

Optimización de rutas de vehículos con enfoque multiobjetivo mediante Sistema Basado en Colonia de Hormigas en una Empresa de transporte de Personal

John P. Portella Melchor, Carool E. Tomasto Farfan, Hugo F. Vega Huerta, Zoraida Mamani Rodríguez

Facultad de Ingeniería de Sistemas e Informática, Universidad Nacional Mayor de San Marcos, Lima, Perú

Recibido el 15 de junio del 2017. Revisado el 21 de junio del 2017. Aceptado el 1 de julio del 2017

Resumen

El presente proyecto tiene como objetivo mejorar la planificación de rutas mediante la disminución de la distancia total recorrida de viaje y la cantidad de vehículos a emplear en una empresa de transporte, la cual se dedica al traslado privado de personal de sus clientes corporativos. El proceso de planificación de rutas de este tipo de empresas es completamente manual y su factor de éxito es la experiencia del planificador y el conocimiento de los conductores que realizan los recorridos diarios sobre las calles de Lima Metropolitana, siendo estas las principales causas de que se originen los problemas de uso excesivo de vehículos y largas distancias en el recorrido. En ese contexto, la propuesta de solución inicia analizando los procesos de planificación de la empresa, para luego proceder a buscar alternativas de solución expresados en algoritmos que puedan realizar de manera óptima el cálculo de las rutas. Se escoge el Sistema Basado en Colonia de Hormigas, luego se elabora el modelo matemático, se adapta el algoritmo a las condiciones del negocio para luego realizar la implementación, y así finalmente realizar pruebas y calibrar los parámetros del algoritmo con el fin de obtener valores que mejoren los resultados obtenidos. Los beneficios que se obtienen para la empresa son la reducción de vehículos y distancias más cortas, lo que concluye en reducción de costos y mejora de la calidad del servicio.

Descriptores: *Planificación, Optimización, Rutas, Sistema Basado en Colonia de Hormigas, Algoritmo.*

Abstract

This project is aimed at improving route planning by decreasing the total travel distance and the number of vehicles used by a private transportation company dedicated to driving its corporate clients' staff. This type of companies carries out their route planning process manually and their success is highly dependent on the planner's experience and the know-how of drivers who go along the streets of Lima Metropolitana on a daily basis. These are the main reasons behind the problems related to excessive use of vehicles and the long travel distances to cover. In this context, the proposed solution starts by analyzing the company's planning processes, and then proceed to search for alternative solutions expressed in algorithms that can optimally perform the calculation of the routes. A system based on an ant colony is chosen, then a mathematical model is prepared, and the algorithm is adapted to the conditions of this business. Finally, we implements and test it, adjusting the algorithm's parameters in order to get values that improve the obtained results. The benefits for the company are the reduction in the number of vehicles and the shortening of distances, which in turn results in cost reduction and service quality improvement.

Keywords: *Planning, Optimization, Route, Ant Colony System, Algorithms.*

1. Introducción

Los problemas de ruteos de vehículos (VRP) consisten en hallar un conjunto de rutas que optimicen ciertos criterios (distancia, número de

vehículos, etc.) de tal manera que satisfaga toda la demanda de los clientes. Este tipo de problemas suelen representarse como la estructura de un grafo, y dependiendo de las restricciones que se asocien al problema, éste adopta varios nombres según la teoría sustentada por distintos autores. Los problemas en el cual los puntos de inicio y llegada de los vehículos son distintos, se denominan problemas de ruteos de vehículos abiertos (OVRP). Éste tipo de problemas se encuentran dentro de la clasificación de complejidad NP-Completo, ya que es un problema combinatorio cuyas soluciones no pueden ser halladas en tiempo polinomial. Para solucionar este tipo de problemas se aplican distintos métodos como heurísticas, métodos exactos, metaheurísticas, entre otros.

El problema al que se desea dar solución en este estudio, es el de automatizar la planificación y programación de rutas y asignación de vehículos de una empresa de transporte el cual ofrece como servicio brindar movilidad privada a los trabajadores de las empresas contratistas, recogiénolos desde el domicilio de cada uno de sus empleados hasta el mismo centro de trabajo en un plazo de tiempo determinado. Este problema ha sido identificado como un OVRP heterogéneo y con capacidad limitada, ya que los vehículos se pueden encontrar en diferentes depósitos, además de tener diferente capacidad y todos finalizan en un punto en común el cual sería el Centro laboral de la empresa contratista. Para resolver dicho problema, el presente trabajo, muestra una adaptación de la metaheurística "Sistema basado en colonia de hormigas con múltiples objetivos" (MOACS) para la construcción de rutas basadas en criterios de ubicación teniendo en cuenta las restricciones del problema real, además de contar como objetivos principales: el disminuir la cantidad de vehículos a utilizar y la distancia total de recorrido.

