

## **Sistema competitivo multi-enjambre en la asignación adaptativa de recursos para un sistema multiprocesos**

### **Competitive multi-swarm system in adaptive resource allocation for a multi-process system**

*Juan Marulanda\*, Wilfredo Alfonso, Eduardo Caicedo*

Escuela de Ingeniería Eléctrica y Electrónica. Universidad del Valle. Carrera 100 No. 13-00, Cali, Colombia.

(Recibido el 13 de julio de 2010. Aceptado el 18 de enero de 2013)

#### **Resumen**

En este artículo se presenta una nueva propuesta que realiza asignación adaptativa de recursos para controlar un sistema multiproceso utilizando técnicas bio-inspiradas, basadas específicamente en algoritmos de inteligencia de enjambres (*Swarm Intelligence*, SI). Estos algoritmos solucionan problemas de alta complejidad demostrando características de adaptación y cooperación entre los diversos agentes individuales, a partir de reglas simples. Ello conlleva a observar con expectativa el desempeño sobre el nivel de aplicabilidad que estos algoritmos tienen en el control de sistemas complejos con múltiples entradas y salidas. En este trabajo se propone ejecutar diferentes modelos de enjambres de forma independiente recreando un sistema competitivo desde el punto de vista operacional, ya que estos se basan únicamente en propiedades asignadas a su propio modelo de enjambre. Esta propuesta fue estudiada y analizada utilizando diversas métricas de desempeño. Todos los algoritmos evaluados lograron controlar todos los procesos asignados especialmente el modelo de hormigas que mostró mayor estabilidad.

----- **Palabras clave:** Control Inteligente, Inteligencia de Enjambres, Sistema multiproceso, asignación adaptativa de recursos

#### **Abstract**

This article presents a new proposal that performs adaptive resource allocation to control a multi-process system using bio-inspired techniques, based specifically on swarm intelligence algorithms (SI). These algorithms solve highly complex problems from simple rules to show adaptive and cooperative

---

\* Autor de correspondencia: teléfono: 57 + 2 + 315 67 73, fax: 57 + 2 + 339 23 61 ext. 112, correo electrónico: [juanfemarula@gmail.com](mailto:juanfemarula@gmail.com). (J. Marulanda)

characteristics among various individual agents. This leads to observe with expectation the performance in the applicability that these algorithms handle to control complex systems with multiple inputs and multiple outputs. This paper presents a proposal of different swarm models run independently that recreate a competitive system from an operational standpoint, because each model is based only on its own assigned properties. This proposal was studied and analyzed using various performance metrics. All tested algorithms were able to control all assigned processes and especially the ant model showed more stability.

----- **Keywords:** Intelligent Control, swarm intelligence, multi-processing system, adaptive resource allocation

### Introducción

En el campo de la Inteligencia de Enjambres, el comportamiento de agentes simples ha sido estudiado ampliamente demostrando que son sistemas eficaces para la solución de problemas. Los mecanismos de toma de decisiones, surgen tanto de la inteligencia individual como colectiva que progresivamente cada agente muestra para favorecer a su enjambre [1]. Las estrategias implementadas por el algoritmo tales como redundancia, asociación, colectividad, división de labores, entre otras, determinan la supervivencia de los enjambres modelados [2].

Con técnicas basadas en *Foraging* Bacterial se ha propuesto solucionar el problema de sintonización de controladores [3], como alternativa para el control de sistemas complejos. Posteriormente éste y otros modelos de *foraging* social, como el modelo depredador-presa, fueron usados para resolver sistemas de mayor complejidad como en [4], donde se trató un sistema no-lineal de temperatura con múltiples entradas y salidas; casos similares implementando un conjunto de algoritmos de enjambres, simulando el comportamiento de asignación de tareas en hormigas, *foraging* social de bacterias y búsqueda de néctar de las abejas, fueron presentados para dar solución en una grilla de temperatura con mayor número de entradas y salidas (MIMO) con el fin de regular una superficie uniforme de temperatura, con resultados satisfactorios en [5].

Adicionalmente, el algoritmo de colonia artificial de abejas (ABC, por sus siglas en inglés) ha

sido implementado con éxito en campos como el diseño de filtros digitales, donde es posible optimizar un filtro IIR al lograr mejorar su superficie de error multimodal [6]. De esta forma, es posible observar que el rango operación de los algoritmos basados en Inteligencia de Enjambres no solo se enfoca en problemas relacionados con sistemas de control, lo cual muestra un amplio campo en investigación sobre este tipo de estrategias. Otras aplicaciones que involucran el uso de algoritmos de enjambres se encuentran en [7 - 14], entre otros.

Sin embargo, no se ha hallado evidencia que muestre sistemas con emulación de multi-enjambres dentro de sistemas multiprocesos. El objetivo en sí es proponer una herramienta alternativa que ofrece un enfoque diferente al modelo tradicional para en las estrategias de control. Por lo tanto, este artículo presenta un sistema complejo de procesamiento de datos basado en inteligencia de enjambres para llevar a cabo el control de un sistema multiproceso, basado en la asignación adaptativa de recursos. En esencia, el sistema usa  $m$  controladores bioinspirados para controlar  $n$  procesos. Para este modelo cada enjambre debe enfrentarse a formas de disturbio atípicas y no medibles, tales como los retardos aleatorios generados en una red y la incidencia de otro enjambre en el sistema. Este tipo de eventos permite observar el alto nivel exploración y aprendizaje que estos algoritmos poseen.

Este artículo está organizado de la siguiente forma. Primero, se presenta una breve descripción

de la teoría de *foraging* social, haciendo énfasis en los algoritmos de enjambres implementados como: asignación adaptativa de tareas (ATAA), *foraging* bacterial (BFO) y búsqueda de néctar de abejas (BNSO). Después, se presenta la plataforma multiproceso experimental desde una perspectiva general implementada con sistemas multi-enjambre. Posteriormente, se presenta el diseño experimental sobre un sistema específico de 4 procesos y 4 enjambres; seguido de los resultados obtenidos a través de métricas de desempeño. Finalmente, se presentan algunas conclusiones relevantes.

