

Recibido: 20 de septiembre de 2011/ Aceptado: 3 de enero de 2012
Publicado en línea: 30 de marzo de 2012

Sintonización de un Controlador Proporcional-Integral-Derivativo: Una Comparación entre Dos Algoritmos Evolutivos

¹Juan Fernando García Mejía ^{1,2}, José Antonio García Mejía¹, Citlali Y. Gutierrez Estrada³

¹ Centro Universitario UAEM Atlacomulco. Carretera Toluca Atlacomulco Km 60
Atlacomulco México.

² Universidad Popular Autónoma del Estado de Puebla. 21 sur 1103 Barrio Santiago C.P.
72410, Puebla, México

³ Instituto Tecnológico de Toluca. Av Tecnológico Ex Rancho la Virgen Metepec Mex
Carretera Toluca Atlacomulco Km 60 Atlacomulco México.
fgarciam@uaemex.mx

Resumen. Una de las aplicaciones de las tecnologías de la información se encuentra en el control automático, generando un nuevo paradigma denominado control inteligente, el cual emula las características de sistemas biológicos, permitiendo crear controladores con nuevas capacidades que suelen convertirse en productos que se constituyen como importantes ventajas competitivas para las compañías de tecnología. Con esto en mente en este artículo se propone calcular los parámetros de un controlador del tipo Proporcional-Integral-Derivativo, mediante algoritmos evolutivos, específicamente el Algoritmo inmunológico clonal y el Enjambre de partículas, constituyéndose como alternativa al ajuste de Nichols, convirtiendo un problema de diseño de controladores en uno de optimización. Cabe destacar que se utilizó LabView para implementar los algoritmos antes mencionados

Palabras clave: Controlador Proporcional-Integral-Derivativo, Enjambre de partículas, Algoritmo inmunológico clonal.

Abstract. One of the applications of information technology is in automatic control, a new paradigm called intelligent control, which emulates the characteristics of biological systems, allowing you to create controllers with new capabilities that are becoming products that are as important competitive advantage for technology companies. With this in mind in this article is to estimate the parameters of a Proportional-Integrated-Derivative controller using evolutionary algorithms, specifically the immune clonal algorithm and particle swarm, establishing itself as an alternative to setting Nichols, turning a controller design problem in an optimization. Note that we used LabView to implement the above algorithms

Keywords: PID controller, swarms of particles, immune clonal algorithm

¹ Autor de correspondencia: fgarciam@uaemex.mx

1. Introducción

Un signo del progreso de la ingeniería de control es el desarrollo de redes de sensores y actuadores, algoritmos de computo distribuido e híbridos lo que ha dado un nuevo significado al termino “sistemas dinámicos”. Estos solían ser modelados mediante ecuaciones diferenciales ordinarias, con algunas incertidumbres en los parámetros ruido o disturbios externos, actualmente es posible integrar a estos factores tareas dinámicas, sensores y actuadores reconfigurables, detección y aislamiento de fallas así como cambios estructurales en el modelo de la planta, lo cual plantea una reconsideración en la forma de estimar y adaptar y controlar fenómenos físicos, cuya alternativa se encuentra en el uso de las tecnologías de la información[1], que es un término de uso general que hace referencia a todo lo que involucre una computadora , cuya complejidad obliga a dividir su estudio en software y hardware [2], generando así una nuevo paradigma en la teoría de control denominado “control inteligente”, el cual emula las características de sistemas biológicos, permitiendo crear controladores con nuevas capacidades que suelen convertirse en productos que se constituyen como importantes ventajas competitivas para las compañías de tecnología, esto a través de las siguientes herramientas [3]:

1. Lógica difusa (FL, por sus siglas en inglés) que se basa en la experiencia de un operador humano expresado en forma de lógica de predicados
2. Redes neuronales artificiales (NNA, por sus siglas en inglés) que emulan el

proceso de aprendizaje de la neurona biológica a través de métodos de entrenamiento ya sea supervisado o no supervisado

3. Métodos Predictivos que son técnicas matemáticas que proveen de información sobre el comportamiento futuro del sistema
4. Algoritmos Evolutivos que se basan en procesos tales como la evolución natural, estos esencialmente se utiliza en la optimización de procedimientos.

