

Control del péndulo de Furuta aplicando Programación Genética inspirado en un controlador por modos deslizantes

Luis N. Coria^a, Ramón Ramírez-Villalobos^a, Christian Aldrete-Maldonado^a, Efraín Sanchez-Martínez^a

^a Tecnológico Nacional de México / Instituto Tecnológico de Tijuana, Departamento de Ingeniería Eléctrica y Electrónica, Posgrado en Ciencias de la Ingeniería, Blvd. Alberto Limón Padilla s/n, Tijuana B.C., México. luis.coria, ramon.ramirez, christian.aldrete18@tectijuana.edu.mx, efrain.sanchez.martinez.itt@gmail.com

Resumen

En este artículo se describe una propuesta para obtener un controlador aplicando Programación Genética utilizando series de tiempo provenientes de un controlador por modos deslizantes. Las señales para llevar a cabo la regresión simbólica se obtienen de la plataforma QNET-ROTPENT®. La búsqueda se realiza utilizando el software Eureqa®, que obtiene por regresión simbólica computacional aplicando Programación Genética diversos modelos matemáticos que representan la solución del problema equivalente a la que ofrece un controlador por modos deslizantes, reduciendo algunas características no deseadas de este tipo de controladores, como consumo de energía y oscilaciones no deseadas. De la infinidad de modelos que calcula Eureqa® se eligen de manera heurística los mejores, tomando en cuenta criterios óptimos en desempeño probados en escenarios reales; tales como complejidad del modelo, mejora de índices de desempeño y consumo de energía. Se realizan experimentos utilizando LabView® para la programación de los controladores obtenidos en conjunto con la plataforma QNET-ROTPENT®. Se logra obtener diversas propuestas de modelos matemáticos para la solución del problema, los resultados del controlador encontrado por Programación Genética, inspirado en modos deslizantes muestran un menor consumo de potencia eléctrica en comparación con el control por modos deslizantes. Se muestran tablas comparativas y gráficas para visualizar los resultados.

Palabras clave—Controlador por Modos Deslizantes, Eureqa®, Programación Genética, Péndulo de Furuta.

Controlling the Furuta pendulum via Genetic Programming inspired on a sliding mode controller

Abstract

This article describes a proposal to obtain a controller applying Genetic Programming using time series from a sliding mode controller. The symbolic regression comes from signals of a QNET-ROTPENT® platform. The Eureqa® software is a tool that provides various mathematical models from a search of computational symbolic regression using Genetic Programming. These models represent an

alternative solution to the problem equivalently to a sliding mode controller. This proposal reduces unwanted characteristics of this type of controllers, such as power consumption and chattering. From the models that Eureqa® calculates, we chose the best heuristically, taking optimal performance criteria tested in a real-world scenario, such as model complexity, performance improvement, and power consumption. We realized experiments over LabView® environment for programming the obtained controllers on the QNET-ROTPENT® platform. Several mathematical models solve the problem with better performance, the results of the GP control inspired on the sliding-modes show a lower consumption of electrical power compared to the classic PD control. Comparative tables and graphs allow visualizing of the results.

Keywords— Sliding mode controller, Eureqa®, Genetic Programming, Furuta Pendulum.