### Técnicas basadas en *foraging*

La teoría de *foraging* presentada por [3], describe los mecanismos de toma de decisiones que tienen los animales a tareas típicas para asegurar la supervivencia de la especie, tales como la búsqueda de alimento, mecanismos de defensa, tareas reproductivas, entre otras; de esta forma, se busca maximizar la energía (E) por unidad de tiempo (T). Así, el Provecho (P) se describe como:

$$P = \frac{E}{T} \quad (1)$$

Esta función es básica para toda especie pero depende exclusivamente de la cantidad de energía que acumula con su dieta y la facilidad que tiene para obtenerla. La teoría de *foraging* involucra todas las decisiones para realizar el proceso completo: encontrar las zonas, evaluarlas, ingerir el alimento, abandonarlas, buscar nuevas, etc.

La teoría de *foraging* no solo evalúa a cada individuo, también trata la conducta de todo un grupo de individuos; esto se denomina *foraging* social o cooperativo [3]. Las ventajas incluyen:

- La tasa de la ganancia energética por individuo puede ser más alta por cada animal en un grupo, aun si la ganancia no está repartida equitativamente.
- Cooperación para la consecución de alimento. En algunos casos, la tasa neta de ganancia energética de un individuo puede ser muy pequeña. Pero si pertenece a un

grupo, sus probabilidades mejorarán al haber un mayor número de animales buscando nutrientes. Si un animal encuentra algún nutriente, éste puede informar al resto del grupo sobre su ubicación.

- Aumenta la capacidad para cazar una presa de mayor tamaño. El grupo puede reunirse para atacar una presa grande mientras que un único depredador no podría ser capaz de lograrlo.

### Algoritmo de asignación adaptativa de tareas (ATAA)

El algoritmo ATAA simula el comportamiento estigmergético de las hormigas donde los individuos son motivados a resolver tareas específicas de la colonia [15- 17]. Así, la probabilidad  $P_{ij}$  de que un individuo  $i$ , localizado en la zona  $z(i)$ , responda a una demanda  $S_j$  en la zona  $j$  esta dado por la ecuación 2.

$$P_{ij} = \frac{S_j^2}{S_j^2 + \alpha \theta_{i,j}^2 + \beta d_{z(i),j}^2} \quad (2)$$

Donde,  $\theta_{i,j} (\in [\theta_{min}, \theta_{max}])$  es el umbral de respuesta del agente  $i$  a la demanda de la zona  $j$ ,  $d_{z(i),j}$  es la distancia entre la ubicación del agente  $z(i)$  y la zona  $j$  (distancia que puede ser euclidiana o incluir otros factores), y  $\alpha$  y  $\beta$  son 2 coeficientes positivos que modulan las influencias respectivas de  $\theta$  y  $d$ . Cada vez un agente  $i$  se asigna una zona  $j$ , sus umbrales de respuesta son actualizados de la siguiente forma:

$$\theta_{i,j} \leftarrow \theta_{i,j} - \xi_0$$

$$\theta_{i,n(j)} \leftarrow \theta_{i,n(j)} - \xi_1, \quad \forall n(j) \quad (3)$$

$$\theta_{i,k} \leftarrow \theta_{i,k} + \varphi, \quad \forall k \neq j, k \notin \{n(j)\}$$

Donde,  $\{n(j)\}$  es un grupo de zonas que rodean a  $j$ ,  $\xi_0$  y  $\xi_1$  son dos coeficientes de aprendizaje correspondientes a la zona  $j$  y a sus zonas vecinas respectivamente, donde ( $\xi_0 > \xi_1$ ). El coeficiente de olvido  $\varphi$  es aplicado a los umbrales de respuesta asociados con otras zonas.

Cada agente no asignado previamente debe de evaluar su condición actual para atender una zona aleatoria basándose en la ecuación 2, la cual depende del umbral de respuesta ( $\theta_{ij}$ ) y su distancia a la zona determinada ( $d_z(i, j)$ ), valores que es posible considerarlos como las características de entrenamiento en cada agente.

**Algoritmo de optimización de foraging bacterial (BFO)**

El algoritmo BFO propuesto en [3, 18] simula el comportamiento de la bacteria E.Coli para conseguir alimento a través de un proceso quimiotáctico. El objetivo es optimizar una función  $J(\theta)$ , donde la ubicación  $\theta \in \mathbb{R}^n$  (siendo  $\mathbb{R}$  un espacio de  $n$  dimensiones), y del cual el enjambre desconoce una descripción analítica del gradiente  $\nabla J(\theta)$ . La función  $J(\theta)$  posee efectos combinados de sustancias atrayentes y repelentes de otros individuos, así como la calidad en la posición  $\theta^i$  de la  $i$ -ésima bacteria. Para cada agente ubicado en  $\theta^i$  si la función es  $J(\theta) < 0$  el entorno es saludable y rico en nutrientes, si  $J(\theta) = 0$  se considera una zona neutra, y por último si  $J(\theta) > 0$  es una zona nociva. Esta conducta se denomina quimiotaxis [3], y se especializa en buscar las concentraciones más saludables y de altas concentraciones de alimento.

De esta forma el algoritmo evalúa cada paso quimiotáctico del agente e intenta seguir un gradiente positivo de nutrientes. El movimiento completo se describe en la Ecuación 4, donde el índice  $i$  representa al agente evaluado y los índices  $j, k$  y  $l$  representan el cálculo del paso quimiotáctico, la etapa de reproducción y el evento de eliminación respectivamente. La relación  $C(i) \emptyset(j)$  añade la dimensión y la dirección del paso en su desplazamiento. Para una descripción más detallada revisar [3, 19, 20].

$$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i) \emptyset(j) \quad (4)$$

**Algoritmo de optimización de enjambre de abejas (BNSO)**

Este algoritmo fue propuesto por Alfonso [6] basado en el modelo orientado al individuo planteado por de Vries y Biesmeijer [21] y Muñoz [22]. El modelo describe los mecanismos en la búsqueda y recolección de néctar de las abejas a través de la distinción de clases y categorías (tabla 1) como un mecanismo de control y asignación de tareas, con el propósito de llegar a fuentes más provechosas y explotarlas al máximo. Para ello, se utiliza información local suministrada en las danzas, y un equilibrio entre exploración y explotación.

