

ALGORITMO PARA EL 4AP HACIENDO USO DE LA METAHEURISTICA SISTEMA HORMIGA

ALGORITHM FOR THE 4AP MAKING USE OF THE METAHEURISTIC ANT SYSTEM

MANUEL VICENTE CENTENO ROMERO¹
HEBEL ALEXANDER SALAZAR SALAZAR²

Universidad de Oriente, Venezuela
Unidad Educativa Madre Alberta Giménez, Fe y Alegría Cumaná, Venezuela

RESUMEN

La metaheurística sistema hormiga consiste en la analogía entre el procedimiento que utilizan las hormigas reales para la búsqueda de alimentos, encontrando la ruta más corta, y los problemas de optimización combinatoria para encontrar la mejor solución. Entre estos problemas se encuentra el de asignación multidimensional (*mAP*), el cual es un problema *NP*-difícil para $m > 2$. En la actualidad no se ha desarrollado trabajo alguno sobre el 4AP (*mAP* con $m=4$), tampoco existe publicación sobre la aplicación del sistema hormiga para el *mAP*. En este trabajo se desarrolló un software que hace uso de la metaheurística sistema hormiga para encontrar la mejor solución o aproximada al 4AP, con problemas generados aleatoriamente. Se implementó una metodología híbrida entre la metodología de Investigación de Operaciones descrita por Taha (1991) y la ingeniería de software (1990). El número de asignaciones tomadas en cuenta para verificar qué tan buenas son las soluciones arrojadas por el software, varía desde $n=2$ hasta $n=25$, obteniéndose soluciones exactas para $2 \leq n \leq 6$, ya que al compararse dichas soluciones con las dadas por XPRESS (software de programación lineal que se usa para resolver modelos matemáticos) se observa la igualdad en los resultados. Para $n \geq 7$, XPRESS (la versión utilizada) no ofrece respuesta alguna, pero el software desarrollado arroja buenos resultados en un tiempo computacional razonable, considerando el número de asignaciones que se procesan en cada n .

Palabras claves: asignación, metaheurística, optimización, sistema hormiga.

¹Departamento de Matemáticas, Escuela de Ciencias, Núcleo de Sucre, Universidad de Oriente, Cumaná, estado Sucre, Venezuela.

²Unidad Educativa Madre Alberta Giménez, Fe y Alegría Cumaná, estado Sucre, Venezuela.

¹Avenida Universidad, UDO-Núcleo de Sucre, Departamento de Matemáticas 6101, Tlf: +56 0293-4002153, Fax: +56 0293-4002380, mcenteno@sucre.udo.edu.ve

ABSTRACT

The metaheuristic ant system consists on the analogy among the procedure used by the real ants for searching food, for finding the shortest route; and the problems of combinatoria optimization to find the best solution. Among these problems we can find the multidimensional assignment problem (*mAP*), which is a *NP*-difficult problem for $m > 2$. At the present time, there isn't any work that has been developed, on this topic about the *4AP* (*mAP* with $m=4$), neither publication exists on the application of the ant system for the *mAP*. In this work a software was developed using the metaheurística ant system, in order to find the best solution or an approximate solution for the *4AP*, with aleatorily generated problems. A hybrid methodology was implemented among methodology of Investigation of Operations described by Taha (1991) and the software engineering by Pressman (1990). The number of assignments taken into account to verify how good the solutions given by the software are; varies from $n=2$ until $n=25$, obtaining exact solutions for $2 \leq n \leq 6$, since when they are compared with those given by XPRESS (software of lineal programming that is used to solve mathematical models), the equality is observed in the results. For $n \geq 7$, XPRESS (the used version) doesn't offer any answer, but the developed software gives good results at one reasonable computational time, considering the number of assignments that are processed in each n .

Key words: assignment, metaheuristic, optimization, ant system.

INTRODUCCIÓN

En años recientes, muchos investigadores han centrado su trabajo en el problema de asignación multidimensional (*mAP*), cuyo principio general es organizar las dimensiones adicionales, posiblemente en calidad natural, como por ejemplo el tiempo y el espacio; en la elaboración de asignaciones óptimas, uno de estos casos es el problema de asignación *3-dimensional* (*3AP*), que plantea problemas como asignar el lanzamiento de un satélite X desde la tierra, en dirección a la órbita Y , en diferentes niveles atmosféricos Z . El estudio de este tipo de problemas ha permitido expandir el *3AP* en más de tres dimensiones; por ejemplo, en el caso anterior se podría incluir una dimensión adicional si se consideran varias horas para el posible lanzamiento; dando origen al problema de asignación *4-dimensional* (*4AP*) (Pierskalla, 1967).

