

OPTIMIZACIÓN Y CONTROL USANDO COLONIAS DE HORMIGAS

Mario A. Muñoz, Jesús A. López, Eduardo F. Caicedo

*Grupo de Investigación en Percepción y Sistemas Inteligentes
Escuela de Ingeniería Eléctrica y Electrónica
Universidad del Valle, Cali, Colombia
e-mail: {andremun,jesuslop,ecaicedo}@univalle.edu.co*

Resumen: En este artículo se presenta la manera como un algoritmo de optimización en espacios continuos basado en colonia de hormigas puede ser aplicado para resolver problemas en el campo del control automático. Los algoritmos basados en colonia de hormigas hacen parte de la denominada Inteligencia de Enjambres o Colectiva. En este trabajo se presentan dos aplicaciones de este algoritmo en el campo del control de procesos: primero se presenta la sintonización de un controlador PID para un reactor de agitación continua, luego el control adaptativo indirecto del nivel de un tanque. Los resultados obtenidos permiten mostrar que la Inteligencia Colectiva, y específicamente, los algoritmos basados en colonia de hormigas constituyen una herramienta prometedora para el desarrollo de controladores inteligentes.

Keywords: Control System Design, Adaptive Control, Model-based Control, Nonlinear Control Systems, PID Control

1. INTRODUCCIÓN

Uno de los inconvenientes que presenta el control convencional es, además de necesitarse previamente un modelo matemático del proceso a controlar, se requiere un exhaustivo trabajo matemático para el diseño de un controlador o compensador. Por otro lado, a pesar de la existencia de algunas metodologías eficientes y de bajo costo que permiten la sintonización de controladores convencionales, generalmente del tipo PID, estas metodologías suelen ser de naturaleza heurística por lo requieren un amplio trabajo de ingeniería. En los últimos años se ha aplicado algunas técnicas de Inteligencia Computacional como las Redes Neuronales Artificiales (RNA), los Sistemas de Inferencia Difusa, los Algoritmos genéticos y la Inteligencia de Enjambres o Colectiva al control automático permitiendo diseñar controladores de una manera más sencilla para el ingeniero de control. Un ejemplo clásico es el control difuso que ha permitido la automatización

de una gran variedad de procesos sin necesidad de disponer de un modelo matemático del sistema bajo control, sino a partir de información empírica, incluso pobremente definida. Otro ejemplo es la manera como se puede sintonizar un controlador tipo PID usando un algoritmo genético logrando, en ocasiones, mejores resultados que con las técnicas convencionales. Hoy en día la Inteligencia de Enjambres surge como una fuente de nuevas herramientas o metodologías cuya aplicación al control automático se vislumbra bastante prometedora (Passino, 2002; Andújar *et al.*, 2006).

Uno de los principales paradigmas dentro de la Inteligencia de Enjambres es el algoritmo de Optimización por Colonia de Hormigas (*Ant Colony Optimization – ACO*). Las hormigas son insectos sociales que viven en colonias y cuyo comportamiento está dirigido a la supervivencia de la colonia en lugar de los individuos. Estos insectos han atraído la atención de los científicos debido al alto nivel de complejidad que las colonias exhiben comparado con la simplicidad de sus indivi-

duos. Un comportamiento particular de las hormigas es su capacidad de encontrar el mejor camino entre sus fuentes de alimento y sus nidos. Para lograr este propósito, las hormigas utilizan una forma de comunicación basada en una sustancia química llamada feromona. Esta sustancia, depositada por la hormiga cuando avanza sobre un camino, representa información que es utilizada por las siguientes hormigas para escoger el camino correcto, que corresponde al que posee mayor concentración de feromona. Gracias a ello, las hormigas encuentran la ubicación de fuentes de alimento al igual que su nido. Se ha demostrado que los rastros de feromona permiten una optimización lenta y distribuida en la cual cada individuo simple provee una pequeña contribución en la búsqueda de la mejor solución. Este es un ejemplo de una propiedad asociada a la colonia que no la posee el individuo (Dorigo *et al.*, 1999).