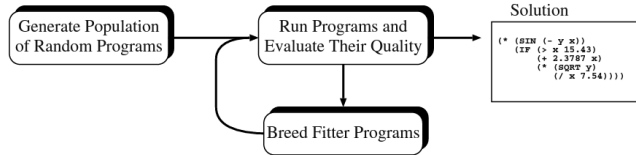
1. INTRODUCCIÓN

El control por modos deslizantes ha sido aplicado en la solución de problemas en sistemas electromecánicos, debido a que las características de elementos de potencia han evolucionado en las últimas décadas y permiten su implementación. Estos controladores conllevan una conmutación rápida (denominada *chattering* en inglés), son robustos ante perturbaciones y logran lidiar con la incertidumbre en las mediciones. Las acciones de control para un controlador por modos deslizantes son funciones discontinuas que se implementan por medio de elementos de potencia que permiten una conmutación encendido-apagado (denominado como *ON-OFF* en inglés). Un sistema por modos deslizantes implica el diseño de una superficie deslizante, la cual se comprueba con la teoría de estabilidad de Lyapunov. De esta forma se asegura la convergencia al punto de equilibrio cuando se logra que las trayectorias sistema permanezcan en la superficie deslizante. Los controladores por modos deslizantes además hacen posible que se obtenga convergencia en un tiempo finito. La comunidad de control por modos deslizantes ha desarrollado diversas propuestas que buscan evitar los problemas de *chattering*, pero en algunos casos se aumenta la complejidad del control, además la robustez y exactitud se sacrifican [1].

La programación genética (GP por sus siglas en inglés) se considera como una técnica de computación evolutiva que resuelve problemas automáticamente, sin tener que decir a la computadora como lo haga. Desde un punto de vista abstracto la GP se puede definir como un método sistemático con un dominio independiente que permite a las computadoras resolver problemas partiendo de que se tiene lograr una métrica de alto nivel. Técnicamente GP es un algoritmo evolutivo donde los individuos son programas de computadora que evolucionan de manera iterativa obteniendo nuevas poblaciones de programas que incluyen operaciones de cómputo especializadas, vea figura 1. Las principales operaciones que permiten la evolución de los programas son los denominados en inglés como *crossover* y

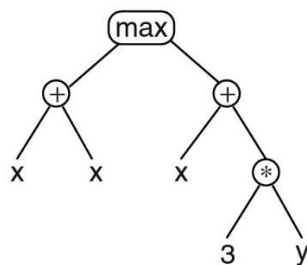
mutation. Los programas en GP usualmente se representan como árboles, ver figura 2, estos árboles representan de manera explícita el programa, donde se incluyen constantes y variables de manera específica. La GP incluye diversas técnicas y conceptos básicos para su aplicación, se recomienda consultar [2] para profundizar en el tema.

Fig. 1. Ejemplo de lazo principal de GP.



Fuente: [2]

Fig. 2. Ejemplo de programa representado como un árbol en GP, en este caso $\max(x*x, x+3y)$.



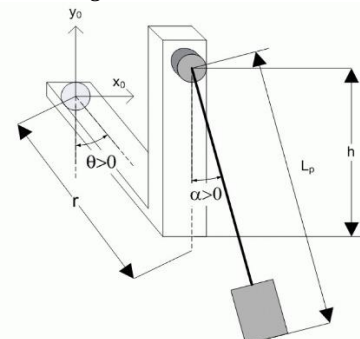
Fuente: [2]

Eureqa[®] es un software que permite la interpretación de datos obtenidos de diversos problemas y construir modelos matemáticos que permiten solucionar problemas que solamente expertos tenían la posibilidad de resolver con interpretaciones analíticas y técnicas sofisticadas. Los ciclos de solución con esas técnicas tradicionales son lentos lo que implica pérdida de ingresos en ciertas áreas de aplicación. Eureqa[®] se define como un motor de modelado que utiliza la inteligencia artificial y a su vez la GP que automatiza gran parte del trabajo pesado inherente a la ciencia de datos. El software revisa los datos para crear modelos predictivos precisos en tiempos rápidos con respecto a técnicas convencionales, construyendo modelos desde cero, partiendo de series de tiempo y métodos de clasificación, generando y actualizando modelos automáticamente. Eureqa[®] se considera un acelerador el descubrimiento de modelos basado en datos [3]. Un antecedente importante de la aplicación de este software en la formulación de modelos matemáticos no lineales en sistemas electromecánicos complejos con un alto grado de precisión [4].

El péndulo de Furuta es una plataforma con fines de estudio de modelizado matemático y diseño de controladores creada por el Dr. K. Furuta. Este sistema cuenta con dos grados de libertad y dado que solamente cuenta con un motor para realizar el movimiento, es considerado como un sistema subactuado. La estructura del sistema se puede observar en la figura 3, donde el movimiento del brazo es perpendicular a la base y el péndulo se encuentra colocado en el extremo del brazo, realizando un giro colineal al eje axial del brazo. Un motor eléctrico realiza el movimiento del brazo y se colocan

sensores para medir las posiciones angulares del brazo (*encoder* en inglés) y el péndulo. El modelo matemático del sistema se obtiene por lo general con la aplicación de las ecuaciones de Euler-Lagrange, con ecuaciones no lineales que en general son linealizadas para la solución del problema de controlar el péndulo en el punto de equilibrio inestable [5].

