive stimulation is superior to purely visual stimulation.

Keywords—exohand; EMG; mirror therapy; stroke.

I. INTRODUCCIÓN

La terapia de espejo es una técnica utilizada en rehabilitación de personas con hemiparesia por lesiones cerebrales, constituyendo una herramienta válida para sujetos con escasa o nula actividad motora. El mecanismo que rige este fenómeno está relacionado con la realimentación visual y somatosensorial en el córtex parietal, o con la teoría de las neuronas espejo, y proporciona una alternativa terapéutica para la rehabilitación.[1]

El componente observacional de las actividades diarias, como también de las tareas de rehabilitación mejoran los efectos terapéuticos debido a la reactivación de las representaciones motoras, lo que se ha venido experimentando desde hace décadas en la terapia con espejo. En el trabajo de Celnik [2] se estimuló con terapia visual (video) a pacientes post Accidente Cerebro Vascular (ACV) examinando la

actividad cerebral con resonancia magnética funcional. Los resultados indicaron que la simple observación de la acción podría contribuir a la rehabilitación de la función motora en una mano paralizada parcialmente, lo cual involucra a más del 50% de los pacientes postACV.

En este trabajo se propone agregar a esta estimulación visual la propioceptiva, provocada por la activación y contracción muscular forzada o coadyuvada por el exoesqueleto. La hipótesis de este trabajo es que la suma de ambos componentes de representación motora en distintas áreas corticales mejorará el resultado de la rehabilitación. Esto se evalúa de la forma habitual a través de escalas funcionales como FIM, Ashworth y otras. [3]

Este trabajo está siendo desarrollado por docentes y estudiantes de la Facultad de Ingeniería de la Universidad Nacional de San Juan y está destinado a diseñar y desarrollar un dispositivo complementario de la rehabilitación de pacientes que sufrieron un accidente cerebrovascular (ACV) con hemiplejía de miembro superior. El dispositivo consta de un exoesqueleto para mano impreso en polímero y accionado por servomotores, controlado por señales musculares de la mano contralateral. Desde una óptica social, el objetivo es elaborar un prototipo de bajo costo y replicable con impresión 3D para distintos pacientes y realizar un aporte original a la rehabilitación convencional de mano y miembro superior, mejorando la motivación de los pacientes. Lleva el nombre XIGUAR, tomado del allentiac, idioma aborigen de la zona, que habla de las acciones con la mano.

II. MATERIALES Y MÉTODOS

A. Diseño del exoesqueleto

El modelo del exoesqueleto se adapta a la anatomía normal de la mano, proporcionando soporte mecánico y seguridad. Tiene como objetivo vencer las fuerzas de hipertonía provocadas por la espasticidad secundaria al ACV. Basándose en un modelo de articulaciones cilíndricas, se considera en primer lugar el movimiento de prensión esférica, ejecutada por la flexo-extensión de los dedos en un solo plano desde el meñique hasta el índice y en un plano ortogonal a éste, el dedo pulgar (Figura 1).



Fig. 1. Movimientos de flexión y extensión de los dedos de la mano.

La fuerza ejercida sobre los dedos está limitada por el sistema de control para evitar lesionar al paciente y por restricciones mecánicas. El rango de trabajo se establece en forma proporcional en base a valores sanos con respecto a la escala de Ashworth modificada, según la tabla I. Esta tabla representa los niveles de espasticidad de una mano parética y los distintos grados de resistencia que este ejerce sobre los servomotores, tomando como referencia porcentajes de movilidad de una mano sana dominante de pacientes en rangos de edad entre 45 y 70 años, edades donde el ACV es de mayor prevalencia (hombres 42,85 +/- 2,62 kg y para mujeres 29,05 +/- 3,7 kg con un rango etario 45-70 años) [4]. Asimismo, se graduó en consonancia con la escala de Ashworth modificada para facilitar la evaluación por el personal de salud.

