

ALGORITMO GENÉTICO PARA OPTIMIZAR LAS GANANCIAS DE UN CONTROLADOR PI EN UN MOTOR DE CD

Alberto García Pérez

Tecnológico Nacional de México en Celaya

13030298@itcelaya.edu.mx

Roberto Toral Herrera

Tecnológico Nacional de México en Celaya

13030283@itcelaya.edu.mx

Alonso Alejandro Jiménez Garibay

Tecnológico Nacional de México en Celaya

alonso.jimenez@itcelaya.edu.mx

Carlos Gerardo Euresty Uribe

Tecnológico Nacional de México en Celaya

gerardo.euresty@itcelaya.edu.mx

Resumen

Este artículo presenta el diseño de un algoritmo genético (AG) para la optimización de las ganancias de un controlador PI para aplicaciones de control de un motor de CD. A diferencia de los AGs existentes en la literatura, el AG propuesto utiliza como variables de interés el tiempo de asentamiento y el sobreimpulso. La implementación del sistema incluye pruebas de laboratorio con un motor de corriente directa de 1 HP y un sistema de torque controlado.

Palabra(s) Clave: Algoritmo genético, Controlador PI, Motor de CD.

Abstract

This paper presents the design of a Genetic Algorithm (AG) for the optimization of the gains of a PI controller for control applications of a CD motor. In contrast to the AGs in the literature, the proposed AG uses settlement time and overshoot as variables of interest. The system implementation includes lab tests with a 1 HP direct current motor and a torque controlled system.

Keywords: DC motor Genetic algorithm, PI controller.

1. Introducción

En cuanto al diseño del controlador, en la literatura se reportan múltiples investigaciones referentes al control de motores eléctricos, siendo la más común el uso de controladores tipo PID [1]. Un caso particular, son los sistemas donde se requiere controlar varios motores bajo condiciones diferentes, lo que dificulta la sintonización de las ganancias, lo que hace necesario el uso de herramientas tecnológicas que permitan optimizar.

Una técnica de inteligencia artificial utilizada para optimizar las ganancias de los controladores PID, son los algoritmos genéticos cuyo principio es la teoría de la evolución [2]. En la literatura se han reportado diferentes tipos de algoritmos, algunos inspirados en el comportamiento mismo de la naturaleza o de las especies, [3-4].

Este artículo presenta el diseño de un algoritmo genético, que utilice como variables de interés: el tiempo de asentamiento y el sobreimpulso para optimizar las ganancias de un controlador PI, cuyos perfiles de par y velocidad dependen de las condiciones y la función a desempeñar.

2. Métodos

Modelo de un motor de CD

La figura 1 presenta el modelo típico de un motor de corriente directa, mientras que, la figura 2 muestra el diagrama de bloques que determina el comportamiento de la velocidad o la posición, en función del voltaje aplicado [5].

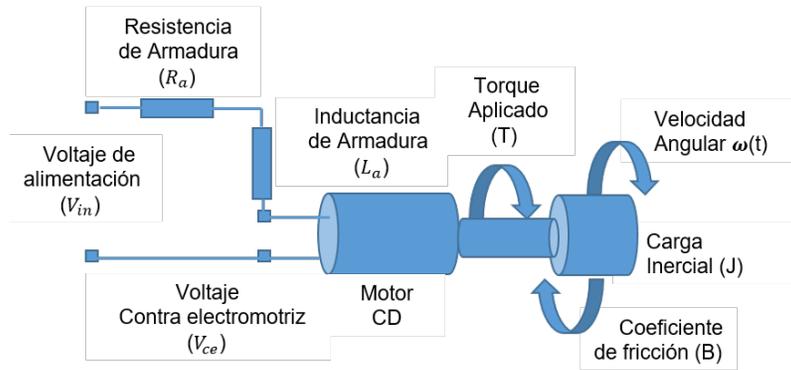


Figura 1 Modelo típico de un motor de CD.