Fig. 3. Péndulo de Furuta.



Fuente: [5]

La plataforma *Rotary Pendulum* (ROTPEN) de Quanser cuenta con un motor de corriente directa de 24 Volts, como ya se mencionó el sistema cuenta con dos encoders, uno colocado en el eje del motor y el otro en el eje del péndulo. El movimiento del motor en esta versión está limitado a +/- 180 grados, dado que cuenta con un cable para transmitir las señales del encoder del péndulo. El sistema está integrado con la plataforma Elvis[®] de National Instrument[®], lo que permite una interacción para el diseño y sintonización de controladores con el software Labview[®]. La plataforma experimental se muestra en la figura 4.

Fig. 3. Plataforma experimental ROTPEN de Quanser.



Fuente: [5]

La propuesta del presente trabajo es el desarrollo de un controlador para el péndulo de Furuta diseñado a través de regresión simbólica aplicando Eureqa[®], basándose en señales provenientes de un controlador por modos deslizantes.

Recientemente se han logrado optimizar algunos controladores de sistemas subactuados con la ayuda de algoritmos de optimización. En [6] y [7] se optimizan los valores de las matrices Q y R para un controlador LQR aplicado a un péndulo invertido con la ayuda de algoritmos

genéticos. En [8] se hace uso de algoritmos genéticos para el auto diseño de controladores Fuzzy para un péndulo Furuta, donde se busca encontrar una configuración óptima de la base de reglas de sintonización. En [9] un control por modos deslizantes para un péndulo invertido y un robot manipulador es optimizado por medio de algoritmos de optimización para mejorar el rendimiento de seguimiento de trayectoria.

La contribución de este trabajo es una nueva metodología para encontrar la solución al problema de estabilización de un péndulo de Furuta en su punto de equilibrio inestable utilizando como base datos experimentales obtenidos de un controlador por modos deslizantes, aplicando Programación Genética para obtener una ley de control equivalente.

El resto del documento se organiza como sigue: La sección 2 incluye la descripción del modelo matemático linealizado y parámetros del péndulo de Furuta, así como la estructura de un controlador por modos deslizantes tomado de la literatura. En la sección 3 se describe de manera breve la obtención de señales y aplicación del software Eureqa® para la obtención de los modelos matemáticos. La sección 4 presenta los resultados obtenidos y la validación experimental. Finalmente, en la sección 5 se presentan las correspondientes conclusiones.

2. MODELO MATEMÁTICO Y CONTROLADOR POR MODOS DESLIZANTES PARA EL PÉNDULO DE FURUTA

Es esta sección se presenta el modelo matemático de del péndulo de Furuta y el diseño de un controlador por modos deslizantes.

2.1 Modelo Matemático del Péndulo de Furuta

El modelo matemático Euler-Lagrange del péndulo de Furuta linealizado al punto de equilibrio inestable $\alpha = \pi$. El modelo completo y el procedimiento de linealización se puede consultar en [5]. Para la representación en espacio de estados se asignan las siguientes variables (1):

$$x_1 = \theta, x_2 = \alpha, x_3 = \frac{\partial}{\partial t} \theta, x_4 = \frac{\partial}{\partial t} \alpha, \quad (1)$$

de esta forma el vector de estados queda como (2):

$$x = [x_1, x_2, x_3, x_4]^T, \quad (2)$$

con una representación dada por (3) y (4):

$$\dot{x}(t) = Ax(t) + Bu(x), \quad (3)$$

$$y(t) = Cx(t), \quad (4)$$

donde:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{rM_p^2 l_p^2 g}{\vartheta_1} & \frac{K_t K_m (J_p + M_p l_p^2)}{\vartheta_1 R_m} & 0 \\ 0 & \frac{M_p l_p g (J_e q + M_p r^2)}{\vartheta_1} & \frac{M_p l_p K_t r K_m}{\vartheta_1 R_m} & 0 \end{bmatrix}, \quad (5)$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{K_t (J_p + M_p l_p^2)}{\vartheta_1 R_m} \\ \frac{M_p l_p K_t r}{\vartheta_1 R_m} \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad (6)$$

con $\vartheta_1 = J_p J_e q + M_p l_p^2 J_e q + J_p M_p r^2$. Los parámetros de la plataforma ROTPEN se muestran en la Tabla 1.