Se diseñaron prototipos de exoesqueleto en el software Fusion 360™, en base a las medidas de una mano estándar [5], articulado de manera que permita el movimiento de los dedos en los grados de libertad considerados en la tarea de

rehabilitación. Se imprimieron dos modelos para evaluación, uno en polímero rígido y otro flexible. Los polímeros utilizados en la impresión 3D tienen certificaciones de FDA [6,7], aspecto importante ya que estarán en contacto directo con la piel del paciente. Sobre la superficie del dorso de la mano y los dedos se montaron sensores FlexiForce® de 2,2" [8], que permiten la realimentación de la posición angular del dispositivo para implementar el control. Para la elección de los sensores se tuvo en cuenta los coeficientes de elasticidad de distintos tipos de hilos de nylon y de acero. Se evaluaron las condiciones en función de su capacidad de tracción, flexibilidad para arrollarse alrededor de la guía y otras consideraciones. Se eligió trabajar con hilos de nylon de 45 mm, considerando que una fuerza superior a la aceptable provoque la ruptura del hilo antes que una posible lesión al paciente.

B. Funcionamiento

Para cumplir con la tarea de rehabilitación se sensó el movimiento de la mano sana mediante la señal electromiográfica en los músculos flexores (flexor radial y ulnar del carpo, flexor común superficial, en cara anterior y proximal del antebrazo). Para esto se utilizó un sensor Myoware® [9], de ganancia variable a través de un potenciómetro (seteada en G=50), RRM=110, y con capacidad de entregar señal cruda o envolvente. La señal se ingresa a una placa Arduino, en la que se digitaliza y se implementa el control de los servomotores.

Los servomotores digitales MG996r [10] poseen una dimensión de 40.7 x 19.7 x 42.9 mm, peso de 55 gr, velocidad de 0.17 s/60° (4.8 V), un torque de 9.4 kgf·cm (4.8 V), y rotación de 120°, con piñonería metálica, lo cual los convierte en una elección adecuada en función de tamaño, prestaciones y bajo costo. En forma solidaria al eje del motor se acopló una guía donde se arrolla el hilo de nylon que provoca la flexión o extensión de los dedos. El conjunto de servomotores, Arduino e hilos de nylon están emplazados sobre la férula, ubicada sobre el antebrazo parético e inmovilizado. La alimentación eléctrica del exoesqueleto completo, motores, la placa Arduino® y los Myoware®, se realiza mediante una batería externa portátil con 2 A/h y 5 V de salida lo que le proporciona sólo 1 hora de autonomía y cumple con los requisitos de seguridad y protección eléctrica para el paciente.

TABLA I. ESCALA DE ASHWORTH DE VALORES SANOS

| | 0 | 1 | 1+ | 2 | 3 | 4 | |
|--|---------------|--|--|---|---|--------------------------------|-------|
| Características | Normal | Ligero aumento del tono muscular, manifestado por bloqueo, prensión y liberación o por una resistencia mínima al final del arco de movimiento. | Ligero aumento del tono muscular, manifestado por prensión, seguida de resistencia mínima a lo largo del resto (menos de la mitad) del arco de movimiento. | Aumento más pronunciado del tono muscular a lo largo de la mayor parte del arco de movimiento, pero la parte afectada se mueve con facilidad. | Aumento considerable del tono muscular, el movimiento pasivo resulta difícil. | La parte afectada está rígida. | |
| Resistencia | 0 | 2 | 40 | 60 | 80 | 100 | |
| Resistencia respecto a la mano sana [Kg/cm²] | Mujer | 0 | 8,57 | 17,14 | 25,71 | 34,28 | 42,85 |
| | Hombre | 0 | 5,81 | 11,62 | 17,43 | 23,24 | 29,05 |

Para realizar el método de terapia espejo se fabricó una caja con dos secciones, una sección abierta donde se ubica la mano sana a la vista del paciente y otra cerrada donde se ubica la mano espástica ocultándola de la vista. Entre la sección abierta y cerrada se ubica un espejo que refleja la mano funcional para simular que la mano opuesta se mueve de la misma forma. La caja posee unas dimensiones de 50x60x30 cm. La Figura 2 muestra las etapas involucradas en el funcionamiento del sistema.