Ahora bien, los algoritmos evolutivos (EA, por sus siglas en inglés) son series de pasos definidos con aplicación en la optimización o de aprendizaje con la capacidad de evolucionar. Estos tienen tres características principales [4]:

- Basado en la población: Un EA mantiene un conjunto de soluciones, llamada población, para optimizar o aprender el problema en forma paralela. Este es un principio básico del proceso evolutivo.
- Fitness-orientado: Todas las soluciones en una población se llama individuo, tiene una representación llamada código y la evaluación de desempeño se denomina aptitud (o fitness). Un EA privilegia a los individuos más aptos, que permitan la optimización y la convergencia de los algoritmos
- Operadores de combinación: Los individuos de una población que se someten a una serie de operaciones con la finalidad introducir variaciones en su código, esto es fundamental para buscar un espacio de soluciones

Uno de los primeros algoritmos evolutivos fue el algoritmo genético simple (GA, por sus siglas en inglés), fue desarrollado por John Holland en la década de los 60 en la Universidad de Michigan, basado en los principios de Charles Darwin y Gregory Mendel presentes en la naturaleza [5].

Dentro de la clasificación de los algoritmos evolutivos se encuentra el sistema inmunológico artificial (AIS) que se encuentra inspirado en los principios de funcionamiento del sistema defensivo de los mamíferos superiores ante enfermedades, esta idea fue propuesta por Doynne Farmer en 1986 tomando una importancia significativa a finales de la década de los 90 [6].

La optimización por enjambre de partículas (PSO, por sus siglas en inglés) es otra técnica de los algoritmos evolutivos desarrollada por James Kennedy y Russ Eberhart en 1995, se basa en imitar el comportamiento de enjambres de aves que intenta buscar comida en un área desconocida. Tienen una estructura aritmética simple una alta velocidad de convergencia y habilidad para optimización globales [7].

En base a la información anteriormente presentada se propone en este artículo realizar una optimización de un controlador del tipo PID diseñado en el dominio discreto, de una planta teórica, mediante un sistema inmunológico artificial y de un enjambre de partículas.

2. Fundamentación teórica

En esta sección la información teórica que soporta el presente desarrollo; la teoría de los controladores del tipo Proporcional-

Integral-Derivativo, la de los sistemas inmunológicos artificiales y del enjambre de partículas.

2.1 Controlador Proporcional-Integral-Derivativo

La teoría de control es una rama de la ingeniería que se encarga de mejorar el desempeño de dispositivos físicos (velocidad de motores, temperatura, posición de dispositivos mecánicos, por mencionar algunos) a una cierto estímulo de entrada, mediante controladores, los cuales pueden ser de varios tipos. La figura 1 ilustra un esquema de control clásico en el dominio discreto, donde $C(z)$ es el controlador, $G(z)$ es el fenómeno físico también denominado planta y $R(z)$ es el estímulo de entrada

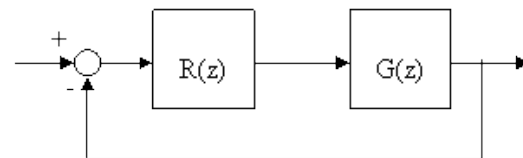


Fig. 1. Esquema de control clásica a lazo cerrado, donde se muestran el controlador, la planta y la referencia

A pesar de la abundancia de sofisticadas herramientas y métodos avanzados de control, el controlador Proporcional-Integral-Derivativo, (PID) es uno de los más utilizados en la industria moderna. En la actualidad, muchos de éstos se transforman en formas digitales mediante el uso de microprocesadores. [8]. La expresión matemática de un controlador PID en términos de z^{-1} es mostrado en la ecuación 1

$$C(z) = K_p + \frac{K_i T}{2} \frac{z+1}{z-1} + \frac{K_d}{T} \frac{z-1}{z} \quad (1)$$