Tabla 1. Parámetros de la Plataforma experimental ROTPEN.

Parámetro	Descripción	Valor
M_p	Masa del péndulo.	0.027Kg
l_p	Centro de masa del péndulo.	0.153m
L_p	Longitud total del péndulo.	0.191m
r	Longitud del brazo al péndulo.	0.08260m
J_m	Momento de inercia del motor.	3.00E-005Kg · m ²
M_{arm}	Masa de brazo.	0.028Kg
g	Constante gravitacional.	9.81m/s ²
J_{eq}	Inercia equivalente del motor.	1.23E-004Kg · m ²
J_p	Inercia del péndulo.	1.10E-004Kg · m ²
B_{eq}	Fricción viscosa del brazo.	0
B_p	Fricción viscosa del péndulo.	0
R_m	Resistencia del motor.	3.3 Ω
K_t	Constante de torque del motor.	0.02797N · m
K_m	Fuerza electromotriz del motor.	0.02797V(rad/s) ⁻¹

Fuente: Adaptación de [5]

2.2 Controlador por modos deslizantes

En esta sección se presenta un controlador por modos deslizantes (MD) para el sistema (3)-(4) que estabiliza la posición del péndulo en su punto de equilibrio inestable. Una forma generalizada de describir a los modos deslizantes es como un control por estructura variable que busca introducir señales de control de alta frecuencia que permiten atenuar perturbaciones y lograr el objetivo de control a pesar de las incertidumbres en las mediciones. El problema de diseño se reduce a encontrar una superficie de deslizamiento que cumpla con ciertas especificaciones. En la solución del problema, se requiere considerar que existen dos comportamientos principales, que se denominan el modo de alcance y el modo deslizante. Como su nombre lo indica, el modo alcance consiste en el modo antes de alcanzar la superficie deslizante. Cuando la trayectoria del sistema entra en la superficie deslizante, su dinámica convergerá hacia un punto de equilibrio estable.

Una manera encontrar un controlador por modos deslizantes del sistema (3)-(4) es definiendo (7) como

$$\frac{d\tilde{x}}{dt} = A\tilde{x} + Bu \quad (7)$$

donde $\tilde{x} = x - x_d$ con x_d como la posición deseada. Para encontrar una ley de control se propone una función de

Lyapunov como $V(t) = \frac{1}{2} \tilde{x}^T \tilde{x}$, cuya derivada con respecto al sistema (3)-(4) está dada por (8):

$$\frac{dV}{dt} = \tilde{x}^T A \tilde{x} + B \tilde{x} u, \quad (8)$$

se propone una entrada de control $u = -K\tilde{x} - K_s \text{sgn}(\tilde{x})$, donde $K = [k_1 \ k_2 \ k_3 \ k_4]$ es la ganancia del control lineal y $K_s = [k_{\sigma 1} \ k_{\sigma 2} \ 0 \ 0]$ es la ganancia del control discontinuo. De esta forma, sustituyendo, la derivada (8) queda como (9):

$$\begin{aligned} \frac{dV}{dt} &= \tilde{x}^T (A - BK) \tilde{x} - BK_s \tilde{x} \text{sgn}(\tilde{x}) \\ &\leq (A - BK) \|\tilde{x}\| - BK_s |\tilde{x}|, \end{aligned} \quad (9)$$

se elige K de tal forma que la matriz $A - BK$ sea Hurwitz, además $k_{\sigma 1} > 0$ y $k_{\sigma 2} > 0$, con ello se concluye que $\frac{dV}{dt} < 0$, asegurando estabilidad local del punto de equilibrio inestable.