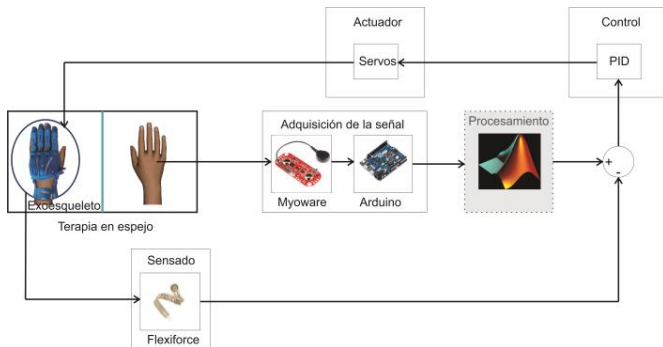


Fig. 2. Funcionamiento general del sistema. El bloque de procesamiento se utilizó sólo en las etapas preliminares, quedando luego todo el código cargado en la placa Arduino.

El usuario genera el movimiento de prensión o extensión con la mano sana y la señal electromiográfica registrada a través del Myoware® es llevada al ordenador por medio de la placa Arduino. Posteriormente la señal de EMG es procesada y acondicionada con el fin de utilizarla como señal de control de posición de los servomotores. En primer lugar, se produce la rectificación de la señal. La salida $x(n)$ representa la señal modelada de EMG cuya rectificación está dada por:

$$x_{rect}(n) = |x(n)| \quad (2)$$

Donde n es el número de muestras y $x(n)$ el valor en el instante. El siguiente paso en el procesamiento es la obtención de la envolvente de la señal de EMG rectificada. Para ello se produce el filtrado de la señal mediante un filtro de Butterworth de 5° orden, con una frecuencia de corte de 10 Hz, cuya expresión viene dada por la siguiente fórmula:

$$H(j\omega) = 1 / \sqrt{1 + \varepsilon^2 (\omega/\omega_p)^{2N}} \quad (3)$$

Donde N representa el orden del filtro, ω es igual a $2\pi f$ y ε es la máxima ganancia de la banda de paso, establecida en 20.

Finalmente se establece una umbralización de la señal para determinar el valor mínimo de la señal muscular que acciona los servomotores del exoesqueleto.

C. Control

La etapa de control del exoesqueleto se planteó en dos partes. La primera parte comprende el sistema a lazo abierto en el cual la señal acondicionada es empleada como señal de control de posición de los servomotores, estableciendo un

ángulo de giro determinado. La figura 3 muestra el diagrama de bloques de lazo abierto.

La segunda parte corresponde a la etapa de lazo cerrado, que plantea la realimentación del sistema a través de sensores Flexiforce®, cuya deformación es utilizada para inferir el grado de flexión de los dedos de la mano espástica. Varios trabajos de investigación muestran las curvas de calibración de estos sensores, evidenciando la relación que existe entre la fuerza aplicada y el voltaje de salida generado por el sensor [11,12], como así también la relación entre el ángulo de flexión y el voltaje generado. El esquema de control a lazo cerrado implica la incorporación de un controlador del tipo PID, cuyos parámetros fueron ajustados empíricamente. La entrada del controlador es la señal de error producida entre la señal de referencia (obtenida del procesamiento de la señal de EMG) y la señal de salida del bloque de realimentación, este último establece una relación entre el ángulo de giro de los servomotores y la señal obtenida de los sensores Flexiforce®, y entrega una salida de dimensiones comparables con la señal de referencia. La figura 4 muestra el esquema de control propuesto para esta segunda fase. El objetivo de este sistema es otorgar un factor de seguridad para el usuario del exoesqueleto.

D. Protocolo Experimental

Para poder evaluar el funcionamiento del prototipo se diseñó un protocolo experimental preliminar en voluntarios sanos, ya que conservan intacta la sensibilidad en ambas manos y esto permite la valoración del sistema.