En la actualidad existen problemas reales que pueden formularse como un problema de asignación *3-dimensional*, pero estos son difíciles de resolver porque consumen demasiado tiempo en cálculo, por ser problemas más complejos, motivo por el cual se consideran problemas de la clase *NP-difícil* (González y Centeno, 2001).

En los últimos años se han presentado novedades sobre el *3AP*, destacándose los estudios de Balas y Saltzman que propusieron un algoritmo para resolver el siguiente problema: dados tres conjuntos disjuntos, I , J y K y un peso c_{ijk} asociado a cada tripleta ordenada $(i,j,k) \in I \times J \times K$, hallar una colección de peso mínimo de n tripletas disjuntas. A este problema se le denominó *3AP-axial*, para diferenciarlo de otro problema denominado *3AP-planar*, siendo éstos *NP-difícil* (Balas y Saltzman, 1991). Otras investigaciones sobre el problema de asignación multidimensional (*mAP*) son las realizadas por Pierskalla (1967), quien desarrolló un algoritmo que se basa en la técnica de árbol de búsqueda en la variedad de ramificar y acotar. También en la Universidad de Oriente Núcleo de Sucre, Cumaná, se han realizado algunas investigaciones en esta área, tales como las desarrolladas por: González y Centeno (2001) y Centeno y Urbina (2005).

El *mAP* que tratamos en este trabajo es el *mAP* para $m=4$ (*4AP*). Supóngase que se tienen n individuos, n trabajos, n máquinas y n insumos. c_{ijkl} es el costo que se incurre al asignar al individuo i en el trabajo j en la máquina k con el insumo l . Se desea encontrar el costo mínimo de asignar los individuos en los trabajos con máquinas establecidas, utilizando diferentes insumos. Sea x_{ijkl} la variable de decisión para este problema; entonces, cada solución básica factible $x_{ijkl} = 1$ significa que el i -ésimo individuo en el j -ésimo trabajo con la k -ésima máquina utilizando el l -ésimo insumo es asignado, y $x_{ijkl} = 0$ indica que el i -ésimo individuo en el j -ésimo trabajo con la k -ésima máquina utilizando el l -ésimo insumo no es asignado. Es de hacer notar que una vez que los i -ésimo, j -ésimo, k -ésimo, l -ésimo elementos son asignados, no pueden ser utilizados en otras asignaciones.

Un modelo matemático para este problema es el siguiente:

$$\text{Minimizar } \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \sum_{l=1}^n c_{ijkl} x_{ijkl}$$

sujeto a:

$$\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n x_{ijkl} = 1, \quad l = 1, \dots, n. \quad (1)$$

$$\sum_{i=1}^n \sum_{j=1}^n \sum_{l=1}^n x_{ijkl} = 1, \quad k = 1, \dots, n.$$

$$\sum_{i=1}^n \sum_{k=1}^n \sum_{l=1}^n x_{ijkl} = 1, \quad j = 1, \dots, n.$$

$$\sum_{j=1}^n \sum_{k=1}^n \sum_{l=1}^n x_{ijkl} = 1, \quad i = 1, \dots, n.$$

$$x_{ijkl} \in \{0, 1\}, \quad i, j, k, l = 1, \dots, n.$$

Se denotará por *4APA* el problema de asignación *4-dimensional* dado en (1).

El método de solución para este problema es meramente una forma de enumeración implícita de todos los puntos factibles básicos, el cual es utilizado por los métodos exactos para obtener la solución (Burkard y Rudolf, 1993).

Un aumento considerable del valor de *n*, implicaría un incremento en el tamaño de la matriz *A* (matriz de coeficientes del conjunto de restricciones), trayendo como consecuencia la utilización de mayor espacio en memoria, y además, mayor tiempo en la realización de las operaciones correspondientes.

