

Algoritmo Bioinspirado en inteligencia de enjambres de Optimización de Colonias de Hormigas para el caso del problema de distribución en planta

Jineth Paola Obando Solano¹

Universidad Distrital Francisco José de Caldas-
Facultad Tecnológica

Bogotá D.C., Colombia – Suramérica

Jhonny Zamora²

Universidad Distrital Francisco José de Caldas-
Facultad Tecnológica

Bogotá D.C., Colombia – Suramérica

y

Frank N. Giraldo Ramos³

Universidad Distrital Francisco José de Caldas-
Facultad Tecnológica

Bogotá D.C., Colombia – Suramérica

RESUMEN

Este artículo es resultado de un trabajo de investigación sobre las aplicaciones de algoritmos bioinspirados e inteligentes en el ámbito de ingeniería en producción, en la Universidad Distrital Francisco José de Caldas, abarcando las temáticas de investigación de operaciones, distribución en planta industrial de manufactura entre otros. Se pretende buscar la optimización a problemas propios de esos campos, aplicando inteligencia artificial de enjambres, a partir de la implementación de un algoritmo de Optimización de Colonias de Hormigas (Ant Colony Optimization - ACO) como herramienta metaheurística de planificación y optimización del problema de distribución en planta, con el objetivo de buscar la mejor asignación espacial de estaciones o celdas de trabajo. Se presentan los conceptos teóricos explorados y los resultados obtenidos.

ABSTRACT

This paper is the result of a research on the applications of bio-inspired intelligent algorithms in the field of engineering production in the University Francisco José de Caldas, covering the topics of operations research, distribution, industrial manufacturing facility etc. It is intended to seek the optimization problems of these areas, applying artificial swarm intelligence, from the

implementation of an algorithm by Ants Colonies Optimization (ACO), like a metaheuristic tool for planning and optimization of the Facility Layout Problem, with in order to find the best spatial allocation of stations or work cells. Theoretical concepts explored and results are presented.

Palabras Claves: Problema de distribución en planta, Optimización de colonias de hormigas, agentes artificiales, Optimización combinatorial.

1. INTRODUCCIÓN

El problema de distribución en planta, consiste en buscar la acomodación o configuración más óptima y flexible de máquinas, equipos, recursos, sobre el espacio físico disponible, que faciliten la circulación y manejo de materiales y que permitan a la planta tener un desempeño y flujo de producción optimizados, todo esto bajo un mínimo tiempo y costo de producción total [1]. Tradicionalmente, el estudio de este tipo de problemas sobre distribución y optimización espacial y de flujos, ha sido abordado mediante técnicas de programación dinámica y optimización combinatorial [2] [3] y en investigación de operaciones [4] [5] y suele denominarse “Facility Layout Problem - FLP” [6] [7] [8].

Cuando se asumen como constantes o de forma determinística, las demandas, flujos de materiales y relaciones entre las celdas o estaciones de trabajo que forman la planta, el problema se denomina estático o de periodo sencillo (Single period Facility Layout Problem - SFLP) [9], el cual puede formularse como un problema de asignación cuadrática (Quadratic Assignment Problem - QAP) [10]. Cuando el flujo de materiales entre las celdas o estaciones trabajo no es determinístico y varía, el problema se denomina dinámico o de periodos múltiples (Multiple period Facility Layout Problem - DFLP) [11]. Los proyectos de redistribución de planta (Facilities Relayout Problem, FRLP) consisten en reorganizar los equipos, departamentos y recursos existentes en la planta, con el fin de mejorar su desempeño.

Este artículo se centra en la aplicación de la optimización por colonias de hormigas (ACO), como herramienta para la solución de problemas de optimización combinatorial, como es el caso del problema de distribución en una planta de manufactura industrial. Con ese objetivo se han propuesto diversos algoritmos de optimización, entre los cuales se pueden destacar los basados en inteligencia de colonias de hormigas, que fueron propuestos por Colorni, Dorigo y Maniezzo [12][13][14]. La idea consiste en que si no se garantiza la mejor solución, se alcance la solución más óptima.