Se contó con la participación de 6 adultos de entre 25 y 40 años, tres de sexo femenino y tres de sexo masculino, los cuales no presentaban ningún tipo de afección neurológica y/o motriz. Previo consentimiento informado, los voluntarios fueron instruidos para realizar las tareas sugeridas con una fuerza de prensión confortable que les permitiera cumplir con la prueba sin dificultades ni signos de fatiga muscular. Los experimentos fueron supervisados por un kinesiólogo, indicando la repetición de la tarea cinco veces por cada usuario. A fin de estandarizar el protocolo para el uso posterior en pacientes en rehabilitación, es necesario primero evaluar la capacidad de fuerza de prensión en la mano dominante, lo que luego se realizará con mediciones de espasticidad del paciente en su mano parética utilizando la escala de Ashworth citada previamente. Según el mismo serán los ángulos máximos y mínimos que podrán actuar los servomotores. La prueba se realiza con el paciente sentado con el antebrazo apoyado en la superficie de prueba con la mano en posición neutra o de reposo y se le hace contener una pelota blanda (de uso difundido en rehabilitación) para activar los movimientos de prensión. El exoesqueleto es colocado en su mano no dominante y se completa una encuesta de satisfacción al finalizar el experimento.

III. RESULTADOS

Se propusieron dos posibles prototipos con ligeras variaciones de diseño y funcionalidad. Ambos se materializaron mediante impresión 3D para ser fácilmente

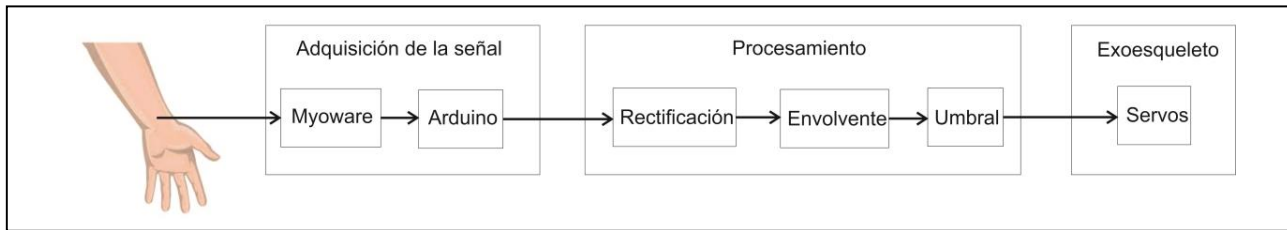


Fig. 3. Esquema de control a lazo abierto

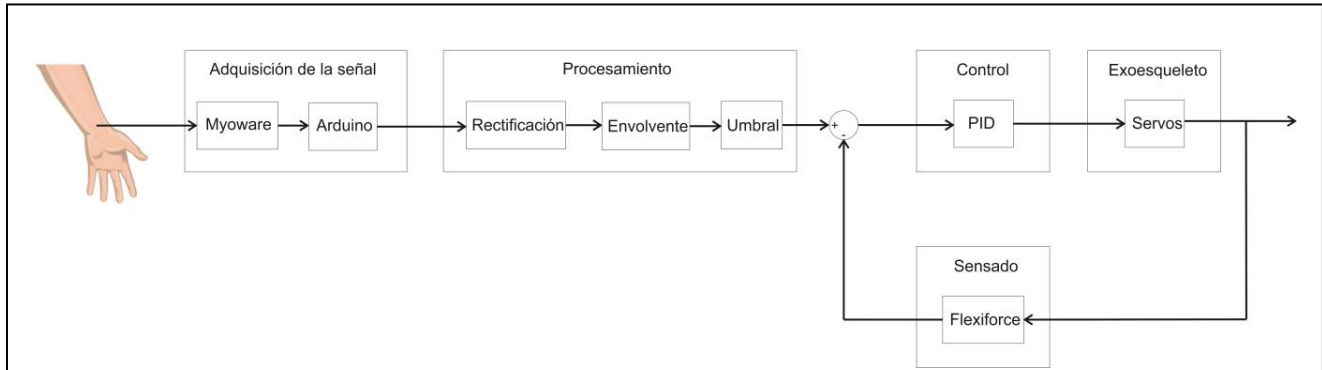


Fig. 4. Esquema de control a lazo cerrado.

replicables, baratos y adaptables para cada paciente. Los diseños son ergonómicos, lo que los hace cómodos para usarse y en un futuro podría preverse que el individuo los use de manera portable en la vida diaria y no sólo para rehabilitación. En la Figura 5 se presenta el diseño del primer prototipo, realizado en polímero rígido. Este diseño fue descartado dado que el modelo realizado en polímero flexible mostró un mejor desempeño. La Figura 6 muestra la impresión 3D del segundo prototipo donde pueden apreciarse los elementos constitutivos del exoesqueleto y los medios de sujeción al brazo del paciente, colocados en una férula sobre el antebrazo.