T es el tiempo de muestreo

K_p es la constante proporcional

K_i es la constante integral

K_d es la constante derivativa

Ziegler y Nichols sugirieron varias reglas para sintonizar los controladores PID (lo cual significa establecer valores K_p , K_i , K_d con base en las respuestas escalón experimentales o basadas en el valor de que se produce en la estabilidad marginal cuando sólo se usa la acción de control proporcional. Las reglas de Ziegler-Nichols son convenientes cuando no se conocen los modelos matemáticos de las plantas.

Las reglas propuestas por Ziegler y Nichols determinan los valores de la ganancia proporcional K_p , K_i , K_d del tiempo integral y del tiempo derivativo, con base en las características de respuesta transitoria de una planta específica.

2.2. Sistema inmunológico artificial

El sistema inmune ha servido como inspiración para solucionar problemas complejos de ingeniería y la ciencia con gran éxito, debido principalmente a que es un sistema de aprendizaje distribuido con interesantes características. Una de las principales tareas del sistema inmune es mantener al organismo sano. Así, debe ser capaz de reconocer y eliminar microorganismos que son patógenos para el sistema [9].

Esta respuesta defensiva del sistema inmune presenta interesantes características desde el punto de vista del procesamiento de información. Es por ello que se ha usado como inspiración para

crear soluciones alternativas a problemas complejos de ingeniería y la ciencia. Esta es un área relativamente nueva a la cual se le llama sistema inmune artificial [10].

El sistema inmune artificial es un modelo computacional del sistema inmune biológico que tiene la capacidad de realizar algunas tareas como el reconocimiento de patrones, aprendizaje, adquisición de memoria, generación de diversidad, tolerancia al ruido, generalización, detección distribuida y optimización. Están basados en los principios inmunológicos, que son nuevas técnicas computacionales desarrolladas no sólo para una mejor comprensión del sistema mismo, sino también a los problemas de ingeniería [11].

Existen varios modelos del sistema inmune artificial, debido a que su campo de aplicación es muy amplio y a la gran complejidad del sistema inmune biológico. Los modelos pueden resumirse en cuatro grandes grupos:

1. Selección negativa
2. Células dendríticas
3. Teoría de Red inmune
4. Principio de selección clonal

Para la propuesta se utilizó el principio de selección clonal, que se discutirá en la sección 3 de este trabajo

2.3. Enjambre de partículas

Esta técnica evolutiva se basa en la imitación del comportamiento de las sociedades poco complejas, compuestas por individuos poco sofisticados, ejemplos de esto se encuentra en los bancos de peces, las colonias de hormigas o las bandadas de pájaros, que se comportan

como si fueran un único individuo. Este enfoque general se ha concretado en la optimización por colonias de hormigas y la optimización por enjambre de partículas (ACO y PSO respectivamente, por sus siglas en inglés). [12]

El comportamiento de un enjambre se puede modelar con unas cuantas reglas sencillas. Los enjambres de peces (banco, cardumen) y de pájaros (bandadas, parvadas) se pueden modelar con tales modelos simples. Por ejemplo, aún si las reglas de comportamiento de cada individuo (agente) son simples, el comportamiento del enjambre puede ser complicado. En [13] se utilizó los siguientes tres vectores como reglas simples en sus investigaciones [14]:

- Alejarse del agente más cercano.
- Ir hacia el destino.
- Ir hacia el centro del enjambre.

3. Algoritmos evolutivos propuestos

En esta sección se refiere a los algoritmos que fueron empleados en la constitución de este trabajo.