¹ Estudiante de Ingeniería en producción en la Facultad Tecnológica - Universidad Distrital Francisco José de Caldas. E-mail: nenita.pao@hotmail.com

² Estudiante de Ingeniería en producción en la Facultad Tecnológica - Universidad Distrital Francisco José de Caldas -mail: xxxxx@yahoo.es

³ Ingeniero en Control Electrónico e Instrumentación, Docente adscrito a la Facultad Tecnológica e Investigador del grupo DIGITI en la Universidad Distrital Francisco José de Caldas. E-mail: fngiraldor@udistrital.edu.co

Los algoritmos de Optimización de Colonias de Hormigas, son técnicas adaptativas y metaheurísticas (algoritmos heurísticos de propósito general que pueden aplicarse a diferentes problemas con modificaciones mínimas), que se inspiran o emulan la conducta de las colonias de hormigas en su hábitat natural, esto es su organización para la búsqueda de recursos o alimento cuyo objetivo es la supervivencia de los individuos [15]. La conducta en la cual se inspira el algoritmo ACO, se denomina forrajeo. Durante la búsqueda de alimento, las hormigas inicialmente exploran los alrededores de su nido de forma aleatoria. Las hormigas comunican la información acerca de la ubicación de una fuente de alimento, marcando ese recorrido mientras se mueven con un rastro de una sustancia denominada feromona la cual se evapora paulatinamente, durante el retorno al nido la hormiga deja una concentración menor o mayor de feromona dependiendo la cantidad y calidad del alimento encontrado en el lugar explorado. De esta forma las demás hormigas encuentran rutas cortas y óptimas hacia fuentes de alimento confiables, ya que las rutas con altos niveles de concentración de feromonas, marcan recorridos donde existe una mayor probabilidad de encontrar alimento en cantidad y calidad, mientras que en aquellas rutas con baja probabilidad de llevar a una fuente de alimento o para rutas demasiado lejanas, van a presentar poco tránsito y por lo tanto se va atenuando con mayor rapidez la concentración de feromonas al perderse la intensidad del rastro por evaporación. Se pueden aplicar entonces, para hacer converger una solución [16].

El algoritmo básico de optimización ACO se presenta a continuación:

1. Inicializar el rastro de feromonas para cada una de las rutas establecidas
2. Para todas las hormigas crear una solución usando las feromonas
3. Evaporar el rastro de feromonas de cada ruta en una cantidad determinada para todas las soluciones.
4. Actualizar el rastro de feromonas para cada solución creada
5. Si el criterio de parada del algoritmo no está satisfecho retornar al paso 2
6. Fin

La optimización puede efectuarse considerando los siguientes requerimientos:

- a. Minimización de la manipulación de material y del recorrido total entre máquinas y departamentos.
- b. Minimización del costo de manipulación del material
- c. Minimización del flujo de material
- d. Minimización de la distancia total recorrida por el material

2. METODOLOGÍA

En el desarrollo de la solución al problema de optimización de colonias de hormigas para el problema de distribución en planta, se implementó el ACO en el software Matlab®, considerando los siguientes pasos:

El modo de operación básico de un algoritmo de optimización de colonia de hormigas ACO, es:

Modelo del Layout: El Layout de la planta puede considerarse discreto, de tal forma que la planta puede dividirse en bloques con la misma forma y área una sección de la planta puede estar formada por varios bloques (Fruggiero, Lambiase, & Negri, 2006).

Inicio (función objetivo): Tratamiento de los valores de los parámetros que se van a considerar para diseñar el algoritmo ACO

$$\max \sum_{i=1}^n \sum_{j=1}^n (r_{ij}) x_{ij} \quad (1)$$

where n = total number of departments; $i = 1, 2, \dots, n$; $j = 1, 2, \dots, n$.

Functional parameters associated with distance-based computation are

- 1 distance between departments in a predefined, which often is Euclidian or rectilinear, distance measurement metric
- 2 frequency of material flow between departments
- 3 unit cost of handling material between departments.

Construcción de una solución por la hormiga k: Se inicia con una solución parcial vacía, que se extiende a cada paso añadiéndole un componente de solución factible elegido entre los vecinos de la solución actual.

Actualización del rastro: Se especifica el estado inicial desde que la hormiga empieza su camino y almacena la componente correspondiente a la memoria de la hormiga anterior, después puede comenzarse una nueva iteración. El resultado final es la mejor solución encontrada a lo largo de todas las iteraciones realizadas.

```

{
Inicio          □ Inicialización de parámetros y estructuras
Para c=1 hasta #Ciclos
  Para k=1 hasta #Hormigas
    ConstrucSoluciónk □ hormiga-k construye solución k
    ActualizarMejorSolución □ selección de la mejor solución
    ActualizarRastro □ actualización de rastro de feromonas
  FinPara
  RegresarMejorSolución
}

```

Inicialización nodos y agentes hormigas:

Tomando $\tau_{ij}(t)$ como la concentración o cantidad de feromona total depositada sobre las rutas ij en un tiempo t y $\eta_{ij}(t)$ como el valor heurístico de la ruta ij en un tiempo t de acuerdo a la medida de la función objetivo. La probabilidad de transición de nodo i , a nodo j , para un periodo de tiempo t será:

$$P_{ij}^k = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{i \in q} [\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta} \\ 0 \end{cases}$$

La tasas de actualización de las feromonas:

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k(t)$$

Donde m es la cantidad de hormigas, ρ la tasa de evaporación y $\Delta\tau_{ij}^k(t)$ es la cantidad de feromona dejada por la hormiga k en cada ruta ij durante un tiempo t

El decrecimiento de la cantidad de feromona:

$$\Delta\tau_{ij}^k(t) = \begin{cases} 1 \\ L^k(t) \\ 0 \end{cases}$$

Evaluación del desempeño de los recorridos:

Todas las rutas iniciales que se plantearon inicialmente, deben ser evaluadas por una función objetivo para determinar su calidad y puntuar su grado de ajuste específico (fitness). Lo cual consiste en este caso, en la evaluación de la sumatoria de cada una de las distancias entre los nodos de la ruta, para determinar cuáles son las secuencias mejor adaptadas de acuerdo a su puntuación. Entonces, para verificar la calidad de cada individuo, se efectúa la sumatoria de la distancia de puntos entre las coordenadas del nodo actual (x_i, y_i) y el nodo siguiente

(x_{i+1}, y_{i+1}) hasta completar los todos los nodos o estaciones que forman la planta para así obtener la longitud total de cada una de las rutas planteadas:

$$f_{objetivo} = \sum_{i=1}^{n-1} \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2}$$

Ec. (1) Sumatoria de distancia coordenadas (x, y) entre nodos

Ahora bien, en el caso de una distribución en planta orientada a una fabricación multi-producto, establecer el costo total del flujo de materiales o del personal entre las diferentes secciones se expresa por la ecuación 3.1.

$$CT_{(min)} = \sum_{i=1}^m \sum_{j=1}^m t_{ij} d_{ij} c_{ij} \quad (3.1)$$

Donde:

m = número de actividades.

t_{ij} = número de cargas o personas a trasladarse durante un período de tiempo.

d_{ij} = distancia manhattan entre los centros de gravedad de las actividades.

c_{ij} = costo de traslado.

Entradas

Mapa, corresponde a las coordenadas x , y de la ubicación de los nodos o estaciones de trabajo.

m - The number of ants (default: 100)

ρ - The evaporation rate (default: 0.1)

α - The pheromone influence parameter (default: 1)

β - The heuristic influence parameter (default: 3)

visualization - Visualize the run of the algorithm (default: true)

% Output:

popt - A vector containing the optimal path

fopt - The length of the optimal path

stat - A MATLAB struct containing algorithm specific output statistics

pheromones = ones(1,1) * m ./ C_nn;

heuristics = ones(1,1) ./ (1e-10 * eye(1) + dmat);

heuristics(1:l+1:l*1) = 0;

pheromones_heuristics_product = pheromones.^alpha .* heuristics.^beta;

Criterio de finalización del algoritmo:

Se itera usando como criterio un máximo número, una vez se alcanza ese valor se culmina el proceso, aunque no en todos los casos se puede garantizar que se llegue a esta solución, sí se puede encontrar una optimizada.

3. RESULTADOS

En la figura, se muestra el plano de la planta sobre la cual se piensa efectuar la optimización de los recorridos, sobre un área de 90 x 42 metros, se indican también los nodos o secciones que la conforman y por los cuales debe pasar la ruta. Se implementó el algoritmo planteado previamente, como un script *m-file* en Matlab®, las coordenadas espaciales (x, y), de ubicación de cada estación o nodo se ingresaron como un vector para la coordenada en x y otro vector para la coordenada en y. El área total en la que está distribuida la planta son 3780 metros cuadrados. La cantidad total de producción es de 1000 piezas mensuales y cada pieza recorre actualmente una distancia de 800m para estar finalizada.