3. DISEÑO DE CONTROLADOR POR PROGRAMACIÓN GENÉTICA

La propuesta de este trabajo buscar mejorar las características de un controlador básico por modos deslizantes utilizando GP para ello. En ese sentido, se utiliza el software Eureka® alimentando las series de tiempo de las entradas y salidas que arroja el controlador por modos deslizantes desde una implementación real. De esta forma se realizan una serie de experimentos para recolectar series de tiempo de la implementación de controlador por modos deslizantes en la plataforma ROTPEN de Quanser. Se realizan dos casos de estudio que se describen a continuación. Se utilizan las ganancias $K = [-6.5 \ 80 \ -2.75 \ 10.5]$ y $K_s = [k_{\sigma 1} \ k_{\sigma 2} \ 0 \ 0]$, seleccionadas de manera heurística durante el experimento para lograr mejor robustez que el controlador PD original.

3.1 Péndulo sin perturbar.

Esta prueba consiste en la adquisición de datos del sistema controlado por modos deslizantes, la figura 1 muestra los datos adquiridos. Los datos adquiridos se ubican de tal forma que se requiere colocar en Eureka® como entradas los estados del sistema y como salida la posición del péndulo. El Eureka® se configura para expresiones como productos, potencias, multiplicación por constantes. Después de realizar pruebas en la plataforma ROTPEN de Quanser con algunas opciones de controlador que se generan por el software, se elige (10)

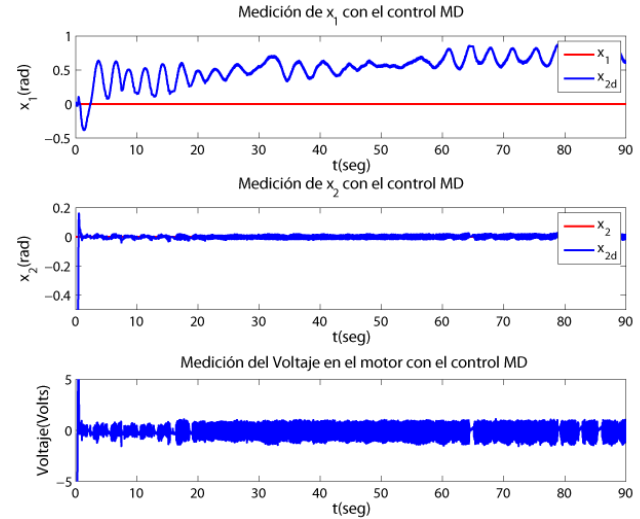
$$u_{GP1} = 143 - 3.79x_1 + 313x_2 - x_3 + 9.11x_4, \quad (10)$$

con la que se obtiene la respuesta mostrada en la figura 2.

Al observar las figuras 4 y 5, no es posible comparar de manera objetiva los resultados, sin embargo, se muestran dos

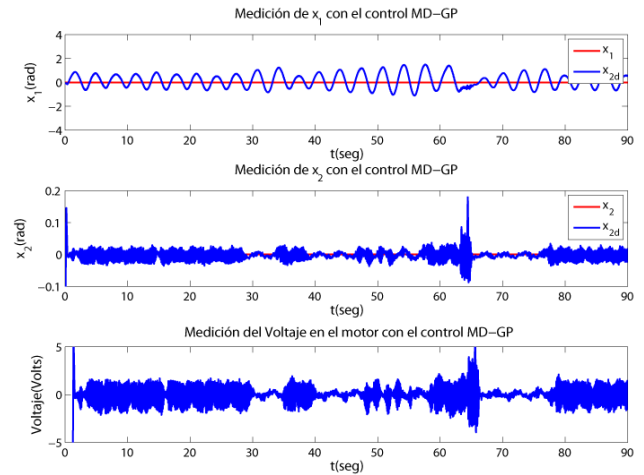
acercamientos en la figura 6, se observa una señal más pequeña por lo que el error es menor. En la figura 7 se observan las señales de voltaje del motor, con un consumo mayor del controlador obtenido por GP para este caso de estudio.