En este artículo se examinará el uso de algoritmos de optimización por colonia de hormigas para el control automático. El artículo está organizado de la siguiente manera: Inicialmente en la sección 2 se describe el algoritmo de Colonia de Hormigas Adaptativo (*Adaptive Ant Colony Algorithm - AACA*) y se presenta una aplicación a la optimización de funciones en espacios continuos, a continuación en la sección 3 se presenta el uso del algoritmo AACA para la sintonización de un PID para un reactor de agitación continua, posteriormente en la sección 4 se aplicara al control adaptativo indirecto de nivel de un tanque. Finalmente en la sección 5 se deja a consideración algunas conclusiones del trabajo realizado.

2. OPTIMIZACIÓN POR COLONIA DE HORMIGAS EN ESPACIOS CONTINUOS

Los algoritmos de hormigas han encontrado múltiples usos en la solución de problemas de optimización combinatoria (Dorigo *et al.*, 1999; Maniezzo *et al.*, 2004). Sin embargo, la investigación para la solución de problemas en un dominio continuo ha sido poca. Un algoritmo que utiliza conceptos de exploración en espacios discretos para resolver problemas continuos se muestra en (Yan-jun and Tie-jun, 2003), llamado Algoritmo de Colonia de Hormigas Adaptativo (*Adaptive Ant Colony Algorithm - AACA*), el cual utiliza un árbol binario como espacio de búsqueda.

El algoritmo define las siguientes estructuras:

1. Una cadena binaria $x \Leftrightarrow \{b_N, b_{N-1}, \dots, b_1\}$ donde $b_j \in \{0, 1\}$ para $j = 1, 2, \dots, N$, donde b_1 es el bit menos significativo y b_N es el más significativo.
2. Un árbol binario $C = (V, S)$ definido como una serie de nodos

$$V = \left\{ v_s, v_N^0, v_{N-1}^0, \dots, v_1^0, v_N^1, v_{N-1}^1, \dots, v_1^1 \right\} \quad (1)$$

y un grupo de arcos dirigidos

$$C = \left\{ \begin{array}{l} (v_N^0, v_{N-1}^0), \dots, (v_j^0, v_{j-1}^0), \\ (v_j^0, v_{j-1}^1), (v_j^1, v_{j-1}^1), \\ (v_j^1, v_{j-1}^0), \dots, (v_2^1, v_1^1) \end{array} \right\} \quad (2)$$

donde v_s es el único nodo inicial; los nodos v_j^0 y v_j^1 para cada j representan los dos estados del bit b_j respectivamente. Los nodos v_j^0 y v_j^1 para $j = 1, 2, \dots, N$, existen solo arcos que apuntan a los nodos v_{j-1}^0 y v_{j-1}^1 .

Cada hormiga A_i inicia su recorrido por v_s , por el camino creado por los arcos a través de N nodos, formando un camino w_i consistente de la secuencia $\{v_N^{i_N}, v_{N-1}^{i_{N-1}}, \dots, v_1^{i_1}\}$ para $i_N \in \{0, 1\}$ correspondiente a la secuencia binaria x_i , la cual puede ser decodificada por la ecuación 3.

$$\theta_i = \frac{X_i}{2^N - 1} \cdot (x_{max} - x_{min}) + x_{min} \quad (3)$$

donde X_i corresponde al número decimal codificado por la cadena x_i , x_{max} y x_{min} corresponden a los límites del espacio de búsqueda. La asignación de la feromona se realiza por medio de las ecuaciones 4 y 5.

$$\Delta\tau_{i,j}(J_s(t), k) = \frac{1}{1 + e^{\beta \cdot k \cdot J_s(t) \cdot (J_s(t) - (J_{min}(t) + \delta))}} \quad (4)$$

$$\tau(t+1) = \lambda \cdot \tau(t) + (1 - \lambda) \cdot \Delta\tau \quad (5)$$

donde $J_s(t)$ es el costo de la i -ésima hormiga en la iteración t , $J_{min}(t)$ es el costo mínimo durante el experimento hasta la iteración t ; $\beta, \delta, \lambda \geq 0$ corresponden al índice de feromona, índice de costo y el factor de evaporación, y k corresponde al nivel del árbol. Además, se realizó una modificación para incluir los conceptos de exploración y explotación del conocimiento, por medio de una decisión probabilística p_{exp} .