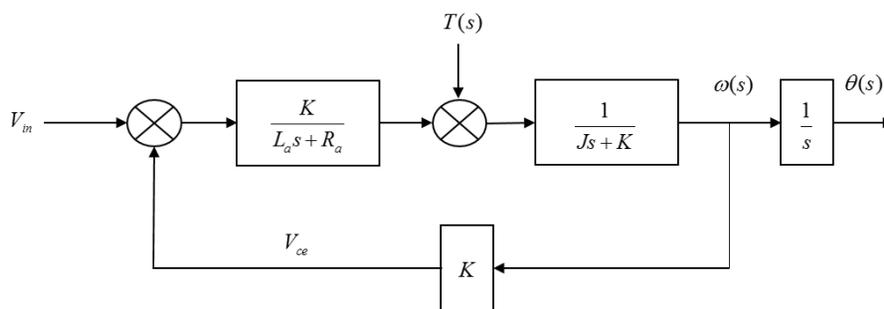


Figura 2 Diagrama de bloques para un motor de CD.

Métodos de parametrización

Para obtener los valores, se utilizó la herramienta de Matlab-Simulink “estimation parameter” [6]. La figura 3 muestra los valores estimados para cada una de las variables que intervienen en el modelo del motor.

Estimation parameters						
Parameter	Value	Estimate	Initial Guess	Minimum	Maximum	Typical Va...
J	0.00058037	<input checked="" type="checkbox"/>	J	0	+Inf	J
K	0.39421	<input checked="" type="checkbox"/>	K	0	+Inf	K
L	0.015046	<input checked="" type="checkbox"/>	L	0	+Inf	L
R	2.2776	<input checked="" type="checkbox"/>	R	0	+Inf	R
b	0.00079239	<input checked="" type="checkbox"/>	b	0	+Inf	b

Figura 3 Estimación de parámetros – Matlab-Simulink Estimation Parameters.

A través de una reducción de bloques y los valores obtenidos de la estimación, es posible determinar la ecuación 1, que describe el comportamiento de la velocidad con respecto al voltaje de alimentación.

$$\frac{\omega(s)}{V_{in}(s)} = \frac{0.394}{(0.015s + 2.27)(0.00058s + 0.00079) + 0.394^2} \quad (1)$$

Sin embargo, si se considera la dinámica del motor de CD, y de acuerdo con literatura existente, es posible despreciar la parte eléctrica, fundamentado principalmente por el criterio de polos dominantes, reduciendo la ecuación 1 a un modelo de primer orden, La figura 4 muestra los resultados de la estimación a través de la herramienta de Matlab “Ident” [7] y reflejados en la ecuación 2.

$$\frac{\omega(s)}{V_{in}(s)} = \frac{369.60}{s+147} \quad (2)$$

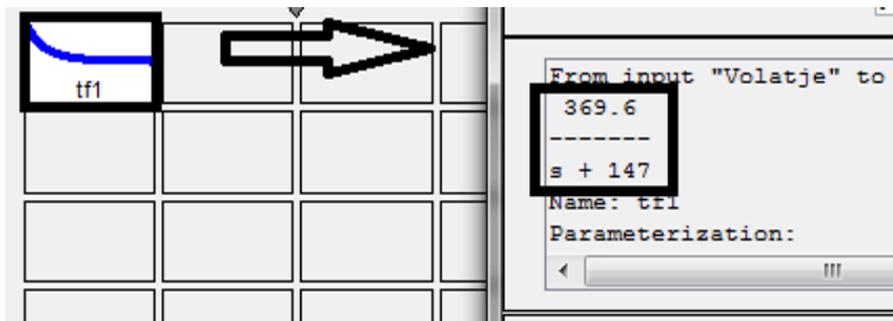


Figura 4 Estimación de parámetros – Matlab Ident.

Es importante hacer notar que las herramientas de Matlab utilizadas no concluyen en valores reales de las variables que intervienen en el modelo matemático. Sino que realizan mediante métodos numéricos una estimación de parámetros.

Los datos estimados a través de las herramientas computacionales antes mencionadas, fueron realizados a partir de una prueba experimental a través de un escalón de 70 VCD. La tabla 1 muestra las características de placa del motor de prueba.

Tabla 1 Características eléctricas de placa del motor de prueba.