3.1. Algoritmo del principio de selección clonal

El algoritmo desarrollado para resolver el problema de optimización mencionado en este trabajo, utilizando un sistema inmune artificial, se describe a continuación, así como su diagrama de flujo en la figura 1

1. Se genera la población inicial (linfocitos) de manera aleatoria, para los valores de K_p , K_i , K_d distribuidos a

lo largo del espacio de búsqueda de manera uniforme.

2. Con esta población de controladores se evalúan la función objeto que se considera antígeno
3. Los resultados obtenidos después de evaluar la función objetivo, se ordenarán de mayor a menor, entendiéndose que el valor máximo es el mejor resultado.
4. De éstos valores ordenados se seleccionan n número de resultados para ser clonados (se toman las mejores muestras)
5. Realizar el proceso de mutación con los valores clonados, mediante la ecuación 2

$$x_i^{new} = ((1 - \beta) * x_i) + (\beta * x_{random}) \quad (2)$$

Donde:

x_i^{new} es el nuevo elemento mutado.

β es valor aleatorio entre 0 y 1.

x_i es el valor clonado

x_{random} es un valor aleatorio de toda la población de controladores

6. Estos nuevos valores mutados sustituirán a las peores muestras de la población inicial P_i , generando una nueva población de controladores.
7. La población P_i , se evalúa nuevamente con las funciones objetivos.
8. Si los resultados no son óptimos, se vuelven a realizar todo el proceso desde el paso 3, hasta encontrar las mejores soluciones para el controlador.

3.2. Enjambre de partículas

El algoritmo desarrollado para resolver el problema de optimización mencionado en este trabajo, utilizando un sistema inmune artificial, se describe a continuación, así como su diagrama de flujo en la figura 1

1. Se genera la población inicial de manera aleatoria, para los valores de K_p, K_i, K_d (posición de partículas) distribuidos a lo largo del espacio de búsqueda de manera uniforme.
2. Generar los vectores de velocidad correspondientes a K_p, K_i, K_d
3. Con esta población de controladores se evalúan la función objeto que se considera antígeno
4. Actualizar velocidad mediante la siguiente formula

$$v_i(k+1) = \gamma v_i(k) + c_1 r_1 (pbest - p_i(k)) + c_2 r_2 (gbest(k) - p_i(k)) \quad (3)$$

Donde:

v_i es velocidad de partículas

γ es aceleración, que decae conforme se incrementan las generaciones

c_1 es constante [0,1]

r_1 es constante [0,1]

$pbest$ es mejor local

p_i es posición actual de partícula

c_2 es constante [0,1]

r_2 es constante [0,1]

$gbest(k)$ es el mejor global

k generación

5. Actualizar posición de partícula

$$p_i(k+1) = p_i + v_i(k+1) \quad (4)$$

6. Obtener $pbest$ y $gbest(k)$

7. Verificar criterio de terminación, si no saltar a 3

4. Desarrollo y resultados

En esta sección se muestra el desarrollo del algoritmo inmunológico propuesto con una planta en dominio S. La función de transferencia en dominio S esta expresada en la ecuación 5, que se muestra a continuación:

$$\frac{Y(s)}{X(s)} = \frac{s+1}{s^2+5s+6} \quad (5)$$

La función expresada en (3) se convirtió a dominio z mediante un retenedor de orden 0, con un tiempo de muestreo de 50 milisegundos, generando la expresión mostrada en la ecuación 6

$$G(z) = \frac{Y(z)}{X(z)} = \frac{-0.00796z + 0.00837}{0.95z^2 + 1.96z + z} \quad (6)$$

Cuya respuesta a lazo abierto se muestra a continuación en la figura 2

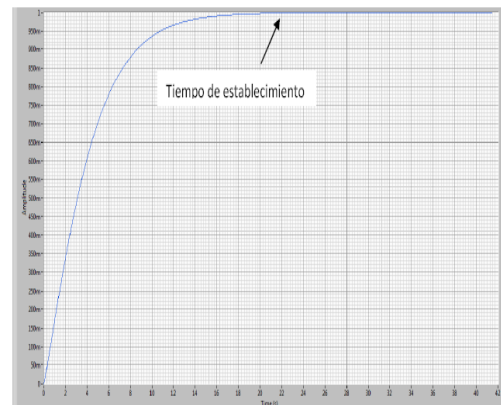


Fig. 2. Respuesta a lazo abierto de la planta propuesta, en ella se puede observar el tiempo de establecimiento, así como el valor final