**Tabla 1** Clasificación de agentes para el algoritmo BNSO

		<i>Clases</i>			
		<i>Novato</i>	<i>Explorador</i>	<i>Recluta</i>	<i>Inspector</i>
Categoría	Desempleado	X	X	X	
	Empleado			X	
	Experimentado		X	X	X

En la colmena algunos de los agentes que ha explorado el entorno intentarán atraer la atención de otros miembros a través de las danzas con el propósito de aumentar el nivel de visitas sobre una fuente de alimento. Por lo tanto, las danzas se constituyen como un mecanismo de comunicación directa en la colmena. A partir de

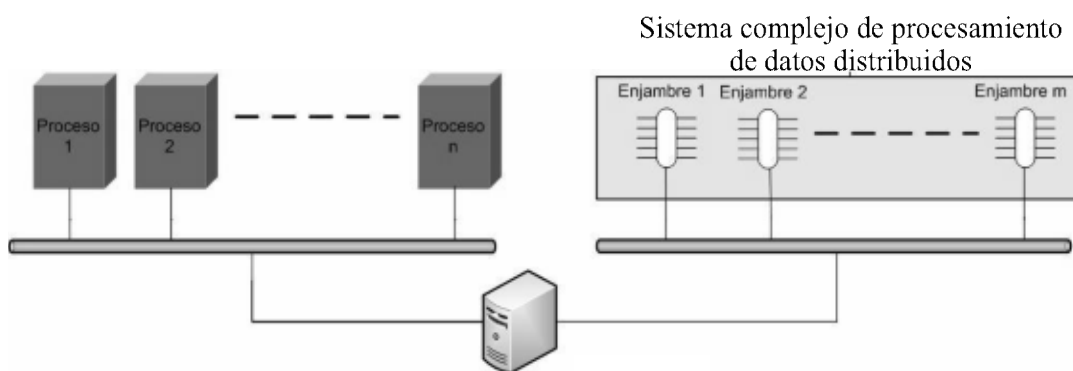
una regla de decisión proporcional, los agentes se vinculan a una de las danzas, donde normalmente reciben esta información con cierto margen de error. El error es considerado como un mecanismo que contribuye a la exploración desde el punto de vista de las fluctuaciones en un sistema auto-organizado [15]. De esta forma, se presenta un

proceso interactivo entre los agentes, los cuales modifican el centro de las fuentes, cuando se encuentran que la fuente informada es de menor provecho que la nueva ubicación.

**Estrategia competitiva multi-enjambre en la asignación adaptativa de recursos para un sistema multi-proceso**

El sistema competitivo multi-enjambre en la asignación adaptativa de tareas en un sistema multi-

procesos (figura 1) está constituido por  $n$  procesos independientes que requieren ser controlados simultáneamente y un sistema complejo de procesamiento de datos distribuidos que posee  $m$  algoritmos basados en modelos de enjambres que actúan de manera concurrente como controladores de los procesos. La comunicación entre estos dos elementos es gestionada por un servidor que se encarga de administrar los datos que fluyen de manera bidireccional entre los  $n$  procesos a ser controlados y los  $m$  enjambres que actúan como controladores.



**Figura 1** Conexión lógica de la red

A partir de esta estructura, se define el conjunto de procesos  $P = \{p_1, p_2 \dots p_i \dots p_n\}$ , donde  $p_i$  es el  $i$ -ésimo proceso a controlar y  $n$  es el número máximo de procesos. Igualmente, se define el conjunto  $E = \{e_1, e_2 \dots e_j \dots e_m\}$  como un conjunto de algoritmos basados en modelos de enjambres, donde  $e_j$  es el  $j$ -ésimo modelo y  $m$  es el número total de algoritmos basados en modelos enjambres en el sistema. El enjambre  $j$ -ésimo está conformado por  $k$  agentes y se define como el conjunto  $e_j = \{a_{j1}, a_{j2} \dots a_{jk}\}$ . El tamaño de  $k$  puede variar para cada enjambre.

Los agentes de cada enjambre poseen el mismo tipo de características y reglas específicas a seguir durante la consecución de los recursos para garantizar su supervivencia. Ya que un agente solo puede comunicarse con los agentes de su propio enjambre, no puede tomar decisiones a partir de la información recolectada de otros enjambres,

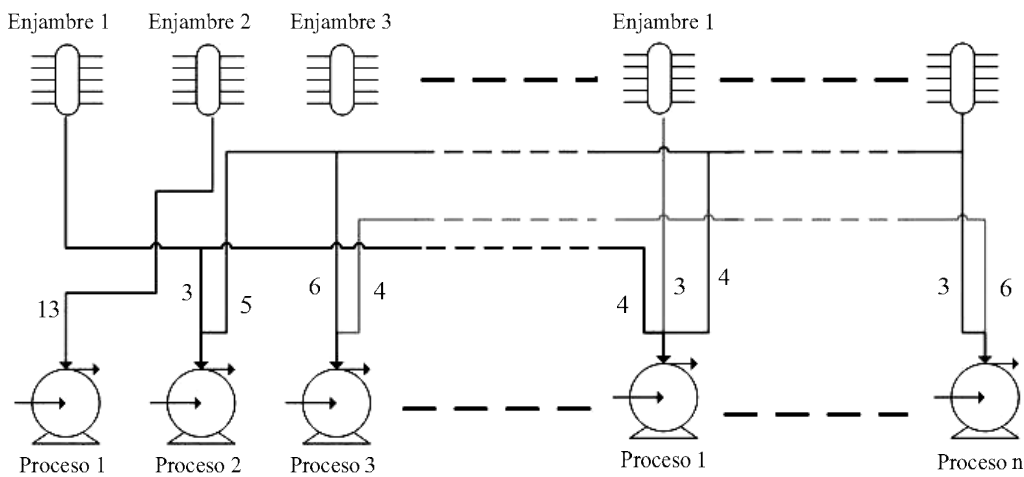
pero tienen la capacidad de observar los cambios de su entorno (información local) generados por los agentes de otros enjambres.