Nombre de la variable	Valor de placa y unidad
Voltaje de armadura (Va)	90 Volts
Potencia	1 H.P.
Corriente nominal	2.3 A

Diseño del controlador

En la literatura existen gran variedad de métodos de sintonización de controladores PID. Uno de los métodos más utilizados es Ziegler- Nichols [8]. Sin embargo, las no linealidades que presentan los motores eléctricos, referentes a su construcción, su vida útil y los esfuerzos inducidos que degradan sus propiedades, generan la necesidad de ajustar la sintonización a través de la experimentación. En la tabla 2 se muestran las ganancias obtenidas con el método se Ziegler-Nichols en lazo abierto y las ganancias que se obtuvieron con el método experimental, partiendo de las ganancias obtenidas con Ziegler-Nichols.

Tabla 2 Ganancias obtenidas con los métodos de Ziegler-Nichols y experimental.

Método	Kp	Ti
Ziegler-Nichols	0.62	0.292
Sintonización experimental	0.223	0.001

Aunque los métodos utilizados brindan un desempeño aceptable, no son las óptimas. La siguiente sección trata sobre el diseño del AG que permita encontrar los valores óptimos para el controlador.

Diseño del AG propuesto

Un algoritmo genético típico, parte de una población aleatoria de posibles soluciones y por medio de la evaluación se obtiene una solución muy aproximada a la ideal, la figura 5 muestra un diagrama de flujo propuesto.

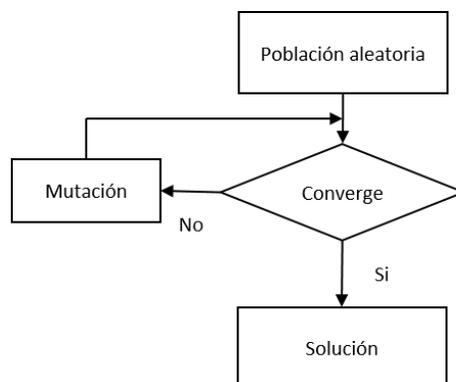


Figura 5 Diagrama de flujo del AG propuesto.

El algoritmo genético utilizado es un algoritmo básico, sin embargo, una aportación importante es la utilización de una red neuronal que permite obtener la eficiencia del AG. La figura 6 muestra la interfaz del programa que ejecuta la red neuronal. La red es de tipo BPN (backpropagation) de 3 capas, este tipo de red consiste en el aprendizaje de un conjunto predefinido de pares de entradas-salidas dados. Donde las entradas son las ganancias K_p y T_i con diferentes valores propuestos y como salidas se tienen el sobreimpulso y el tiempo de asentamiento. Es decir la capa 1 son las entradas y en la capa 2 se distribuye la información para poder generar una salida que se aproxime al valor real de las salidas obtenidas.

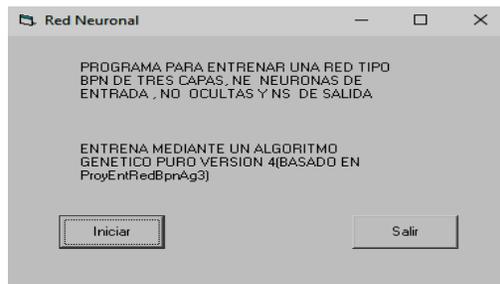


Figura 6 Interfaz gráfica para de la red neuronal.

La función se dividió en dos partes, el tiempo de sobrepaso y el tiempo de asentamiento, de esta manera se obtuvieron dos funciones, cuyos resultados generan la función completa.

La evaluación de la eficiencia de las funciones de las figuras 7 y 8 muestran un error de 0.092 para el sobre impulso y 0.017 para el tiempo de asentamiento, valores que de acuerdo a la literatura validan la expresión obtenida para la eficiencia del AG, ya que se reportan rangos validos de 0-0.1.

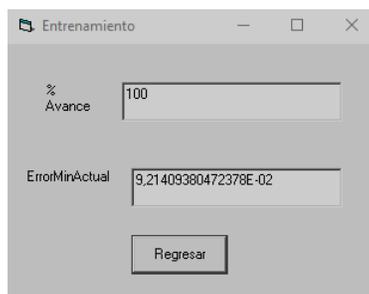


Figura 7 Evaluación del error de sobre impulso.