Al aplicar un ajuste de los parámetros del controlador por medio del criterio de Nichols, se tiene la siguiente respuesta en el tiempo (figura 3)

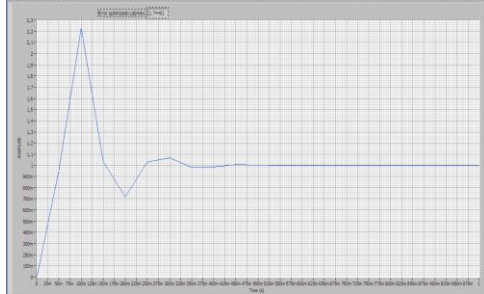


Fig. 3. Respuesta de la planta a lazo cerrado, controlado por medio de un controlador tipo PID ajustado por el criterio de Nichols

Para aplicar el algoritmo inmunológico y el enjambre de partículas propuesto en la sección 3 de este trabajo se utiliza como función de adaptación a la ecuación 7

$$f(z) = \frac{1}{1+e(z)} \quad (7)$$

Donde

$$e(z) = R(z) - C(z) * G(Z)$$

Con los siguientes datos del algoritmo

Población inicial = 50

Porcentaje de clones =80%

Numero de iteraciones=100

Ahora bien, al aplicar el algoritmo inmunológico artificial mostrado en la sección 3 se obtiene lo siguiente en la figura 4

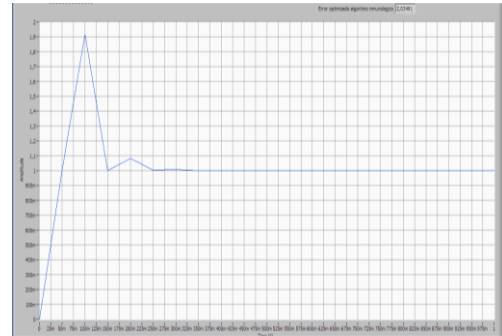


Fig. 4. Respuesta de la planta propuesta al controlador PID ajustado por un algoritmo inmunológico artificial

Ahora bien, si los resultados muestran una mejora en el desempeño del error de controlador, es necesario revisar otras heurísticas, una de ellas que se examina es el enjambre de partículas, proponiendo los siguientes parámetros

Población inicial= 50

No. de iteraciones= 100

Como resultado de lo anterior se tiene la respuesta de un PID a la planta como la que se muestra en la figura 5

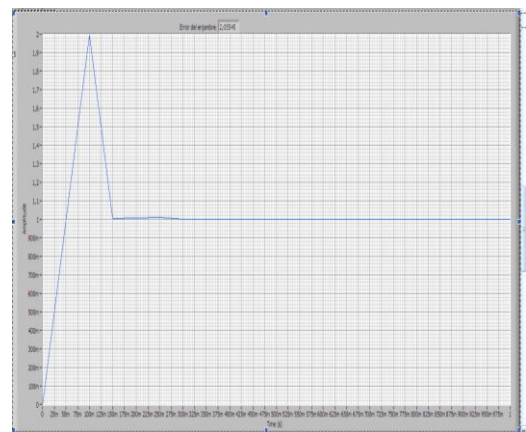


Fig. 5. Respuesta de la planta propuesta al controlador PID por medio de un ajuste de enjambre de partículas

En la tabla 1 se muestra la comparación de los valores que se obtienen al aplicar los 3 algoritmos de cálculo de los parámetros del PID, donde se muestran los atributos de los algoritmos