Para la etapa de asignación de tareas, cada agente  $a_{jk}$  toma una porción del recurso actual en el proceso  $p_i$ , la cual va acorde a las propias capacidades que limitan al agente y que desde el punto de vista biológico, suple una parte de las necesidades del enjambre  $e_j$ . De esta forma, la acción que se suma por el aporte de cada agente permite que la cooperación logre cubrir un proceso. A partir de esto, la problemática del sistema y los recursos o alimento que cada agente debe de buscar se localizan en zonas específicas, las cuales identifican a los procesos asignados al sistema. En este caso la cantidad de alimento en una zona determinada representa el error actual de un proceso y entre mayor sea este valor representará una mayor concentración de alimento para los enjambres del sistema.

En la figura 2 se observa un ejemplo de un comportamiento posible de una cantidad  $m$  de enjambres sobre  $n$  número de procesos, los

cuales tienen una cantidad definida de agentes asignados. Tomando como ejemplo el valor  $k = 20$  se tiene la siguiente posible distribución:

$$\begin{aligned}
 p_1 &\ni \{a_{2,1}; a_{2,2}; a_{2,4}; a_{2,6}; a_{2,7}; a_{2,8}; a_{2,10}; a_{2,11}; a_{2,12}; a_{2,13}; a_{2,15}; a_{2,17}; a_{2,19}\} \\
 p_2 &\ni \{a_{1,4}; a_{1,9}; a_{1,17}; a_{m,6}; a_{m,9}; a_{m,10}; a_{m,18}; a_{m,19}\} \\
 p_3 &\ni \{a_{l,1}; a_{l,5}; a_{l,14}; a_{l,16}; a_{m,2}; a_{m,8}; a_{m,11}; a_{m,13}; a_{m,16}; a_{m,20}\} \\
 &\vdots \\
 p_i &\ni \{a_{l,2}; a_{l,6}; a_{l,14}; a_{l,19}; a_{l,7}; a_{l,10}; a_{l,18}; a_{m,1}; a_{m,4}; a_{m,7}; a_{m,14}\} \\
 &\vdots \\
 p_n &\ni \{a_{l,2}; a_{l,4}; a_{l,6}; a_{l,9}; a_{l,12}; a_{l,20}; a_{m,3}; a_{m,5}; a_{m,15}\}
 \end{aligned}
 \tag{5}$$



**Figura 2** Ejemplo de asignación de agentes en múltiples enjambres

La figura 2 muestra que cada enjambre expone comportamientos muy diferentes en relación a los demás debido a su alto nivel de aleatoriedad. Pero al observar con más detalle su comportamiento de forma individual, se encuentran con las siguientes situaciones específicas:

- No todos los agentes en un enjambre pueden terminar asignados en el momento en que el sistema se estabiliza.
- Si el sistema lo requiere todos los agentes de un enjambre pueden quedar asignados.

- Un enjambre puede asignar todos sus agentes a un único proceso.
- Es posible que ninguno de los agentes quede vinculado a un proceso específico.
- Varios enjambres pueden asignar a sus agentes a un único proceso.

Con este modelo se busca obtener un comportamiento con ciertas características de un sistema de control distribuido, explotar el desempeño de los algoritmos bio-inspirados y observar sus capacidades. Cada modelo a

evaluar en cada controlador tiene la capacidad de observar todos los procesos asignados a la red e influir sobre la acción de control que se aplica sobre cada uno de estos. Esta estrategia permite elevar el nivel de complejidad en el sistema y también distribuir la carga computacional sobre cada algoritmo asignado al sistema.

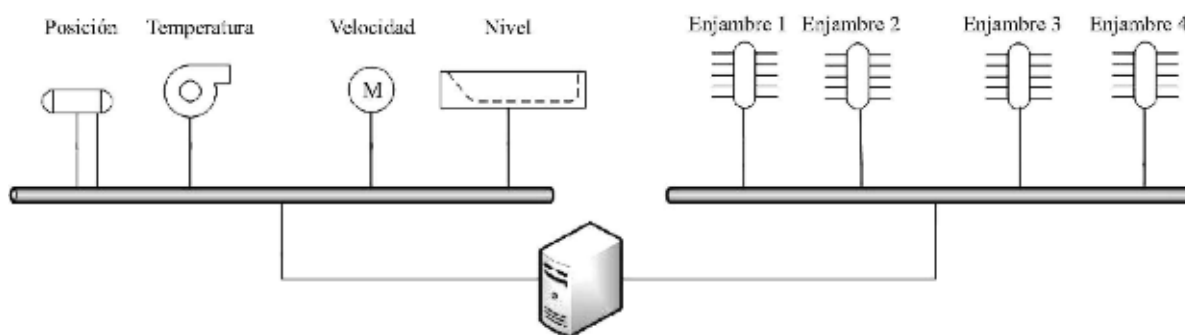
Para que los procesos tengan la posibilidad de ser atendidos por todos los controladores de la red se generó una superficie descriptiva de los procesos. El valor de los ejes X e Y representan la ubicación de los procesos en un espacio normalizando (con el propósito de no dar prioridades en alguno de los procesos), y el eje Z representa los errores a la salida de los sistemas según sus referencias. Una sumatoria de funciones Gaussianas es suficiente para obtener dicha superficie, tal como se presenta en la ecuación 6.

$$S_A = S_{z1} \cdot e^{-\left(\frac{(x_A - x_{z1})^2}{2 \cdot \sigma_{z1}^2} + \frac{(y_A - y_{z1})^2}{2 \cdot \sigma_{z1}^2}\right)} + \dots + S_{zn} \cdot e^{-\left(\frac{(x_A - x_{zn})^2}{2 \cdot \sigma_{zn}^2} + \frac{(y_A - y_{zn})^2}{2 \cdot \sigma_{zn}^2}\right)} \quad (6)$$