La Figura 7 muestra la señal mioeléctrica registrada mediante el Myoware® en la presión forzada de una pelota de rehabilitación, lo que fue usado como señal de control. Se encontró la ubicación óptima para el sensor Myoware® sobre la cara proximal interna del flexor radial del carpo midiendo la flexión de la mano y la masa se ubicó sobre la cara del epicóndilo medial del húmero. Se observa la amplitud variable en función de la fuerza muscular desarrollada, lo que permite modificar la ganancia para pacientes con bajo tono muscular.

Las señales registradas se utilizaron como comandos de control para los servomotores. Es importante destacar que el ruido de base fue descartado por la envolvente (Figura 8) y la umbralización, lo que anuló la posibilidad de deriva o contracciones no deseadas, lo que entorpecería la terapia.

Los servos fueron testeados sin carga (sólo exoesqueleto) logrando el giro de 0° o 120°. El algoritmo de control determina que la contracción muscular realiza un giro flexor proporcional a la amplitud de la envolvente, mientras que el cese de la misma provoca la relajación, y, por consiguiente, el giro inverso para volver a posición de reposo.

El exoesqueleto probado en voluntarios realizó la flexión de la mano en forma exitosa, conservando las condiciones de seguridad impuestas por el lazo de realimentación del sensor FlexiForce®. En la siguiente etapa se incorporará este sistema de sensado como un mecanismo de adaptación del prototipo al paciente. Todos los voluntarios manifestaron no sentir incomodidad ni molestias en el uso del exoesqueleto.

Se realizaron simulaciones del sistema a lazo abierto y cerrado con MatLab/Simulink a fin de ajustar los coeficientes del controlador en forma empírica y luego trasladarlo al Arduino® para garantizar la seguridad eléctrica. Las pruebas preliminares se realizaron en voluntarios sanos, bajo la supervisión de un kinesiólogo, a fin de verificar el funcionamiento y minimizar los riesgos de lesiones por fallas, por forzar la mano en condiciones de espasticidad u otras alteraciones.

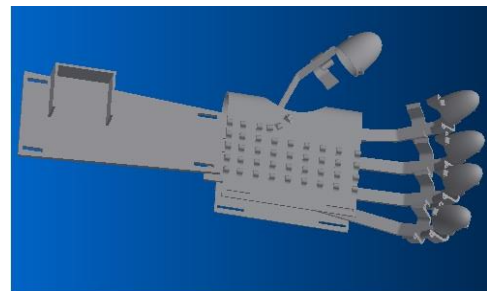


Fig. 5. Modelo del diseño en Inventor®.

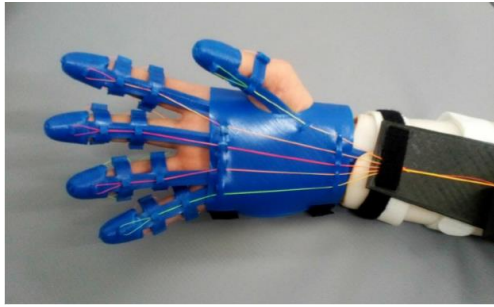


Fig. 6. Primer prototipo 3D del exoesqueleto.

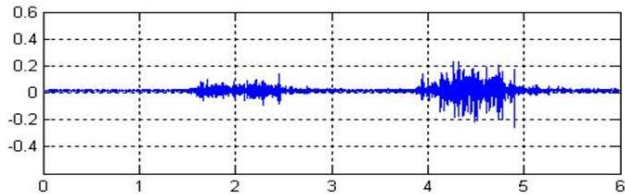


Fig. 7. señal EMG de la cara anterior del antebrazo.