Como ejemplo se muestra la optimización de la función SCB, una función de prueba común para algoritmos de optimización, la cual se muestra en la figura 1 y cuya expresión es la ecuación 6. Esta función presenta seis mínimos locales, de los cuales dos son mínimos globales ubicados en $(-0,0898, 0,7126)$ y $(0,0898, 0,7126)$, con un valor de $J_{min} = -1,0316$.

$$J(\theta_1, \theta_2) = \left(4 - 2,1\theta_1^2 + \frac{1}{3}\theta_2^4 \right) \theta_1^2 + \theta_1\theta_2 + (-4 + 4\theta_2^2) \theta_2^2 \quad (6)$$

Durante un ejercicio de 1000 pruebas, el algoritmo obtuvo un valor mínimo durante el experimento de $J_{min} = -1,0316$, un valor promedio de $\bar{J} = -1,0281$, un error $\epsilon = 0,3295\%$ y una desviación estándar $\sigma = 0,0035$, siendo un algoritmo muy efectivo al encontrar los valores mínimos globales. El desempeño de este algoritmo en una prueba se muestra en la figura 2.

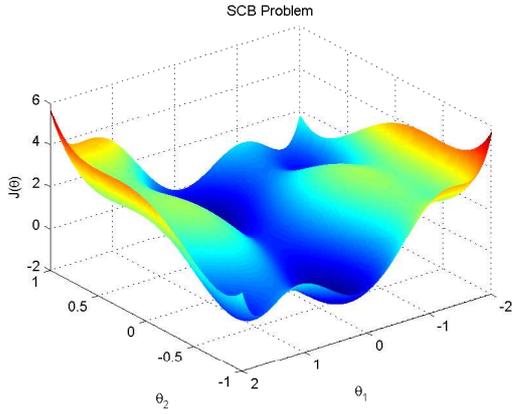


Figura 1. Superficie de la función SCB

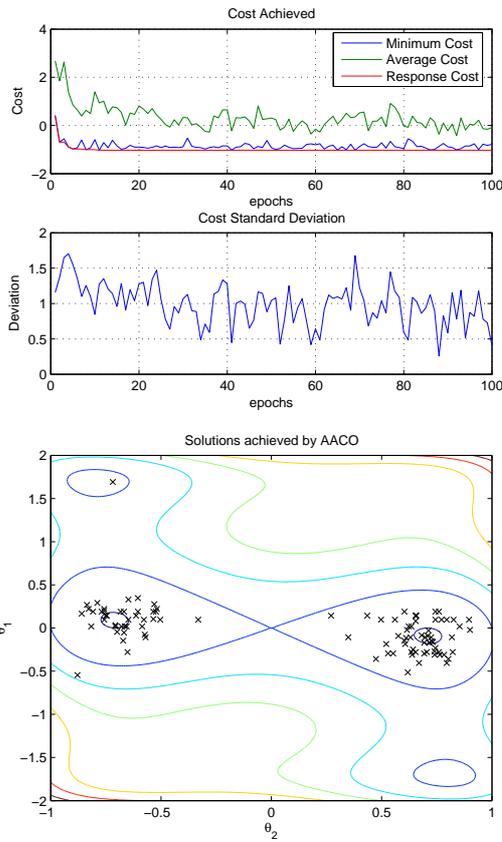


Figura 2. Desempeño del algoritmo AACO durante una prueba: Valores del costo mínimo, mínimo histórico y promedio por iteración, y su desviación estándar (superior); y mejores valores obtenidos en cada iteración (inferior)

3. SINTONIZACIÓN DE UN PID

El control de temperatura en un reactor con agitación continua (Smith and Corripio, 1991) es un problema de estudio clásico en la teoría de control. El reactor se utiliza para calentar un flujo en proceso, de manera que se logre una composición uniforme de los componentes previamente mezclados. El control de temperatura es importante, porque con una alta temperatura se tiende a descomponer el producto, mientras que con una baja temperatura la mezcla resulta incompleta. La

sustancia contenida en el tanque se calienta por medio del vapor que se condensa en un serpentín. Para controlar la temperatura del tanque se realiza una acción de control sobre una válvula de vapor.