Es por lo antes expuesto que se han diseñado métodos para la resolución de este tipo de problemas, los cuales aunque no obtengan la solución óptima producen una “buena” aproximación a esta. Entre estos métodos están las metaheurísticas, las cuales son una clase de métodos de aproximación que se diseñan para atacar problemas difíciles, proporcionando marcos generales que permiten crear nuevos híbridos, combinando diferentes conceptos derivados de las heurísticas clásicas, la inteligencia artificial, la evolución biológica, las redes neuronales, la mecánica estadística y el psicoanálisis freudiano. Entre estas metaheurísticas se encuentra el sistema hormiga.

Se han propuesto muchos algoritmos basados en sistema hormiga para solucionar diversos tipos de problemas de optimización combinatoria; algunos de estos problemas son el del agente viajero, asignación cuadrática, ruta de vehículos, coloreado de un grafo, entre otros, pero hasta ahora no se ha propuesto ningún algoritmo basado en sistema hormiga para resolver el *mAP*.

El sistema hormiga (ant system) se inspira en cómo las hormigas caminan de las fuentes de comida al nido y viceversa, depositando en el camino una sustancia llamada feromona. Las hormigas pueden oler la feromona depositada por hormigas anteriores, y entre varios caminos escogen aquel cuya intensidad de feromona sea más alta, lo que sirve de guía al resto de las hormigas para encontrar el camino más corto. En los problemas combinatorios se pretende generar soluciones aleatorias, a partir de una dada, mediante la composición de posibles soluciones, marcando con un mayor valor las decisiones que llevaron a una mejor solución. Obviamente, un elemento relevante en la calidad de la solución final es la definición de la feromona y del movimiento de las hormigas.

Las hormigas son capaces de encontrar la trayectoria más corta de una fuente de alimento al nido (sin usar señales visuales). También, son capaces de adaptarse a los cambios en el ambiente; por ejemplo, pueden encontrar una nueva trayectoria, más corta, una vez que el viejo camino sea no factible debido a la presencia de un obstáculo.

El medio usado por las hormigas para formar y mantener la línea de camino es por la segregación de cierta cantidad de feromona mientras caminan, y probabilísticamente cada hormiga prefiere seguir una dirección rica en feromona que una más pobre. Este comportamiento elemental de las hormigas se puede utilizar para explicar cómo pueden encontrar la trayectoria más corta, después que un obstáculo inesperado haya interrumpido la trayectoria inicial.

Las hormigas, una vez delante del obstáculo, no puedan seguir el rastro de feromona y, por lo tanto, tienen que elegir entre cruzar hacia la izquierda o hacia la derecha. En esta situación se espera que la mitad de las hormigas elija cruzar hacia la izquierda, y la otra mitad hacia la derecha de ambos lados del obstáculo.

Las hormigas que eligen la trayectoria más corta alrededor del obstáculo, reconstituyen más rápidamente el rastro interrumpido de feromona, comparado con el rastro de aquellos que eligen la trayectoria más larga. Por lo tanto, la trayectoria más corta recibirá una cantidad más alta de feromona por unidad de tiempo, y ésta alternadamente hará que un número más alto de hormigas elija la trayectoria más corta. Debido a este proceso (autocatalítico) positivo de la regeneración, muy pronto todas las hormigas elegirán la trayectoria más corta.

El algoritmo que se define es un modelo derivado del estudio de colonias de hormiga aplicado a problemas de optimización combinatoria, llamado Sistema Hormiga (AS). La idea básica del algoritmo sistema hormiga es generar una gran cantidad de agentes artificiales simples para que pueden construir buenas soluciones a los problemas de optimización combinatoria. Este sistema tendrá algunas diferencias comparado con el real (natural), donde las hormigas artificiales tienen memoria, no son completamente ciegas y viven en un ambiente donde el tiempo es discreto.

MATERIALES Y MÉTODOS

Las características del hardware de la computadora para realizar las corridas del algoritmo desarrollado son: procesador Pentium IV 1.6 GHz, 256 Mb RAM.

Se utilizó el *XPRESS*, el cual es un software de programación lineal para resolver modelos matemáticos, para resolver el *4APA* para $n=2,3,4,6$; ya que fue lo que la versión encontrada del programa permitió.

Para el desarrollo de la implementación de esta investigación se utilizó una metodología híbrida entre la metodología de investigación de operaciones descrita por Taha (1994) y ciclo de vida de desarrollo de software (Pressman,1990), la cual está constituida por las etapas que se describen a continuación.