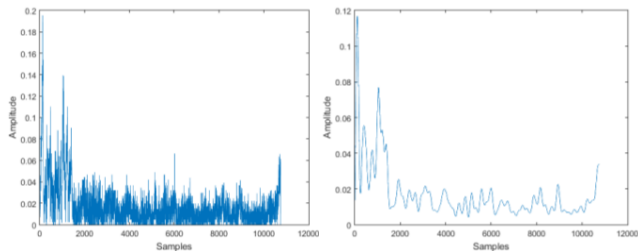


Fig. 8. Rectificación (izq.) y Envolvente (der.) de la señal de EMG.

IV. CONCLUSIONES

El prototipo desarrollado se encuentra en una etapa de diseño y ajustes, ya que la transferencia a pacientes requiere de múltiples cuidados y condiciones de seguridad. Los resultados son satisfactorios en las pruebas con voluntarios sanos, y los objetivos de selección de motores, materiales de impresión y diseño se han cumplido conservando las consignas de accesibilidad y bajo costo.

La validación con pacientes en recuperación de un ACV se comenzará en las próximas semanas, para lo cual se elaborarán protocolos experimentales específicos y escalas de valoración funcional acordes a los criterios de inclusión en cuanto a condiciones cognitivas y comunicacionales del grupo de pacientes.

El sistema de lazo cerrado se encuentra en desarrollo, trabajando particularmente en el bloque de realimentación que relaciona el ángulo de giro de los servos con las señales medidas en los sensores Flexiforce®, basado en las curvas de calibración de estos dispositivos. También se notó que se puede hacer una mejora en la durabilidad de la autonomía del sistema cambiando la batería portátil por una de mayor capacidad para poder realizar estudios más largos.

V. AGRADECIMIENTOS

Agradecemos el financiamiento de la Secretaría de Políticas Universitarias, perteneciente al Ministerio de Educación de la Nación, y al Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET).

VI. REFERENCIAS

- [1] Thieme H, Mehrholz J, Pohl M, Behrens J, Dohle C. Mirror therapy for improving motor function after stroke. *Cochrane Database Syst Rev.* 2012;CD008449. March 14, 2012. doi:10.1002/14651858.CD008449.
- [2] Celnik, P., Webster, B., Glasser, D.M., Cohen, L.G., "Effects of action observation on physical training after stroke". *Stroke.* 2008 Jun; 39(6): 1814-20
- [3] R. Cano de la Cuerda, S. Collado Vázquez, "Neurorrehabilitación: Métodos específicos de valoración y tratamiento," 1er red., Ed. Médica Panamericana: 2015.
- [4] Rojas, J. et al., "Registro de enfermedad cerebrovascular isquémica", Servicio de Neurología, Medicina (B. Aires) v.66 n.6 Buenos Aires nov./dic. 2006, Hospital Italiano de Buenos Aires. ISSN 1669-9106.
- [5] Kochol, R et al., "Accidente cerebrovascular. Prevalencia y mortalidad en un servicio de terapia intensiva", Avila Chaurand,R., Prado León, L.R., González Muñoz, E.L., "Dimensiones antropométricas de población latinoamericana". Universidad de Guadalajara 2007. ISBN: 978-970-27-1193-3
- [6] Filamentos para impresión 3D: Flex, PLA. <http://www.printalot.com.ar/filamentos/>
- [7] U.S Food and Drug Administration, <https://www.fda.gov/>
- [8] Flexiforce A201 sensor Datasheet, <https://www.tekscan.com/products-solutions/force-sensors/a201>
- [9] Sensor Myoware, hoja de datos - <https://cdn-shop.adafruit.com/product-files/2699/AT-04-001.pdf>
- [10] Hoja de datos del servo motor MG996R - http://www.electronicoscaldas.com/datasheet/MG996R_Tower-Pro.pdf
- [11] Lebossé, B. Bayle ,M. de Mathelin, P. Renaud, "Nonlinear modeling of low cost force sensors", International Conference on Robotics and Automation. Pasadena, CA, USA, pp-3437-3442. Mayo,2008
- [12] C. Lebossé, B. Bayle ,M. de Mathelin, "Modeling and evaluation of Low-Cost force sensors". *IEEE Transactions on Robotics*, vol.27,No.4, Agosto, 2011.