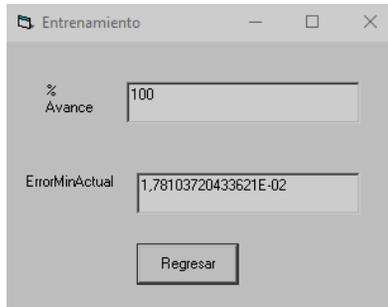


Figura 8 Evaluación del error del tiempo de asentamiento.

3. Resultados

Plataforma de experimentación

Para la obtención de resultados experimentales que permitan validar el AG, la figura 9 muestra el diagrama de bloques del sistema completo la plataforma, conformada mediante un driver de 500 Watts, basado en la topología denominada PWM IGBT H-Bridge, una interfaz gráfica desarrollada en NI LabVIEW que permite: modificar los valores del controlador y monitorear el setpoint y el voltaje de alimentación, tal como lo muestra la figura 10 y un motor de CD acoplado a un sistema de torque controlado.

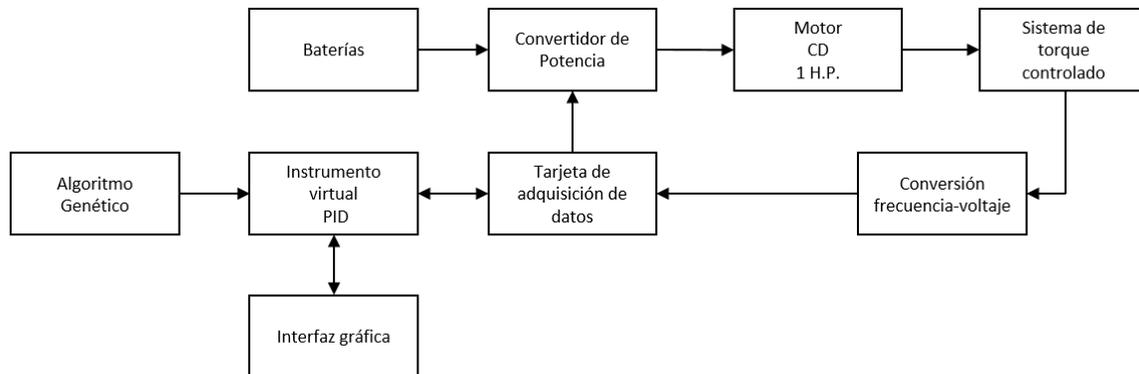


Figura 9 Diagrama de bloques de la plataforma.

La figura 11 muestra la plataforma experimental descrita anteriormente, controlando el torque que se le aplica al eje del motor de CD es posible emular diferentes comportamientos, tal como los tendría el sistema ante diferentes escenarios.

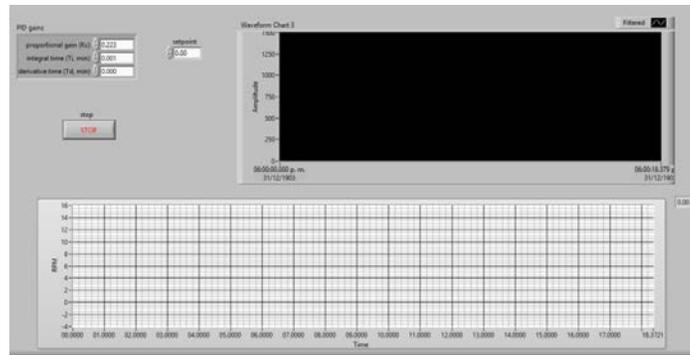


Figura 10 Interfaz gráfica para el monitoreo del setpoint y la variable de control.



Figura 11 Plataforma experimental.

Pruebas

La figura 12 presenta la pantalla de inicio del AG, desarrollado en el software Microsoft Visual Studio. La figura 13 muestra la interfaz del ejecutable, la cual se compone de los 2 cromosomas más aptos de la población generada, tanto en binario como en decimal además de la eficiencia de cada una de estas posibles soluciones.