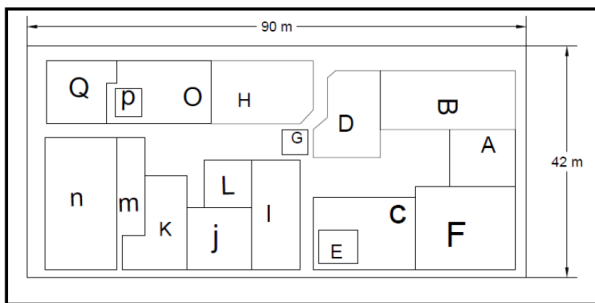


fig. (1), Codificación secciones y estaciones de trabajo de la planta en metros

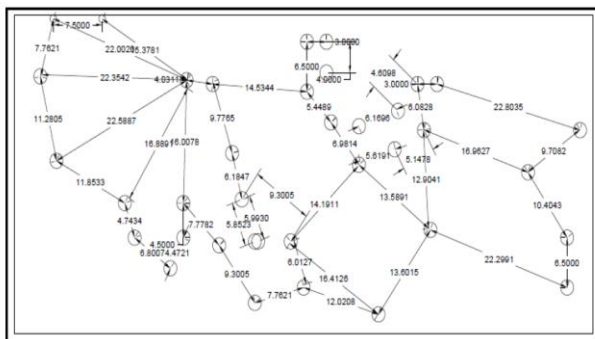


fig. (2), Distancias iniciales en el recorrido de producción en metros

Paso 1: Inicialización de los parámetros de la colonia

El algoritmo finaliza cuando se alcanzan 100 iteraciones. El número de hormigas mínimo debe ser igual al número de nodos o estaciones de trabajo. El peso inicial para la exploración o forrajeo de las hormigas es $q_0=0.5$. El peso

de la información de feromonas inicial es $\alpha = 0.5$. El número de las mejores soluciones de grupo es igual a 20% de la cantidad de hormigas. El número q es uniformemente distribuido entre $[0, 1]$. El valor inicial de feromona τ en cada ruta es inicializado en rango aleatorio entre $(0.1, 0.25)$. El límite inferior del valor de feromona debe ser una constante pequeña y positiva (0.001) para evitar que el algoritmo converga a una solución local.

Paso 2 Construcción de las soluciones en cada iteración t .

Se asigna un paso para cada hormiga k , en la sección i y estación de trabajo o nodo j , aplicando la formula transicional. En primer lugar se toma un número aleatorio q . si $q \leq q_0$, se elige esa sección y nodo de trabajo como parte de la ruta, las hormigas repiten ese proceso hasta agotar la cantidad de secciones y nodos establecidos inicialmente.

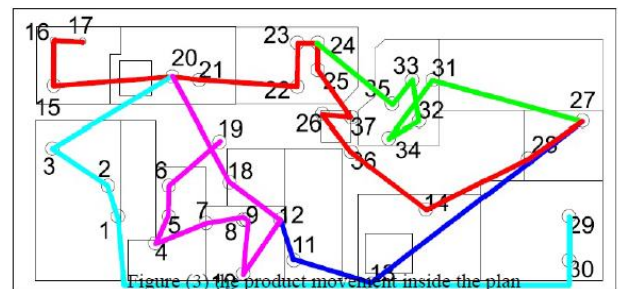


fig. (3), recorrido optimizado

4. DISCUSIÓN

Los costos asociados a la gestión de distribución en planta, dependen entre varios aspectos de la capacidad para optimizar tiempos, flujos de materiales y su manipulación, hasta obtener un producto finalizado. Bajo esa premisa, el alcance de los resultados obtenidos, se evidencia en una estrategia final de ahorro por metro de ruta pero también de optimización del tamaño del flujo de materiales.

La unidad de costos definida son metros recorridos, para estimar los márgenes, se estimó el costo por ruta, para así definir los costos de producción en cada alternativa. En cada uno de los escenarios estudiados, se minimiza también el tiempo total de producción.

Luego de cumplido el número de iteraciones en cada simulación con los escenarios planteados, se efectuaron los cálculos de costos para evaluar la mejor opción y realizar la toma de decisiones: la solución óptima desde el punto de vista de costos de la operación.

5. CONCLUSIONES

A pesar que los algoritmos genéticos no aseguran que las soluciones que se alcancen lleguen al mínimo global, si permiten encontrar un conjunto de soluciones óptimas (rutas de mínima distancia), que brindan un buen rendimiento comparativamente con soluciones obtenidas por otros métodos.

La primera consideración al crear un algoritmo ACO, es definir una representación del problema o modelo del mismo. Se debe definir la función objetivo, teniendo en cuenta, que se pueda alcanzar una mayor aptitud (fitness) y verdaderamente se dé una mejor solución para el problema dado. Los ACO funcionan particularmente bien resolviendo problemas combinatoriales cuyo espacio de soluciones potenciales es demasiado amplio para hacer una búsqueda exhaustiva en un tiempo razonable, como en este caso de optimización de recorridos y flujos para distribución en planta. La característica más importante de utilizar los ACO, es la evaluación de escenarios mediante las técnicas de simulación que permiten recrear las posibles soluciones.