Algoritmo	Tiempo de establecimiento	Valor de sobre impulso	Tiempo donde ocurre el sobre impulso	Valor del Error
Ajuste de Nichols	575ms	2.21	100ms	2.74 484
Algoritmo Clonal	335ms	1.9	100ms	2.03 941
Enjambre de Partículas	285	2	100ms	2.05 545

5. Conclusiones

En base a los anteriores resultados es posible afirmar que el algoritmo inmunológico clonal es una mejor opción sobre el criterio de Nichols, pero no sobre un enjambre de partículas, que presenta un mejor tiempo de establecimiento con respecto al escalón unitario. Es necesario añadir restricciones a los algoritmos propuestos, lo que se traducirá en una mejora al desempeño del algoritmo. También es necesario explorar las demás variables de un sistema inmunológico artificial, como la teoría del peligro y las redes inmunológicas

References

[1].Tariq Samad, Gary BalasSmith: Software-enabled control: information technology for dynamical systems pp 4 Wiley-IEEE, 2003

[2].Catherin López Sanjurjo: Tecnologías De La Informacion: Conceptos Basicos.May, Ideas propias Editorial S.L., 2006 - 150 páginas.

[3].Pedro Ponce-Cruz, Fernando D. Ramírez-Figueroa: Intelligent Control

Systems with LabVIEW, Springer, 2010 - 216 páginas

[4].Xinjie Yu, Mitsuo Gen Introduction to Evolutionary Algorithms Springer, 2010 - 418 páginas

[5].Christoph Burger Propeller performance analysis and multidisciplinary optimization using a genetic algorithm ProQuest, 2007 - 181 páginas

[6].Xin-She Yang Nature-Inspired Metaheuristic Algorithms Luniver Press, 2011 - 160 páginas

[7].Ying Tan, Yuhui Shi, Yi Chai, Guoyin Wang Advances in Swarm Intelligence Springer, 2011 - 587 páginas

[8].Ogata K. "Ingeniería de Control Moderna" Prentice Hall 3ra.Edición Pág. 669-709 USA 2009.

[9]. Cruz Cortés, Coello Coello."Un Sistema Inmune Artificial para Solucionar Problemas de Optimización Multiobjetivo

[10].Nunes de Castro, Timmis John. "An introduction to Artificial Immune Systems: A New Computational Intelligence Paradigm". Springer-Verlag, 2002.

[11].Cortés Rivera. "Un Sistema Inmune Artificial para resolver el problema del Job Shop Scheduling". México, 2004.

[12].Duarte Muñoz, A Metaheurísticas Librería-Editorial Dykinson, 2008

[13].C. Reynolds. "Flocks, herds and schools: A distributed behavioral model". Computer Graphics, 21:25-34. 1987

[14].Marco Antonio Cruz-Chávez , J. Crispín Zavala-Díaz, Aplicación de la Optimización por Enjambre de Partículas a un Problema Inverso de Conducción de Calor Congreso Internacional de Cómputo en Optimización y Software, Memorias del 7mo Congreso de Cómputo CICOS, 17-20

Noviembre 2009, UAEM, México, ISBN(e)
978-607-00-1970-8, 600 CD's, 421 pp



**García Mejía Juan
Fernando** es maestro en
electrónica, profesor de
tiempo completo del
Centro Universitario

UAE Mex Atlacomulco, su línea de
investigación es la instrumentación virtual,
actualmente estudia el doctorado en
Tecnologías de la información en la UPAEP



García Mejía José Antonio
es ingeniero en electrónica
profesor de asignatura en el
Centro Universitario
UAE Mex Atlacomulco, su

línea de investigación son los sistemas
multimedia y la optimización numérica,
actualmente estudia la maestría en
administración de tecnologías de
información en la UAE Mex



Gutierrez Estrada Citlali
es Doctora en
Computación por la
Universidad de Toulouse,
profesora investigadora

del Instituto Tecnológico de Toluca adscrita
a la división de posgrado, su línea de
investigación es el modelado de sistemas
complejos mediante UML