Fig. 4. Datos obtenidos al aplicar el control MD: Péndulo sin Perturbar.



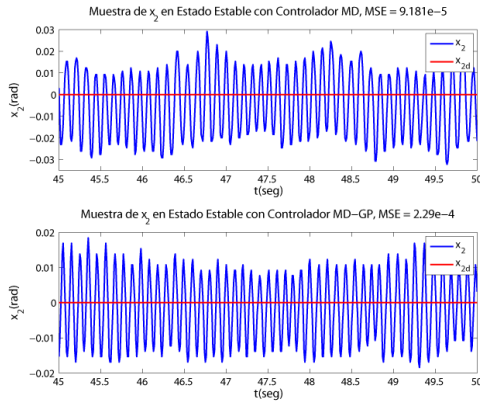
Fuente: Elaboración propia.

Fig. 5. Datos obtenidos al aplicar el control u_{GP1} (10).



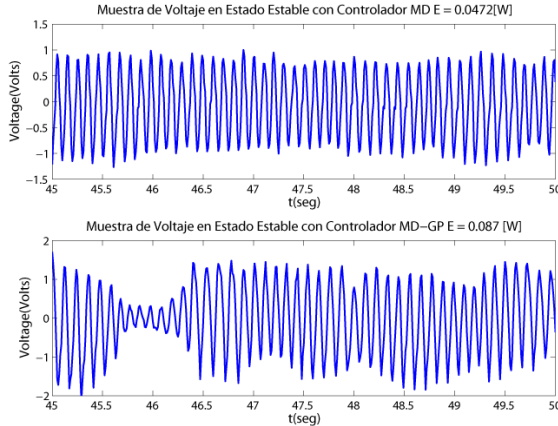
Fuente: Elaboración propia.

Fig. 6. Acercamiento de la medición del control MD y u_{GP1} (10).



Fuente: Elaboración propia.

Fig. 7. Señal de voltaje del motor con MD y u_{GP1} (10).



Fuente: Elaboración propia.

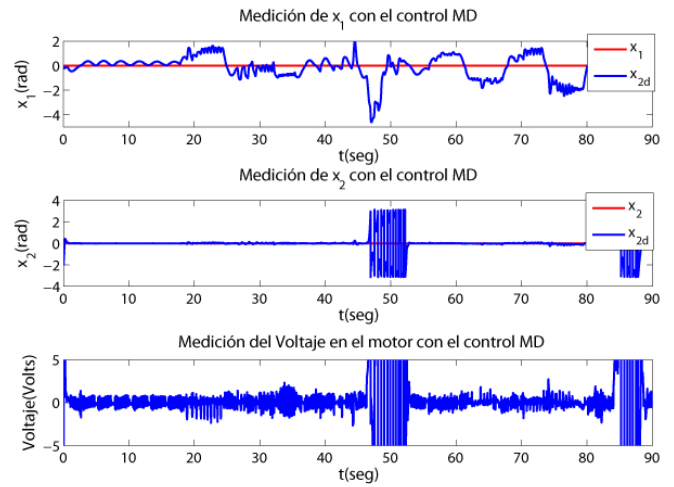
3.2 Péndulo con perturbación.

En este caso se realiza la adquisición de datos cuando se realiza una perturbación al sistema, realizando el mismo procedimiento descrito para el tratamiento de los datos y obtención del modelo del controlador descrito en la sección 3.1, de esta forma se obtiene el controlador (11)