## Resultados

### Plataforma de experimentación

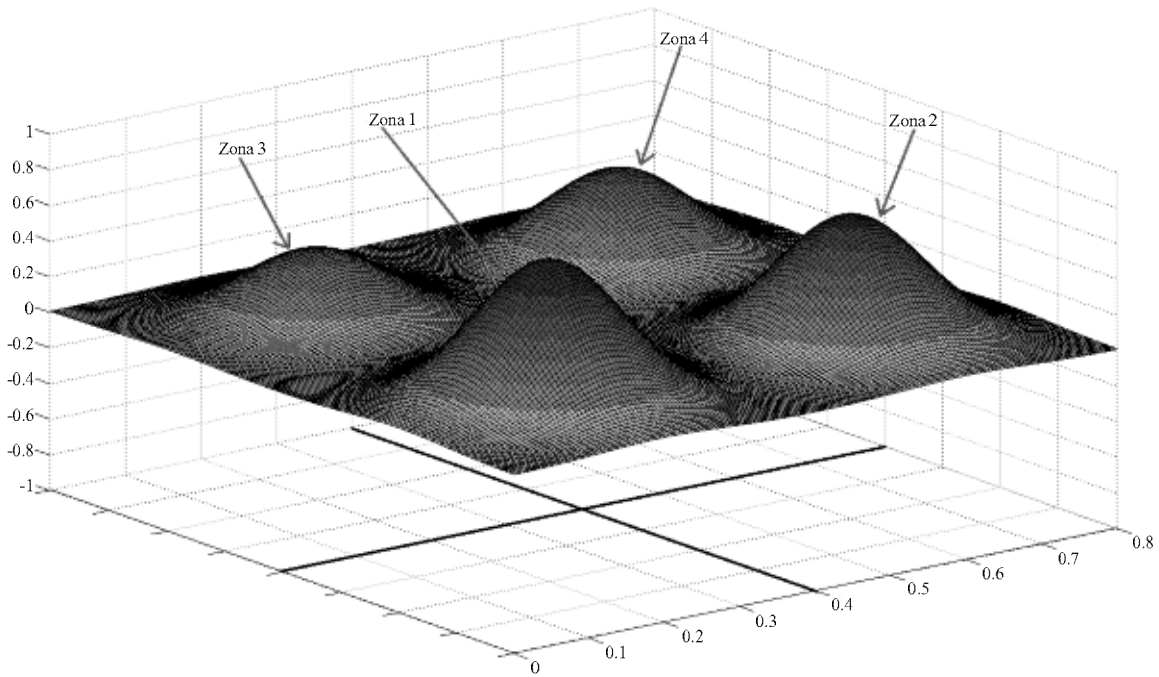
Para evaluar el modelo anteriormente descrito con los algoritmos ATAA, BFO y BNSO asignaron las variables de sistema con los valores  $n = 4$ ,  $m = 4$ ,  $k = 20$ , donde  $n$  es el número de procesos,  $m$  el número de enjambres y  $k$  el número de agentes por enjambre. La figura 3 presenta el esquema de modelo mencionado.



**Figura 3** Modelo implementado para la plataforma

Con las dimensiones de la plataforma a evaluar (figura 4) es posible recrear la superficie que permite activar el comportamiento de los agentes en los 3 algoritmos. De la ecuación 4 se generó el mapa que representa el comportamiento de los

4 procesos como una zona general de alimento (figura 4). Cada proceso se ubicó en un punto específico de la superficie de tal forma que todos tengan la misma probabilidad de ser atendidos por los agentes de cada enjambre.



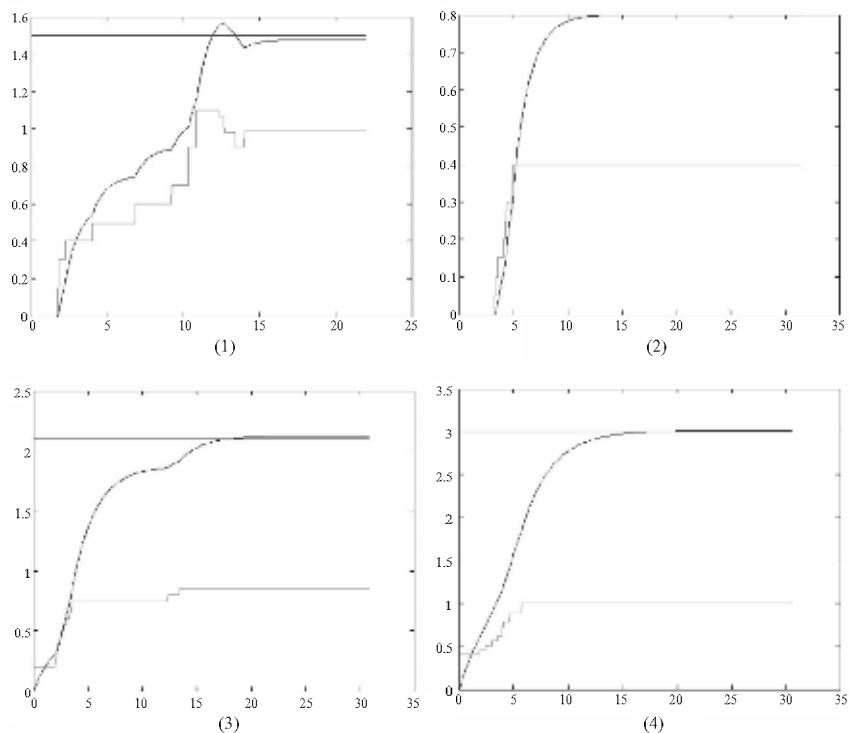
**Figura 4** Ubicación de 4 plantas en el mapa usando función Gaussiana

### Resultados individuales

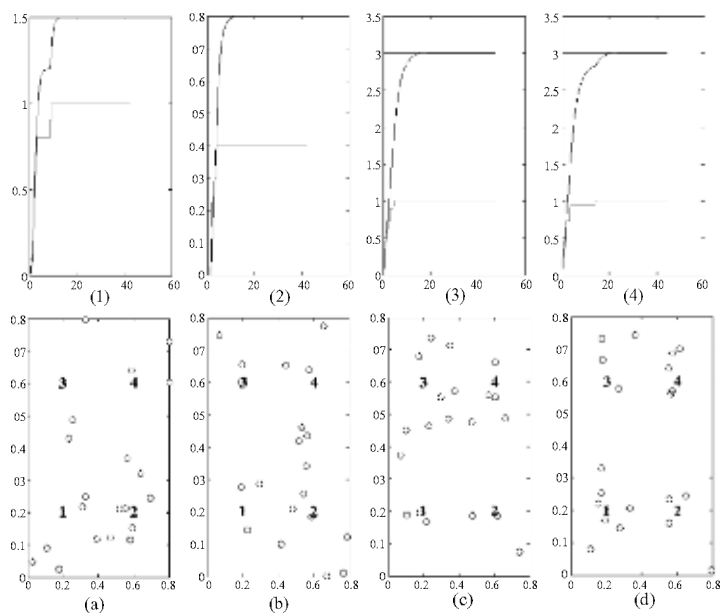
Las pruebas se realizaron para cada una de las técnicas de Inteligencia de Enjambres, donde los 4 computadores que ejecutan los controladores manejan el mismo tipo de algoritmo, sea ATAA, BFO o BNSO. Adicionalmente, en la figuras 5, 6 y 7 se muestra el comportamiento que cada proceso conectado a la red demostró en los