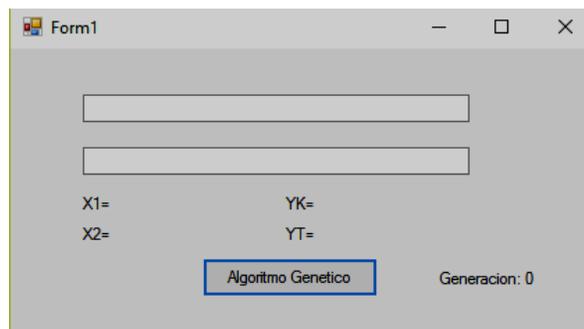


Figura 12 Pantalla de inicio antes de generar la solución.

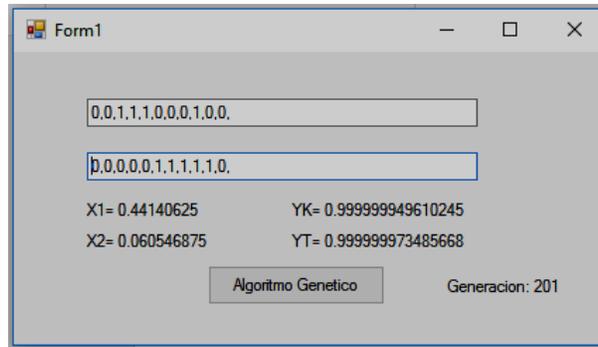


Figura 13 Interfaz con la solución.

El algoritmo genético genera una población inicial (Primera generación) aleatoria de 100 pares de elementos en un rango de valores permitidos (de 0 a 1).

Cada par generado contiene un valor para la variable K (Constante proporcional) y el Ti (Constante integral), los cuales son codificados en una cadena binaria, llamada cromosoma. Los cromosomas evolucionan a través de iteraciones, llamadas generaciones. En cada generación, los cromosomas son evaluados usando las funciones de eficiencia que se obtuvieron por medio de la red neuronal. Las siguientes generaciones (nuevos cromosomas), son generadas aplicando operadores genéticos repetidamente, siendo estos los operadores de selección, cruzamiento (Crossover), mutación y reemplazo.

Cuando la eficiencia que se calcula a través de las funciones es igual a uno, se detiene el AG dando como resultado el cromosoma con el cual la eficiencia fue más alta, pero cuando la función de eficiencia no obtiene valores cercanos a uno se detiene el AG después de correr un número determinado de generaciones y mostrando el mejor cromosoma candidato hasta el momento.

Los valores obtenidos se insertaron en el instrumento virtual, la tabla 3 muestra 10 pruebas con diferentes valores para las ganancias proporcional (K) y el tiempo integral (Ti). El sobreimpulso registrado en rpm y el tiempo de asentamiento en segundos se muestran en la cuarta y quinta columna, siendo la mejor solución la prueba 6, con valores de 2.7 segundos de tiempo de asentamiento y sin sobre impulso, ante una carga de 1 N y un escalón de 70 V equivalente a 1200 rpm.

La figura 14 muestra los resultados exportados, la gráfica 14a es el comportamiento de las ganancias obtenidas al sintonizar el sistema, la gráfica 14b

corresponde a las ganancias de la prueba 6 y por último, la gráfica 14c presenta el mejor comportamiento, de acuerdo a los requisitos de diseño.

Tabla 3 Valores obtenidos del AG propuesto.

Prueba	K	Ti	Sobre impulso (rpm)	Tiempo de asentamiento (s)
P1	0.441	0.006	0	4.5
P2	0.613	0.007	2.19	3.4
P3	0.447	0.006	2.14	4.3
P4	0.568	0.005	0.63	3
P5	0.477	0.005	0	3.2
P6	0.388	0.004	0	2.7
P7	0.494	0.006	0	4.5
P8	0.503	0.005	0	3.4
P9	0.585	0.006	0	3.3
P10	0.464	0.006	0	4.4
Sintonizado	0.223	0.001	149.3	1.4
Optimizado	0.388	0.002	1.34	1.2