Definición del problema: aquí se realizó una descripción del problema en estudio, definiéndose las variables de decisión y sus limitaciones, así como los objetivos.

Construcción del modelo: en esta fase se representó el problema en expresiones matemáticas, definiéndose la función objetivo, restricciones y tipo de variables de decisión que serán empleados.

Fase de diseño: es realmente un proceso multipaso que se enfoca sobre cuatro atributos distintos del programa para resolver el problema: la estructura de datos, caracterización de la interfaz, el detalle procedimental y funcional. Todo este diseño se realizó tomando en cuenta un esquema modular que satisfaga los requerimientos del software de la implementación antes mencionada.

Fase de codificación: en esta fase se escogió el lenguaje de programación C++. Este facilita la programación modular gracias a la inclusión de las estructuras de Clases. Además, este lenguaje incluye funciones que facilitan la manipulación de los datos, lo que permite una reducción importante del tiempo de ejecución. Luego se codificó el algoritmo planteado en el lenguaje seleccionado.

Implementación del modelo: la manipulación de los parámetros a utilizar permitió obtener las soluciones del modelo.

Validación del modelo: se comprobó la confiabilidad del funcionamiento del modelo a través de los resultados obtenidos en la fase anterior; para ello se utilizó la técnica de la media muestral para cada N , comparando los resultados obtenidos por el algoritmo desarrollado con los arrojados por el algoritmo exacto codificado en *XPRESS*.

RESULTADOS

Descripción general del proceso sistema hormiga

En primer lugar se inicializa el rastro de cada una de las acciones posibles. A continuación, cada una de las hormigas construye una solución escogiendo de una lista de nodos los posibles. La elección del nodo se realiza tomando en consideración la información suministrada por la metaheurística y el rastro dejado por las hormigas en las iteraciones previas. El nodo escogido se almacena en una lista tabú para guardar los nodos visitados en tiempo t y prohíbe a la hormiga visitarlos de nuevo antes de la n -ésima iteración. Estos dos pasos se reiteran hasta que la hormiga obtiene una solución factible del problema. Para que esta solución sea candidata a ser la mejor, va a depender de la actualización del rastro de feromona de la hormiga asociada. Para este proceso existen tres variantes: hormiga ciclo: (en este caso, las hormigas, una vez finalizado su recorrido, segrega una cantidad de feromona que es inversamente proporcional a la longitud del recorrido). Hormiga densidad: (en esta variante, cada hormiga segrega una cantidad fija de feromona durante cada recorrido por los nodos). Hormiga cantidad: (cada hormiga segrega feromona durante cada nodo visitado, pero segrega una cantidad que es inversamente proporcional a la distancia entre los nodos).

Cuando todas las hormigas concluyen esta fase constructiva, los incrementos parciales que se han obtenido permiten actualizar el rastro de feromona. Este proceso se repite hasta satisfacer un criterio de parada. La mejor solución, obtenida durante la ejecución del algoritmo, es la solución propuesta por el procedimiento.

En este trabajo se desarrolló un algoritmo de asignación *4-dimensional* y un algoritmo que hace uso de la metaheurística sistema hormiga (AS), adaptado al algoritmo anterior, denominado 4APAS, que se describe a continuación.

Se realiza la fase de inicialización; ésta consiste en una serie de pasos, entre los cuales se tiene: establecer el número n ($2 \leq n \leq 25$) de asignaciones del problema; asignar un valor constante f a la intensidad de feromona (número entero cualquiera) de cada asignación i, j, k, l , en el tiempo cero; es decir $F_{i,j,k,l}(0) = f$. Se toma el incremento total de feromona en i, j, k, l igual a cero, $DF_{i,j,k,l} = 0$. Luego se crean aleatoriamente los costos asociados en las 4-uplas ($c_{i,j,k,l}$), mediante un inicializador semilla. Se procede a encontrar la solución inicial (utilizando un algoritmo *greedy*). Continuando, se ubica aleatoriamente una hormiga en nodos diferentes, representando los nodos de arranque, describiéndose en las matrices: $M_i[h, r]$, $M_j[h, r]$, $M_k[h, r]$, $M_l[h, r]$. Cada uno de los elementos de estas matrices representa una componente de cada 4-upla solución para la hormiga h en la iteración r . Este algoritmo trabaja con veinte (20) hormigas, ya que con más de veinte (20) se genera un tiempo computacional muy elevado, permitiendo mayor exploración de los nodos que pueden formar parte de la solución del problema; y un número de quince (15) ciclos que significan quince (15) intentos que van a tener las veinte (20) hormigas para encontrar una buena solución.