$$u_{GP2} = 0.243 - 2.74x_1 + 158x_2 - 2.43x_3 + 11.3x_4 \quad (11)$$

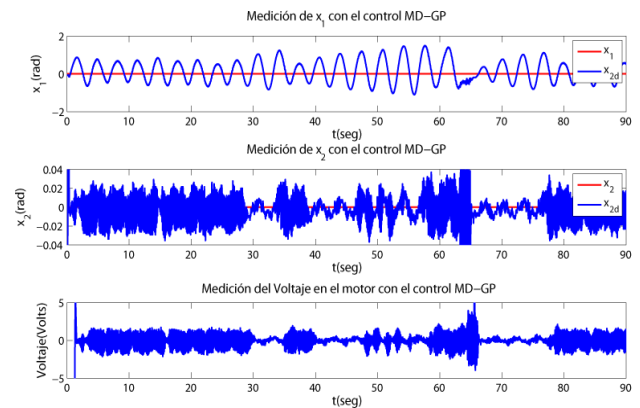
En este caso se muestran los resultados del controlador MD (usado para encontrar el controlador GP) y las señales del controlador (11) implementado en la plataforma, figura 8 y figura 9 respectivamente. Por otro lado, se muestran los acercamientos de la posición del péndulo y el voltaje aplicado al motor en la figura 10 y figura 11 respectivamente.

Fig. 8. Datos obtenidos al aplicar el control MD: Péndulo con perturbación.



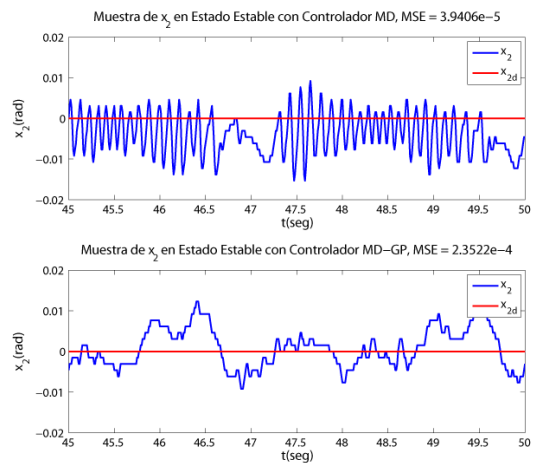
Fuente: Elaboración propia.

Fig. 9. Datos obtenidos al aplicar el control u_{GP2} (11).



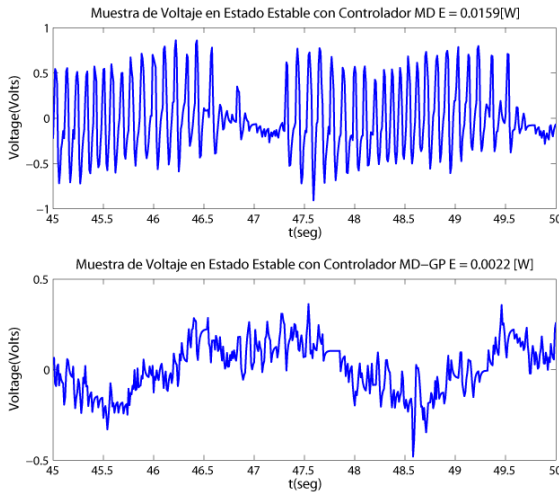
Fuente: Elaboración propia.

Fig. 10. Acercamiento de la medición del control MD y u_{GP2} (11).



Fuente: Elaboración propia.

Fig. 11. Señal de voltaje del motor con MD y u_{GP2} (11).



Fuente: Elaboración propia.

En este caso de estudio en la figura 10 se observa una reducción en el error y las oscilaciones del mismo para el controlador GP (11) con respecto al control por MD. Por otra parte, la figura 11 muestra una reducción del *chattering* y el consumo de energía del controlador GP (11) con respecto al controlador por MD. La Tabla 2 muestra las comparaciones de resultados considerando el análisis del error y consumo de energía de los casos analizados.

Tabla 2. Error RMS de la posición del péndulo x_1 y consumo de potencia del motor.

Caso	Control	Error(RMS)	Potencia[W]
No perturbado	MD	9.1865e-05	0.047255
No Perturbado	GP(10)	2.2944e-04	0.087081
Perturbado	MD	3.9406e-05	0.015970
Perturbado	GP(11)	2.3522e-05	0.002271

Fuente: Elaboración propia.