experimentos realizados por cada algoritmo. Las pruebas consisten en el seguimiento de referencia, donde a cada proceso se le asigna un valor diferente a su referencia para evitar similitudes entres sí. Aquí se observa como el sistema lleva la señal de salida de los procesos a su valor de referencia respectivo. La tercera señal que se observa indica la acción de control a la entrada de cada proceso.

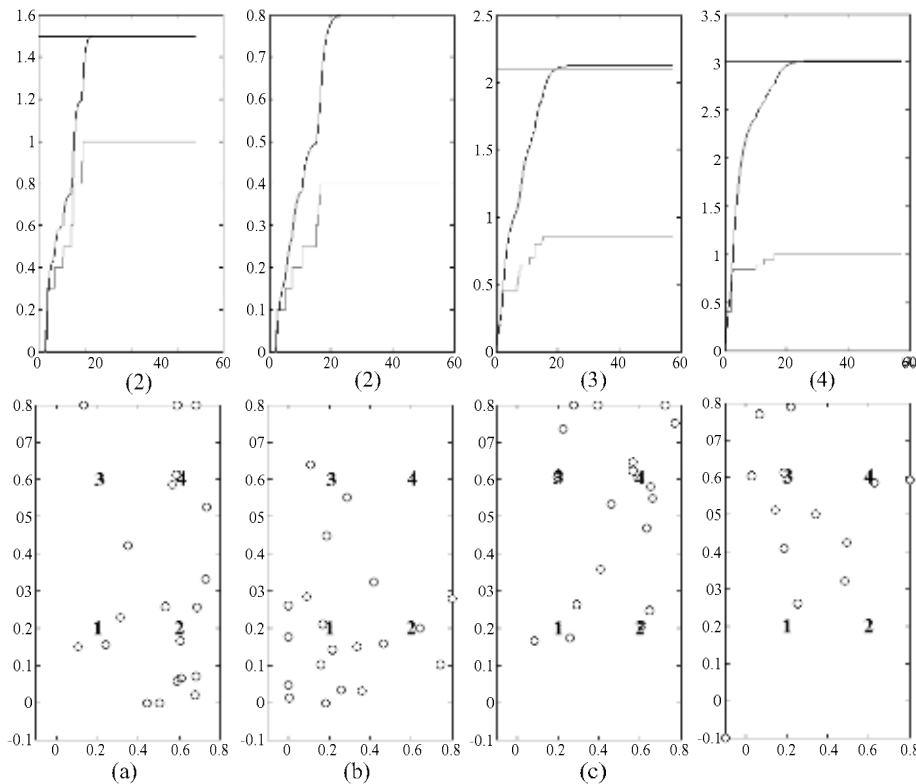




**Figura 5** Control del algoritmo ATAA para 4 procesos diferentes: 1) Proceso 1, 2) Proceso 2, 3) Proceso 3, 4) Proceso 4



**Figura 6:** Comportamiento de control del algoritmo BFO para 4 procesos diferentes: 1) Proceso 1, 2) Proceso 2, 3) Proceso 3, 4) Proceso 4; comportamiento de un grupo de enjambres ante los procesos asignados: a) Enjambre 1, b) Enjambre 2, c) Enjambre 3, d) Enjambre 4



**Figura 7** Comportamiento de control del algoritmo BNSO para 4 procesos diferentes: 1) Proceso 1, 2) Proceso 2, 3) Proceso 3, 4) Proceso 4; comportamiento de un grupo de enjambres ante los procesos asignados: a) Enjambre 1, b) Enjambre 2, c) Enjambre 3, d) Enjambre 4

En las gráficas de la figura 5 se puede observar como el algoritmo ATAA trata de adoptar valores pequeños en la acción de control cuando el error se reduce suficientemente. Esta conducta se entiende como si el agente tratara de obtener una porción de alimento que ha quedado en la zona. También se puede observar que, si llega a ocurrir un sobrepaso, cada algoritmo trata de reaccionar rápidamente a este evento sin generar un cambio brusco en respuesta en el sistema.

En el algoritmo BFO se observó un comportamiento típico en los agentes, donde los agentes se acercan rápidamente a los puntos centrales (figura 6: a, b, c y d) mientras el error se mantuviera alto. Debido a la acción de control ejercida por los agentes sobre la planta, el error empieza a disminuir permitiendo que aumente la influencia de la repulsión entre los agentes. A medida que

este comportamiento continua, el movimiento de los agentes hacia los centros empieza a cambiar gradualmente haciéndose más lento. Cuando el error se reduce al punto de ser un valor cercano a cero, los agentes tienen una mayor influencia del factor de repulsión entre ellos que del error. Por esta razón, el comportamiento que se observa es que los agentes lentamente se van alejando del punto asignado a la planta pero sin distanciarse totalmente.

A diferencia del comportamiento de los agentes observado en el algoritmo BFO, el comportamiento en la figura 7 del algoritmo BNSO muestra a los agentes de cada colmena organizados en grupos más grandes y detallados en la superficie. Debido al error en la comunicación, es posible que un agente quede lo suficientemente alejado como para detectar nuevas zonas, lo que implica

que este algoritmo tiene capacidades adaptativas ante la aparición de nuevos procesos sobre la superficie de error; por otro lado, este comportamiento demuestra el alto nivel de exploración que este algoritmo posee.