Se consideran como condiciones de operación el flujo de alimentación $F(t) = 424,75l/min$ a una temperatura $T_i (^{\circ}C) = 37,7^{\circ}C$. El contenido del reactor se debe mantener a una temperatura $T = 65,5^{\circ}C$. Las posibles perturbaciones son cambios en la tasa de alimentación $F(t)$ y en la temperatura $T_i (^{\circ}C)$. Se digitalizó el sistema presentado en (Smith and Corripio, 1991) por retención de orden cero con un tiempo de muestreo $T_s = 0,7min$, obteniéndose las funciones de transferencia parametrizadas siguientes:

$$G_s(z) = \frac{0,04435z + 0,02726}{z^2 - 1,168z + 0,2281} \quad (7)$$

$$G_f(z) = \frac{0,0596z + 0,03663}{z^2 - 1,168z + 0,2281} \quad (8)$$

$$G_i(z) = \frac{0,0824z - 0,02185}{z^2 - 1,168z + 0,2281} \quad (9)$$

$$G_v(z) = \frac{1,602}{z - 0,0302} \quad (10)$$

Donde la ecuación 7 expresa la relación entre la temperatura del flujo de vapor en el serpentín y la salida; la ecuación 8 corresponde a la relación entre el flujo de sustancia de entrada y la salida; la ecuación 9 provee la relación entre la temperatura del flujo de entrada y la salida; y la ecuación 10 la función de transferencia correspondiente a la válvula de vapor. Para agregar complejidad al modelo se consideran saturaciones en las entradas y salida, y cuantificación de la acción sobre la válvula.

Los controles PID son los más utilizados en la industria, de hecho cerca del 95 % de los lazos de control poseen controladores PID y la mayoría son PI (Herrero *et al.*, 2002). Sintonizar PID por métodos evolutivos no es algo nuevo, se han utilizado en particular los Algoritmos Genéticos en varios trabajos para sintonizar controladores en sistemas no lineales (Herrero *et al.*, 2002; Griffin, 2003). Para este caso, sintonizaremos el controlador PID digital con la estructura mostrada en la ecuación 11. La función de costo de los algoritmos esta definida como la ecuación 12 donde $K_1 = 1/N$ y $K_2 = 1/(N - 1)$, la cual intenta garantizar que se obtenga el menor error de estado estable, rápido seguimiento de la referencia junto a una acción de control suave.

$$U(z) = K_p \cdot E(z) + K_d (1 - z^{-1}) \cdot E(z) + K_i \frac{1}{z - 1} \cdot E(z) \quad (11)$$

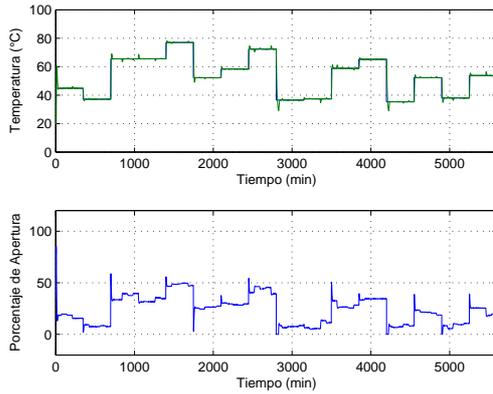


Figura 3. Resultados del controlador PID sintonizado por Colonia de Hormigas

$$J_i = K_1 \cdot \frac{1}{N} \sum_{k=0}^N (r(k) - y(k))^2 + K_2 \cdot \frac{1}{N-1} \sum_{k=1}^N (u(k) - u(k-1))^2 \quad (12)$$