Se pudo evidenciar, el potencial que tienen los ACO en su implementación en problemas de logística, como en el caso de planificación de rutas, con buenos resultados como método de optimización facilitando la toma de decisiones en este campo.

6. REFERENCIAS

- [1] Ballou Ronald H. Logística Administración de la Cadena de Suministro. Quinta Edición, México: Pearson educación, 2004 págs., 14 y 64.
- [3] Bellman, R. & Dreyfus, S. Applied Dynamic Programming. Princeton University Press, 1962.
- [4] Winston, W.L Investigación de Operaciones. Aplicaciones y Algoritmos. Editorial Iberoamérica, 1994.
- [5] V. Maniezzo, L. M. Gambardella, and F. De Luigi, "Ant Colony Optimization," OPTIMIZATION TECHNIQUES IN ENGINEERING. SPRINGER-VERLAG, vol. 141, pp. 101–117, 2004.
- [6] A. Coloni, M. Dorigo, V. Maniezzo, and others, "Distributed optimization by ant colonies," in Proceedings of the first European conference on artificial life, 1991, vol. 142, pp. 134–142.
- [7] M. Dorigo, V. Maniezzo, and A. Coloni, "The ant system: An autocatalytic optimizing process," TR91-016, Politecnico di Milano, 1991
- [8] M.M. Flood, The traveling salesman problem, Operations Research, Vol. 4, No. 1, Feb. 1956.
- [9] G. Dantzig, R. Fulkerson, S. Johnson, Solution of a Large-Scale Traveling-Salesman Problem, Journal of the Operations Research Society of America, Vol. 2, No. 4 (Nov., 1954), págs. 393-410
- [10] Donald Davendra and Yuan-bin Mo. The Advantage of Intelligent Algorithms for TSP, Traveling Salesman Problem, Theory and Applications, 2010.
- [11] Dorigo and C. Blum, "Ant colony optimization theory: A survey," Theoretical Computer Science, vol. 344, no. 2–3, pp. 243–278, 2005.
- [12] Blum, "Ant colony optimization: Introduction and recent trends," Physics of Life Reviews, vol. 2, no. 4, pp. 353–373, 2005.
- [13] M. Dorigo and L. M. Gambardella, "Ant Colonies for the Traveling Salesman Problem," 1997.
- [14] Holland, J.H. - Adaptation in natural and artificial systems, MIT Press, 1975
- [15] Lanlan Kang and Wenliang Cao, "An Improved Genetic & Ant Colony Optimization Algorithm for Travelling Salesman Problem," in 2010 International Symposium on Information Science and Engineering (ISISE), 2010, pp. 498–502.
- [16] T. Stützle and M. Dorigo, "ACO algorithms for the traveling salesman problem," Evolutionary Algorithms in Engineering and Computer Science, pp. 163–183, 1999.
- [17] Kangshun Li, Lanlan Kang, Wensheng Zhang, and Bing Li, "Comparative Analysis of Genetic Algorithm and Ant Colony Algorithm on Solving Traveling Salesman Problem," in IEEE International Workshop on Semantic Computing and Systems, 2008. WSCS '08, 2008, pp. 72–75.
- [18] Contreras, R.: Modelo de optimización en la generación de plantas industriales, considerando las actividades de mantenimiento y las condiciones ambientales mediante el uso de la metodología de los algoritmos genéticos. Tesis para optar el título de doctor. Universidad Politécnica de Valencia, 81-281, (2010).
- [19] Vallhonrat J.; Cormonias A.: Localización, distribución en planta y manutención. Marcombo, 51-53, (1991).
- [20] Dorzán, M.; Gagliardi, E.; Leguizamón, M.; Taranilla, M.: Algoritmos ACO aplicados a problemas geométricos de optimización. Proyecto UPM AL09-PAC12 y Proyecto Tecnologías Avanzadas de Bases de Datos Universidad Nacional de San Luis, Argentina, 1-2, (2009).
- [21] Martí R.: Procedimientos metaheurísticos en optimización combinatoria. Universidad de Valencia. Departament d'Estadística i Investigació Operativa. Facultat de Matemàtiques, 2-9, (2003).
- [22] Mateos, A.: Algoritmos evolutivos y algoritmos genéticos. Universidad Carlos III de Madrid. Inteligencia en redes de comunicaciones. Ingeniería de telecomunicación, 9-10, (2004).