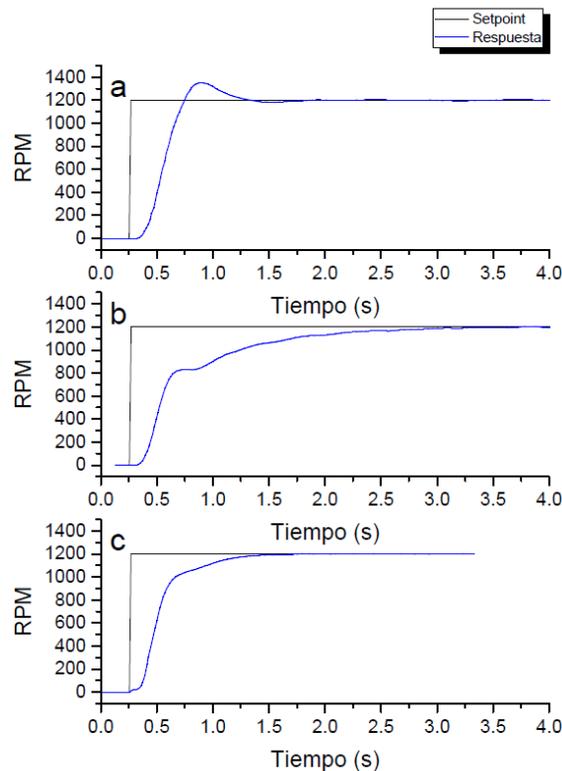


Figura 14 Resultados exportados.

Las condiciones de experimentación fueron: una carga de 1 N y un setpoint de 1200 rpm. En la gráfica 15a, se muestra la respuesta del sistema con ganancias $k=0.223$, $T_i=0.001$, obteniendo un sobreimpulso igual a 149.3 rpm y un tiempo de asentamiento de 1.4 s. La gráfica 15c representa la parte optimizada por el algoritmo genético, podemos observar la mejora que existe tanto en el sobreimpulso y el tiempo de asentamiento, con un sobreimpulso igual a 0 y su tiempo de asentamiento es igual a 1.2 s, 0.2 s más rápido que el sintonizado, además de una mejora en el sobreimpulso, por que no presenta ninguna variación.

4. Discusión

El uso de inteligencia artificial, es sin duda una directriz global en lo que refiere a control y optimización. Este artículo exploró el uso de un AG para optimizar las ganancias de un controlador PI para un motor de CD.

Sin duda, la mayor aportación de esta investigación, es considerar el sobreimpulso y el tiempo de asentamiento como variables de diseño del controlador, lo que provocó la incursión de una red neuronal para la obtención de la eficiencia, expresión vital para el funcionamiento adecuado del AG.

Los resultados validaron la aportación, sin embargo, se pretenden realizar mejoras futuras que potencien la investigación, tales como:

- Pruebas dinámicas para evaluar el desempeño del control PI con las ganancias optimizadas con condiciones de setpoint y carga variable.
- Comparación con técnicas de sintonización comerciales para controladores PID.

5. Bibliografía y Referencias

- [1] Guohan Lin, Guofan Liu "Tuning PID Controller Using Adaptive Genetic Algorithms," The 5th International Conference On Computer Science & Education, 2010, pp 519-523.
- [2] Nayana Patil, Girish Lakhekar, "Improved performance of cascade control system using genetic algorithm," Proceedings of WRFER International Conference 16th, 2017, pp 80-87.

- [3] Son Yan, Yang Guoxing “A genetic algorithm of test paper generation,” The 8th International Conference on Computer Science & Education., 2013, pp 897-901.
- [4] Wang Jingfang “Immune genetic algorithm for solving nonlinear equation,” International Conference on Mechatronic Science, electric and computer, 2011, pp 2094-2097.
- [5] Norman S. Nise, “Modeling in the frequency domain” in Control System Engineering 6ta. Ed. John Wiley & Sons, Inc. 2010, pp 33-97.
- [6] <https://www.mathworks.com/discovery/parameter-estimation.html>
- [7] <https://www.mathworks.com/products/sysid.html>
- [8] Katsugiko Ogata, “Controladores PID y sistemas de control con dos grados de libertad,” Ingeniería de control moderna., 4ta. Ed. Pearson Prentice Hall, 2003, pp. 641–724.