Para la obtención de las constantes, se ejecutó un ejercicio, cuya señal de entrada es un escalón entre $37,7^\circ C$ y $65,5^\circ C$. El algoritmo posee una población de 20 hormigas y se llevo a cabo durante 100 iteraciones en un árbol binario de profundidad 10. Los parámetros del algoritmo fueron: $\tau_{ini} = 0,5$, $\lambda = 0,9$, $\beta = 0,1$, $\delta = 0$ y $p_{exp} = 0,75$. Los valores de las ganancias obtenidas fueron $K_p = 1,2903$, $K_d = 0$ y $K_i = 0,1955$. Para comprobar el desempeño del controlador, se realizó un experimento en el cual se modifica la señal de referencia y se introducen disturbios en el flujo y la temperatura de flujo. El resultado obtenido se muestra en la figura 3.

4. CONTROL ADAPTATIVO INDIRECTO DE NIVEL DE UN TANQUE

Los métodos de optimización como los algoritmos de gradiente y mínimos cuadrados recursivos son utilizados para la estimación de modelos o para modificar los controladores en algunos esquemas de control adaptativo (Passino, 2002). Por lo tanto, la utilización del algoritmo AACA como base para un controlador adaptativo es inmediata.

Se puede considerar un controlador adaptativo indirecto donde se busca aprender el modelo de una planta durante la operación del sistema de control. El aprendizaje se puede visualizar como la búsqueda de buena información del modelo (p.e. información que es realista y útil para alcanzar los objetivos). El “modelo de identificación” utilizado es un modelo parametrizado de la planta en cual el algoritmo AACA realizará la búsqueda dentro del espacio de parámetros de este modelo. Se define el índice de desempeño J como la sumatoria de las últimas $N = 100$ muestras del error

cuadrático entre la salida del modelo y la salida de la planta. El problema a resolver es el control de nivel de un tanque, donde se utiliza el modelo discretizado dado por la ecuación 13.

$$\frac{dh(t)}{dt} = \frac{-\bar{d}\sqrt{2gh(t)}}{A(h(t))} + \frac{\bar{c}}{A(h(t))}u(t) \quad (13)$$

donde $h(t)$ es el nivel del líquido (saturado así no puede ser menor de cero), $u(t)$ es la entrada (también saturada), \bar{c} y \bar{d} son constantes, y $A(h(t)) = \bar{a}h(t) + \bar{b}$, con \bar{a} y \bar{b} también constantes, es el valor desconocido de la área de la sección (creando la necesidad de estimar la dinámica de la planta para su compensación). El modelo parametrizado a identificar es el siguiente:

$$\hat{h}(k) = \hat{\alpha}(k-1)h(k-1) + \hat{\beta}(k-1)u(k) \quad (14)$$

donde $\hat{\alpha}$ y $\hat{\beta}$ son los parámetros a identificar. La ley de control utilizada es

$$u(k) = \frac{1}{\hat{\beta}(k-1)}(r(k) - \hat{\alpha}(k-1) \cdot h(k-1)) \quad (15)$$

Con el algoritmo AACA se busca dentro del espacio de parámetros por valores que permitan minimizar el índice de desempeño J . Lo anterior se hace seleccionando a cada instante el mejor modelo que las hormigas han encontrado y usando el mismo para especificar un controlador. Para el estudio se utilizó un tiempo de muestreo de $T = 0,1$. Los parámetros del algoritmo fueron los siguientes: población de 10 individuos, profundidad del árbol 10 bits, $\lambda = 0,6$, $\beta = 0,1$, $\delta = 0$ y $p_{exp} = 0,61$.

Con este algoritmo se obtuvieron los resultados mostrados en la figura 4, donde se puede observar que toma muy poco tiempo, luego de que las primeras muestras son tomadas, para que el controlador se adapte y obtenga un buen desempeño. El controlador presenta algunos picos al inicio de los pulsos. La estimación del parámetro $\hat{\alpha}$ es buena, a pesar de mantener cierta cantidad de error, sin embargo, el parámetro $\hat{\beta}$ converge al valor mínimo.