4. DISCUSIÓN Y CONCLUSIONES

En este trabajo se propuso el uso de herramientas de Programación Genética para solucionar el problema de controlar un péndulo de Furuta en su punto de equilibrio inestable. Los controladores obtenidos por Programación Genética se obtienen a partir de datos provenientes de la aplicación de un controlador por modos deslizantes que solucionaba de manera correcta el problema de control. Se plantean dos escenarios para la obtención de las señales a utilizar en el proceso de Programación Genética, el primero sin perturbar el sistema y el segundo perturbándolo. De acuerdo a los resultados, el controlador Programación Genética obtenido de datos sin perturbar mejora el error RMS de posición α del péndulo, pero aumenta el consumo de potencia del motor de corriente directa. Se considera que, dado que Programación Genética no obtiene información a priori de la planta o del modelo, es un resultado aceptable. Por otra parte, cuando se toma el caso donde se perturba el péndulo para que GP encuentre el controlador, el desempeño mejora tanto en la disminución del error RMS de posición α del péndulo como en el consumo de potencia del motor de

corriente directa. Ello se atribuye a que Programación Genética recoge las dinámicas no modeladas y las plasma en el controlador obtenido. Cuando se realizan las pruebas físicas se nota una disminución notable en las oscilaciones del sistema, lo que implica un desempeño suave ante las perturbaciones. En este trabajo no se realizó un estudio exhaustivo de las técnicas avanzadas de modos deslizantes, tampoco de Programación Genética, simplemente se intenta demostrar que existe la posibilidad de mejorar el desempeño de un buen controlador realizando una aplicación correcta de Programación Genética. Como trabajo futuro se buscará aplicar esta metodología a controladores por modos deslizantes con mejores características, así como un estudio más exhaustivo de los resultados arrojados por Programación Genética.

5. AGRADECIMIENTOS

Los autores agradecen al Tecnológico Nacional de México / IT de Tijuana, CONACYT y PRODEP por el soporte recibido. Este trabajo fue realizado como parte del proyecto TecNM 13391.21-P.

6. REFERENCIAS

- [1] Raul Rascón Carmona, Andrés Calvillo Téllez, Juan De Dios Ocampo Díaz, José Ricardo Cárdenas-Valdez, "Control por modos deslizantes mejorado," Revista Aristas: Investigación Básica y Aplicada, 4(7), pp 2-9, 2015.
- [2] Langdon, W. B., Poli, R., McPhee, N. F., & Koza, J. R. (2008). Genetic programming: An introduction and tutorial, with a survey of techniques and applications. In J. Fulcher, & L. Jain (Eds.), Computational Intelligence: A Compendium (pp. 927-1028). (Studies in Computational Intelligence; Vol. 115). DOI: 10.1007/978-3-540-78293-3_22.
- [3] Keim, Brandon. (20 de mayo de 2020). "Download Your Own Robot Scientist". Wired Magazine. <https://www.wired.com/>
- [4] M. Schmidt, H. Lipson, Distilling free-form natural laws from experimental data, Science 324 (5923) (2009) 81–85, DOI: 10.1126/science.1165893.
- [5] Quanser. Rotary Pendulum (ROTPEN) Inverted Pendulum Trainer, Instructor Manual.
- [6] N. K. Singhal and A. Swarup, "On Application of Evolutionary Algorithms for Improved Control of Inverted Pendulum," 2019 IEEE 1st International Conference on Energy, Systems and Information Processing (ICESIP), 2019, pp. 1-5, DOI: 10.1109/ICESIP46348.2019.8938379.
- [7] H. Chen, Y. Yang and J. Sun, "Improved Genetic Algorithm Based Optimal Control for A Flying Inverted Pendulum," 2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE), 2019, pp. 1428-1432, DOI: 10.1109/EITCE47263.2019.9094769.

- [8] V. Stanovov, S. Akhmedova and E. Semenkin, "Automatic Design of Fuzzy Controller for Rotary Inverted Pendulum with Success-History Adaptive Genetic Algorithm," 2019 International Conference on Information Technologies (InfoTech), 2019, pp. 1-4, DOI: 10.1109/InfoTech.2019.8860874.
- [9] N. Bordoloi, "A Natural Search Algorithm Based Optimized Sliding Mode Controller for Inverted Pendulum and Robot Manipulator System," 2018 2nd International Conference on Power, Energy and Environment: Towards Smart Technology (ICEPE), 2018, pp. 1-6, DOI: 10.1109/EPETSG.2018.8658800.