### Estudio comparativo

Los comportamientos obtenidos se evaluaron utilizando los siguientes indicadores de desempeño para valorar su calidad y confiabilidad como sistema distribuido:

Promedio del Error al Cuadrado

$$J_1 = \frac{1}{N} \sum_{k=1}^N (r(k) - y(k))^2 \quad (7)$$

Promedio del Error Absoluto

$$J_2 = \frac{1}{N} \sum_{k=1}^N |r(k) - y(k)| \quad (8)$$

Esfuerzo de Control

$$J_3 = \frac{1}{N} \sum_{k=1}^N |u(k)| \quad (9)$$

Suavidad en el Esfuerzo de Control

$$J_4 = \frac{1}{N-1} \sum_{k=2}^N (u(k) - u(k-1))^2 \quad (10)$$

Dispersión del Tiempo de Estabilización

$$J_5 = 3 * \sqrt{\frac{1}{N} \sum_{k=1}^N (t(k) - \bar{t})^2} \quad (11)$$

Donde  $r(k)$  es la señal de referencia,  $y(k)$  la señal de salida y  $t(k)$  es el tiempo en el instante  $k$ .

Con el fin de evaluar adecuadamente el comportamiento de los algoritmos en el sistema, las plantas asignadas a las zonas se organizaron según velocidad de respuesta en lazo abierto siendo la zona 1 la planta más rápida y la zona 4 la planta más lenta. Los índices de desempeño obtenidos en los diferentes experimentos se describen en la tabla 2.

**Tabla 2** Índices de desempeño

		Zona 1	Zona 2	Zona 3	Zona 4
ATAA	J1	0,5044	<b>0,1082</b>	0,2852	1,057
	J2	0,5064	0,1606	0,2507	0,5676
	J3	<b>0,7093</b>	<b>0,342</b>	<b>0,7703</b>	<b>0,9137</b>
	J4	1,96E-04	4,07E-05	4,48E-05	2,13E-04
	J5 (%)	6,35	6,14	5,45	<b>4,26</b>
BFO	J1	0,1511	<b>0,0371</b>	0,264	0,5228
	J2	0,1679	0,0695	0,1932	0,3077
	J3	<b>0,9138</b>	<b>0,3829</b>	<b>0,8231</b>	<b>0,9692</b>
	J4	1,45E-04	5,02E-05	9,36E-05	1,92E-04
	J5 (%)	11,02	10,36	9,92	8,47
BNSO	J1	0,3535	0,1009	0,3881	0,5237
	J2	0,3316	0,1839	0,333	0,353
	J3	<b>0,7948</b>	<b>0,3205</b>	<b>0,7546</b>	<b>0,9423</b>
	J4	9,76E-05	2,78E-05	9,58E-05	2,65E-04
	J5 (%)	13,53	11,81	10,15	9,94

Para las pruebas con el algoritmo ATAA, los algoritmos en la red lograron que las salidas de las plantas alcanzaran las referencias asignadas a cada una de ellas. Los resultados en la tabla 2, tanto en los valores de J5, muestran que los algoritmos tuvieron una dispersión en el tiempo de estabilización menor en las plantas más lentas (Zona 4 con J5=4.26%) que en las rápidas. Aun así, se puede observar que la respuesta a la planta de la zona 4 es la más lenta según el valor obtenido de J1 (1.057).

Para el algoritmo BFO se observó una diferencia muy importante en el índice J5 mostrando tiempos de estabilización en las plantas más dispersos que los obtenidos en el algoritmo ATAA. Esto es debido a la orientación original en la que fue concebido el algoritmo BFO, sin embargo, su alto nivel de exploración es capaz de hallar la solución al problema de control. Comparando

los valores de  $J1$  y  $J3$  del algoritmo ATAA y del algoritmo BFO en la tabla 2, se observó que el índice  $J3$  para BFO fue generalmente más alto que para ATAA, mientras que  $J1$  mostró resultados opuestos. Ya que la estrategia de control de ambas está basada en un control de tipo integral, se puede inferir que el algoritmo BFO tiene en promedio un comportamiento más rápido que el algoritmo ATAA.

A diferencia de los otros 2 algoritmo, BNSO presentó un índice de  $J5$  mayor indicando tiempos de estabilización de las plantas más dispersos. A pesar de este factor, se obtuvieron resultados en los tiempos de estabilización más pequeños que los obtenidos en el algoritmo ATAA y el algoritmo BFO. Los valores del índice  $J3$  fueron muy similares a los obtenidos en el algoritmo ATAA demostrando que el algoritmo BNSO puede igualmente comportarse como un controlador rápido similar al ATAA.

## Conclusiones

Los resultados observados en la fase experimental además de afianzar el alto nivel de adaptabilidad que se conocen de los algoritmos bio-inspirados para resolver problemas, demuestran la ventaja de implementar la conducta de seres simples para cooperar entre sí y funcionar como un solo organismo. Adicionalmente, estos algoritmos permiten el uso de herramientas de recursividad, inteligencia y adaptabilidad a problemas complejos, como el caso planteado de atender un sistema multiproceso.

De igual forma, se logró observar que el tipo de desplazamiento utilizado por los agentes en un algoritmo incrementa la efectividad de los resultados. Basándose en el índice de dispersión del tiempo de estabilización ( $J5$ ), algoritmos con desplazamiento de los agentes de tipo discreto como el algoritmo ATAA mostró resultados menos dispersos que los algoritmos con agentes que implementan desplazamiento de tipo continuo (BFO y BNSO). Por otra parte, este tipo de comportamientos hace susceptible el balance de exploración-explotación de los agentes, como

en los algoritmos BFO y BNSO, donde, según los índices  $J1$  y  $J3$ , se llegaron a obtener mejores respuestas en ciertas pruebas.

El modelo propuesto plantea un punto de vista diferente al común que permite tanto la distribución de controladores en el sistema como una acción de control de tipo competitiva por parte de estos mismos. Adicionalmente, se apreciaron las diferentes cualidades que ofrece este tipo de algoritmos desde una perspectiva muy diferente a la habitual, en donde el controlador centralizado basado en inteligencia artificial es apartado del esquema y la distribución de la carga computacional permitió soluciones alternativas.