A continuación se asignan valores a los siguientes parámetros a los siguientes parámetros: α , número real que indica la importancia relativa del rastro de feromona ($0 < \alpha \leq 1$); β , es un parámetro real que indica la importancia relativa de la visibilidad ($0 < \beta \leq 1$); Q , valor constante que indica el rastro de feromona que recibe cada asignación por ser visitada (número entero positivo cualquiera) y ρ , número real que representa la cantidad de feromona que desaparece de una asignación por efecto de la evaporación ($0 < \rho \leq 1$); .

La siguiente fase es la de construcción, y expresa que mientras queden nodos por visitar, cada hormiga elige aleatoriamente el próximo nodo. Para un i, j, k, l específico se verifica en cada índice, y si ha sido seleccionado se descarta dicha asignación, escogiéndose una nueva aleatoriamente; si no, se escoge el próximo nodo a visitar. Esto se hace según la siguiente función de probabilidad:

$$P_{i,j,k,l}^h(t) = \begin{cases} \frac{[F_{i,j,k,l}(t)] \cdot [S_{i,j,k,l}]}{\sum [F_{i,d,w,z}(t)]^\alpha \cdot [S_{i,d,w,z}]^\beta}, & \text{si } d \in SJ(i)_h, w \in SK(i)_h, z \in SL(i)_h \quad \forall i = 1, \dots, n \\ 0, & \text{en otro caso.} \end{cases}$$

donde η representa el nivel de intensidad de cada 4-upla, determinando que mientras más pequeño sea el costo de una 4-upla más deseable es ésta y $SJ(i)_h$, $SK(i)_h$, $SL(i)_h$ son los conjuntos de todas las asignaciones J , K y L , respectivamente, tomadas por la h -ésima hormiga partiendo de i .

Para la selección de una asignación se escoge un número aleatorio, y si éste es igual o aproximado a una determinada probabilidad acumulada, se escoge la asignación correspondiente a dicha probabilidad, guardándose en una lista para cada hormiga.

Una vez que no existe más nodo a visitar, entonces cada hormiga ha completado un camino; y cuando las veinte (20) hormigas han construido su camino, se ha completado un

ciclo. A continuación comienza el cálculo del incremento del rastro de feromona, obteniéndose la longitud del camino construido por cada hormiga; luego se calcula el incremento de feromona para la asignación (i,j,k,l) , de la h -ésima hormiga, aplicando el método hormiga-ciclo, que ha mostrado buenos resultados en trabajos anteriores (*The ant system*, 1996), el cual consiste en que las hormigas segregan la feromona una vez construida una solución; es decir, una vez finalizado un recorrido, segrega una cantidad de feromona que es inversamente proporcional a la longitud del recorrido, a través de:

$$\ddot{A}F_{i,j,k,l}^h = \begin{cases} \frac{Q}{Long_h}, & \text{si la asignación } (i,j,k,l) \text{ pertenece al camino construido por la } h\text{-ésima} \\ & \text{hormiga} \\ 0, & \text{en otro caso} \end{cases}$$

A continuación se calcula el incremento total de feromona para la asignación (i,j,k,l) , debido a la aportación de todas las hormigas, según la expresión:

$$\ddot{A}F_{i,j,k,l} = \sum_{h=1}^N \ddot{A}F_{i,j,k,l}^h, \text{ la sumatoria se refiere a la suma de los incrementos parciales}$$

debido a cada hormiga, para una asignación (i,j,k,l) determinada.