5. CONCLUSIONES

En este trabajo se examinaron dos aplicaciones de un algoritmo de colonia de hormigas para el diseño de controladores. Los resultados obtenidos en ambos experimentos, muestran que los algoritmos de hormigas presentan una inteligencia emergente que los hace capaces de obtener buenos resultados frente a algunos problemas relacionados con el control de procesos. La sintonización del PID permitió obtener un desempeño adecuado además de presentar buen rechazo a las perturbaciones y una rápida respuesta al cambio de la referencia, sin incurrir en fuertes esfuerzos de control. El sistema de control adaptativo implementado mostró una rápida convergencia a valores adecuados,

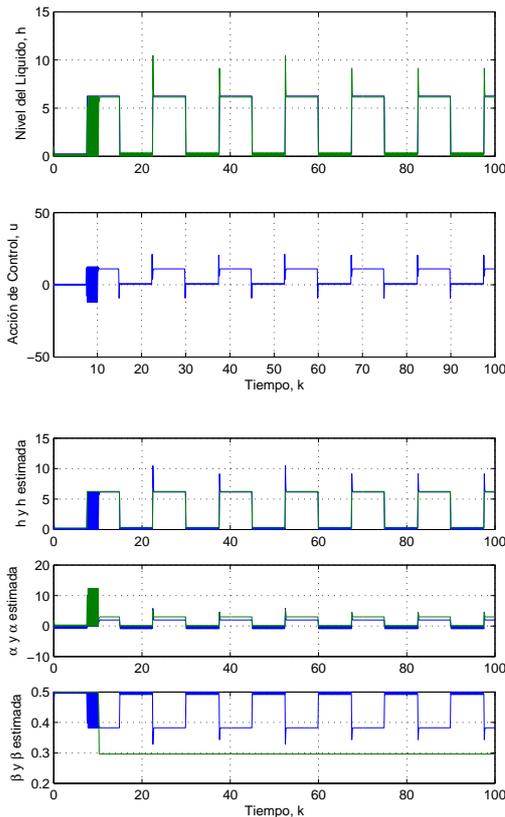


Figura 4. Respuesta en lazo cerrado del controlador adaptativo. Nivel del líquido y acción de control (Superior), y estimaciones de la planta, $\hat{\alpha}$ y $\hat{\beta}$ (Inferior).

permitiendo el seguimiento de la referencia casi al iniciar el periodo de adaptación. El desarrollo dentro de la teoría de control de estos algoritmos requiere el estudio comparativo frente a otras técnicas de sintonización y el análisis de estabilidad de los controladores obtenidos.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por Colciencias y la Universidad del Valle, a través de una beca de investigación otorgada a Mario A. Muñoz en el marco del proyecto “Plataforma de Procesamiento Distribuido, con Acceso Remoto Multiusuario y Emulación de Sistemas Dinámicos, para la Investigación y Educación en Ingeniería”, contrato No. 1106-11-17707.

REFERENCIAS

Andújar, J. M., A. J. Barragán, J. M. Córdoba and I. F. de Viana (2006). Diseño de sistemas de control borroso: Modelado de la planta. *Revista Iberoamericana de Automatica e Informatica Industrial* **3**(1), 75–81.

Dorigo, M., G. Di Caro and L. M. Gambardella (1999). Ant algorithms for discrete optimization. *Artificial Life* **5**(3), 137-172.

Griffin, I. (2003). On-line pid controller tuning using genetic algorithms. Master’s thesis.

Herrero, J. M., X. Blasco, M. Martínez and J. V. Salcedo (2002). Optimal pid tuning with genetic algorithms for non linear process models. In: *IFAC 15th Triennial World Congress*.

Maniezzo, V., L. M. Gambardella and F. de Luigi (2004). *Ant Colony Optimization*. Chap. 1, pp. 101–117. Springer-Verlag Berlin Heidelberg.

Passino, K. M. (2002). Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Systems Magazine* **22**(3), 52–67.

Smith, C. A. and A. B. Corripio (1991). *Control Automático de procesos – Teoría y Practica*. Noriega Editores.

Yan-jun, L. and W. Tie-jun (2003). An adaptative ant colony system algorithm for continuous-space optimization problems. *Journal of Zhejiang University SCIENCE* **4**(1), 40–46.