Debido al nivel de aleatoriedad que utilizan los algoritmos para tomar una decisión específica en el comportamiento de los agentes del enjambre, los resultados que se pueden llegar a obtener son muy diferentes. No obstante, mediante el ajuste de parámetros de conducta del enjambre y el sondeo del comportamiento del algoritmo a través de la ejecución de pruebas es posible obtener el tipo de resultado deseado variando la relación exploración-explotación que este tipo de algoritmos implementa. Se debe tener en cuenta también que los algoritmos pueden encontrar soluciones en la medida que estos recolecten mayor información sobre las tareas a solucionar, otorgándose una ventaja en el manejo de problemas.

## Agradecimientos

Los autores desean expresar sus agradecimientos a la Universidad del Valle, por facilitar los laboratorios para la ejecución de las pruebas, al Dr. Jesús Alfonso López Sotelo y al Ing. Mario Andrés Muñoz Acosta, por sus ideas y asesoría en el desarrollo de la fase experimental de este trabajo.

## Referencias

1. C. Blum, D. Merkle. *Swarm Intelligence Introduction and Applications*. Ed. Springer-Verlag. Berlin, Heidelberg, Germany. 2008. pp. 3-41.

2. J. Kennedy, R Eberhart. *Swarm Intelligence*. Ed. Academic Press, San Diego, CA, USA. 2001. pp. 288-308.
3. K. Passino. *Biomimicry for Optimization, Control, and Automation*. Ed. Springer-Verlag London Limited. Cap. 1, 5. 2005. pp. 9-92, 764-892.
4. N. Quijano, B. Andrews, K. Passino. "Foraging Theory for Multizone Temperature Control". *IEEE Computational Intelligence Magazine*. Vol. 1. 2006. pp. 18-27.
5. W. Alfonso. "Regulación de Temperatura en la Plataforma UV-PTM01 basada en Agentes Cooperativos para la Asignación Dinámica de Recursos". Tesis de grado. Universidad del Valle. Cali, Colombia. 2007. pp. 144.
6. N. Karaboga. A New "Design Method Based on Artificial Bee Colony algorithm for Digital IIR Filters". *Journal of the Franklin Institute*. Vol. 346. 2009. 328-348.
7. W. Alfonso, M. Muñoz, J. López, E. Caicedo. "Optimización de Funciones Inspirada en el Comportamiento de Búsqueda de Néctar en Abejas". Memorias del Congreso Internacional de Inteligencia Computacional (CIIC2007). Bogotá - Colombia. 2007. pp. 1-9.
8. M. Fleischer. *Foundations of Swarm Intelligence: From Principles to Practice*. Cornell University Library. Ithaca, NY, USA. 2005. pp. 1-13.
9. M. Muñoz, J. López, E. Caicedo. "Optimización por Colonia de Hormigas para la Asignación Dinámica de Recursos en una Plataforma de Experimentación de Temperatura Multizona." *IEEE Latin American Transactions*. Vol. 5. 2007. pp. 82-87.
10. N. Quijano, K. Passino. "Resource Allocation Strategies For Multizone Temperature Control". 2<sup>nd</sup> IFAC Symposium on System, Structure and Control. Oaxaca (México). Diciembre 2004. pp. 1-6.
11. N. Quijano, A. Gil, K. Passino. *Experiments for Distributed and Networked Dynamic Resource Allocation, Scheduling, and Control*. Junio 2004. pp. 1-57. [http://ece.osu.edu/~passino/CSM03039\\_Revised.pdf](http://ece.osu.edu/~passino/CSM03039_Revised.pdf). Fecha de consulta: Diciembre 2009
12. M. Muñoz. "Asignación Dinámica de Recursos con Técnicas Bio-Inspiradas Para Un Sistema de Control de Temperatura MIMO". Tesis de grado. Universidad del Valle, Cali, Colombia. 2005. Fecha de consulta: enero 2010. pp. 151. Impreso.
13. N. Quijano, A. Gil, K. Passino. "Experiments for Dynamic Resource Allocation, Scheduling, and Control". *IEEE Computational Intelligence Magazine*. 2004. Vol. 25. pp. 63-79.
14. N. Quijano. "Experiments and Technologies For Decentralized Temperature Control". Thesis. The Ohio State University. Columbus, Ohio, USA. 2002. Fecha de consulta: Febrero 2010. [http://www2.ece.ohio-state.edu/~passino/tesis\\_final.pdf](http://www2.ece.ohio-state.edu/~passino/tesis_final.pdf).
15. E. Bonabeau, M. Dorigo, G. Theraulaz. *Swarm Intelligence From Natural to Artificial Systems*. Ed. Oxford University Press. New York, NY, USA. 1999. Cap. 2, 3. pp. 25-56, 109-144.
16. M. Dorigo, V. Maniezzo, A. Colomi. "The Ant System: Optimization by a Colony of Cooperating Agents". *IEEE Transactions on Systems, Man, and Cybernetics-Part B*. 1996. Vol. 26. pp. 1-13.
17. M. Dorigo, G. Di Caro, L. Gambardella. *Ant Algorithms for Discrete Optimization*. Artificial Life. Cambridge, MA, USA. 1999. pp. 137-172.
18. K. Passino. "Bacterial Foraging Optimization. International". *Journal of Swarm Intelligence Research*. Vol. 1. 2010. pp. 1-16.
19. M. Muñoz, J. López, E. Caicedo. "Inteligencia de Enjambres: Sociedades Para la Solución de Problemas (Una Revisión)". *Ingeniería e Investigación*. Vol. 28. 2008. pp. 119-130.
20. K. M. Passino. "Biomimicry of Bacterial Foraging for Distributed Optimization and Control". *IEEE Control Systems Magazine*. Vol. 22. 2002. pp. 52-67.
21. H. de Vries, J. Biesmeijer. "Modelling Collective Foraging by Means of Individual Behaviour Rules in Honey-Bees". Vol. 44. 1998. pp. 109-124.
22. M. Muñoz, J. López, E. Caicedo. "An Artificial Bee Hive for Continuous Optimization". *International Journal of Intelligent Systems*. Vol. 1. 2008. pp. 1-7.