La siguiente fase es la de actualización de feromona, donde se obtiene el nuevo rastro de feromona para las asignaciones (i,j,k,l) ; esto se realiza mediante la siguiente fórmula:

$F_{i,j,k,l}(t+n) = \tilde{\alpha} * F_{i,j,k,l}(t) + \ddot{A}F_{i,j,k,l}$, es decir, la feromona actual $(F_{i,j,k,l}(t+n))$, de la cual la asignación (i,j,k,l) ha experimentado un cambio con respecto a la anterior feromona en función de una constante ρ , dada por el usuario $(\tilde{\alpha} * F_{i,j,k,l}(t))$, más el aporte de las hormigas que han escogido esta asignación $(\ddot{A}F_{i,j,k,l})$. La idea de utilizar el parámetro ρ radica en hacer que las hormigas olviden asignaciones (i,j,k,l) que pudieron ser malas; así, si una asignación ya no es escogida por las hormigas, la cantidad de feromona irá decreciendo, de manera que cada vez tendrá menor probabilidad de ser elegido. Este proceso es llamado conducta de estancamiento.

La última fase es verificar el criterio de parada, el cual consiste en parar el algoritmo cuando se alcance el número máximo de ciclos y no exista conducta de estancamiento. Es de hacer notar, que el número máximo de ciclos es igual a catorce (14), ya que para valores mayores el tiempo computacional aumenta muy rápidamente. Si no se cumple el criterio de parada, entonces se actualiza el camino más corto encontrado por las veinte (20) hormigas, guardándose en los vectores $A_j[r]$, $A_k[r]$, $A_l[r]$, $A_i[r]$, los cuales representan el valor de asignación i, j, k, l , respectivamente, que forma parte de la solución del problema encontrada en la r -ésima asignación, donde r es el número de iteraciones o número de ciclos, volviendo luego a la fase de construcción. Si se cumple el criterio de parada se imprime la mejor solución encontrada y el tiempo empleado.

A continuación se muestra, en pseudocódigo, el algoritmo descrito:

```

Inicio
Leer  $n$ ;
Leer semilla;
Leer el número máximo de iteraciones;
Inicializar el rastro de feromona;
Generar matriz de costos;
Escoger solución inicial factible;
Repetir
    Guardar el nodo de arranque de cada hormiga;
    Para cada hormiga, hacer el recorrido de los nodos;
    Actualizar el rastro de feromona;
    Calcular y guardar el camino mas corto;
    Actualizar el número de iteraciones;
Hasta que no se alcance el número máximo de iteraciones;
Imprimir el camino más corto;
Fin
  
```

Resultados

Los resultados obtenidos haciendo uso del paquete *XPRESS* se compararon con los emitidos por el *4APAS*, que inclusive resolvió casos para n entre 8 y 25. El comportamiento del *4APAS* en comparación con el *XPRESS*, utilizando los siguientes valores para los parámetros: $\alpha=0.8$, $\beta=0.5$, $feromona=10$, $Q=1000$, $\rho=0.5$, es mostrado en la tabla 1, la cual contienen los promedios, entre los cinco problemas generados aleatoriamente para cada n , de los costos y los tiempos obtenidos por el *XPRESS* y el *4APAS*.

En la tabla 1 puede observarse cómo el *4APAS* alcanza la solución óptima para $2 \leq n \leq 6$, y para $8 \leq n \leq 25$ el *4APAS* arroja valores pequeños, lo que induce a pensar que son soluciones próximas al óptimo. Además, en la tabla puede observarse que el tiempo es también pequeño.

Tabla1. Resultados obtenidos por el *4APAS*

n	<i>XPRESS</i>	<i>4APAS</i>	gap	Tiempo (seg.)
2	54	54	0	0
3	35	35	0	0
4	19.2	19.2	0	0
6	17.8	17.8	0	0
8	-----	12.6	-----	1.2
10	-----	14.2	-----	3.2
12	-----	19.2	-----	6.0
16	-----	19.6	-----	20.6
20	-----	20.8	-----	57.2
25	-----	29.6	-----	176.2

DISCUSIÓN

El programa trabaja con matrices dinámicas para satisfacer la presencia de objetos dinámicos como resultado de peticiones específicas de asignación de memoria por parte del programa, permitiendo alcanzar un alto grado de eficiencia y rapidez, ya que el acceso a estas matrices (las cuales son de dimensiones muy grandes) se realizó de manera directa.

Los resultados de *XPRESS* comparados con los obtenidos por el *4APAS*, no difieren, considerando el hecho de que las corridas en *XPRESS* fueron para $n=2,3,4,6$.

El algoritmo desarrollado en esta investigación se considera idónea para ser implementado, ya que encontró buenas soluciones para valores de n entre 2 y 25.

AGRADECIMIENTOS

Deseamos agradecer al Consejo de Investigación de la Universidad de Oriente por el aporte concedido para la realización de esta investigación enmarcada dentro del proyecto: "Búsqueda de soluciones a problemas de optimización *NP-difíciles*, a través de algoritmos heurísticos y exactos". De igual forma al Dr. Juan José Salazar, de la Universidad de La Laguna, Tenerife-España, por la colaboración suministrada.

REFERENCIAS

- Almirón M., Chaparro E. & Barán B. (1998). **Optimización basada en sistema de hormigas con heurística de inicialización**, *IX Panel de Informática Expomática'98*, Asunción-Paraguay.
- O. Babaoglu. 1996. **The ant system, optimization by a colony of cooperating agents. *Ant Colony Optimization***. <www.cs.unibo.it/babaoglu/courses/cas/tutorials/AntColonyOptimization.pdf>. (2 de febrero de 2002).
- Balas, E. & Saltzman, M. (1991). **An Algorithm for Tree-Index Assignment Problem**. *Operation Research*, 39(1), 150-161.
- Burkard, R. E. & Rudolf, R. (1993). **Computational investigations on 3-dimensional axial problems**. *Bel. J. of Oper. Res. Stat. and Comp. Sc.*, 32, 85-98.
- Centeno, M.V. & Urbina, J. (2005). **Desarrollo de una aplicación para resolver el problema de asignación 3-dimensional axial a través de un algoritmo de búsqueda tabú**. *ICHIO*.
- González, J. & Centeno, M. V. (2001). **Desarrollo de un algoritmo genético para resolver el 3AP**. *SABER*, 13 (2), 150-161.
- Pressman, R. (1990). **Ingeniería de software: un enfoque práctico**. Madrid-España, Segunda edición, Editorial McGraw-Hill Interamericana.
- Pierskalla, W. P. (1967). **The Multidimensional Assignment Problem**. *Operation Research*, 16, 422-431.

- Taha, H. (1994). *Investigación de Operaciones*. México D.F: Quinta edición, Ediciones Alfaomega, S.A. de C.V.
- “The ant system, optimization by a colony of cooperating agents”. (1996). “**Ant Colony Optimization**” <www.cs.unibo.it/babaoglu/courses/cas/tutorials/Ant_Colony_Optimization.pdf>. (2 de febrero de 2002).

RESEÑA CURRICULAR DE LOS AUTORES

Manuel Vicente, Centeno Romero. Licenciado en Matemática. Magister Scientiarum en Matemáticas. Actualmente realizo mi tesis doctoral en Matemáticas en la Universidad de La Laguna, Tenerife-España, la cual lleva por título “Un estudio metodológico y computacional del problema de asignación multidimensional”. Soy miembro del personal docente y de investigación adscrito al Departamento de Matemáticas del Núcleo de Sucre de la Universidad de Oriente (UDO) y comparto labores docentes en el Programa de la Licenciatura en Informática y en los Postgrados en Matemáticas, Educación Mención Enseñanza de las Matemáticas Básicas, Ingeniería Civil y Gerencia en Informática de dicha institución. He sido y soy investigador responsable de proyectos de investigación, igualmente he asesorado y he sido jurado examinador de varias tesis de pregrado y postgrado y participado como autor y/o coautor de varias publicaciones. También he asistido en condición de ponente a varios congresos tanto nacionales como internacionales. He recibido algunos premios, tales como: premio CONADES 1998. Suficiencia Investigadora, Universidad de La Laguna Tenerife-España 2001. Premio CONABA 2002. Premio Promoción al Investigador (PPI) 2004 (Candidato). Premio Estímulo a la Investigación (PEI) 2006 (UDO). PPI 2007 (Nivel I).

Hebel Alexander, Salazar Salazar. Licenciado en Informática. Actualmente curso estudios de Maestría en Docencia en Educación Superior en la Universidad Gran Mariscal de Ayacucho, Cumaná estado Sucre. Soy miembro del personal docente de la Unidad Educativa Madre Alberta Giménez, Fe y Alegría, dictando las asignaturas Informática Básica, Estructura de Datos, Sistemas Operativos, Telemática y Sistemas de Información. También soy docente en la Universidad Gran Mariscal de Ayacucho sede Cumaná, estado Sucre. He asistido como ponente a varios congresos a nivel nacional.