2.1. El problema de rutas de vehículos

Según C. Zabala [6] el problema de Ruteo de Vehículos es el nombre genérico dado a una gran familia de problemas referentes a la distribución de mercadería o personal, búsqueda de información o prestación de servicios, a un conjunto de clientes mediante una flota de vehículos. Los vehículos realizan sus recorridos a través de una red de rutas partiendo de puntos fijos, llamados depósitos. Cada tramo de esta red, que puede ser de una sola ruta o bidireccional, tiene asociado un costo o tiempo de viaje que puede depender de muchos factores, como

el tipo de vehículo o el periodo durante el cual el tramo es recorrido. En el estudio de I. García [3] se indica que en estos problemas, generalmente se busca minimizar el costo global de transporte que involucra la distancia total recorrida, el número de vehículos, entre otros factores.

Una revisión de la literatura asociada a la resolución de este tipo de problemas empleando Optimización basada en el Sistema de Colonia de Hormigas, nos muestra que existen diferentes enfoques propuestos por los autores, las cuales brindan un aporte diferente según el contexto del problema. B. Barán y M. Schaerer [1] proponen una variación del tradicional algoritmo (MOACS-VRPTW) el cual tiene dos objetivos ponderados. Requiere optimizar tres elementos: el número de vehículos, el tiempo total de viaje y el tiempo total de entrega, los cuales tienen la misma importancia. Para poder hallar la solución considerando la misma ponderación de objetivos, Barán aporta un contexto multi-objetivo mediante la implementación del Conjunto de Pareto. Esta técnica es la empleada en el presente proyecto para poder obtener soluciones que consideren todos los objetivos.

En un estudio realizado por P. Zabala [7] se propone un algoritmo para dar solución a un CVRPTW, mediante la minimización de la distancia total recorrida. En él se utiliza dos métodos importantes: Heurística de Inserción y Búsqueda local. La Heurística de Inserción permite completar soluciones de las hormigas en las que se hayan omitido nodos, mientras que la Búsqueda Local permite mejorar las soluciones completas de una hormiga. Este nuevo enfoque claramente supera el original algoritmo MOACS – VRPTW, por ello ambas técnicas mencionadas (Búsqueda local e Inserción) son consideradas en el proyecto presentado.

2.2. La metaheurística de optimización basada en colonia de hormigas

Esta Metaheurística fue presentada por M. Dorigo en su Tesis Doctoral en el año 1992; básicamente se trata de una técnica probabilística para resolver problemas computacionalmente complejos que pueden ser reducidos a la búsqueda de buenos caminos en un grafo [2].

J. Rodríguez [5] menciona que los problemas de optimización de rutas se adaptan perfectamente al algoritmo ya que tienen características válidas y se pueden representar mediante grafos.

2.3. Modelos de optimización basados en colonias de hormigas

Las hormigas de la colonia se mueven concurrentemente y de manera asíncrona, a través de los estados adyacentes de un problema, que se representan en forma de grafo con pesos. Este movimiento se realiza siguiendo una regla de transición basada en la información local disponible en las componentes o nodos. Esta información incluye una heurística y una memorística (rastros de feromona) para guiar la búsqueda.

“Calcular las probabilidades de transición” y “Aplicar política de decisión” se tienen en consideración el estado actual de la hormiga y el conjunto de arcos del grafo (A), los valores actuales de la feromona visibles en dicho nodo y las restricciones del problema (W) para establecer el proceso de transición probabilístico hacia otros estados válidos.

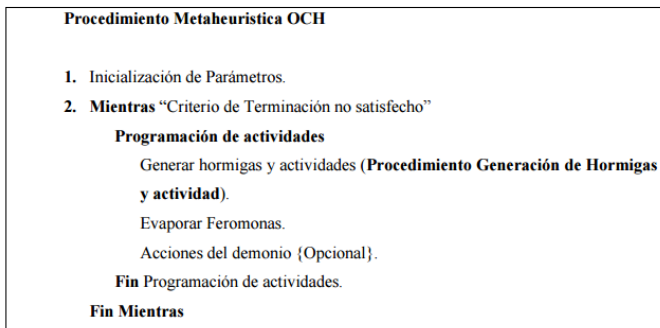


Figura 1: Algoritmo base de una optimización por colonias de hormigas

La “inicialización de parámetros” depende del algoritmo específico, generalmente deben tenerse en cuenta parámetros como: el rastro inicial de feromona asociado a cada transición o arco, el número de hormigas en la colonia, los pesos que definen la proporción en la que afectará la información heurística y memorística en la regla de transición probabilística.

En “programación de actividades” se controla la planificación de tres componentes: la generación y puesta en funcionamiento de las hormigas artificiales; la evaporación de feromona, que se usa como un mecanismo para evitar el estancamiento en la búsqueda y permitir que la hormigas busquen y exploren nuevas regiones del espacio; y las acciones del demonio, utilizadas para implementar tareas desde una perspectiva global que no pueden llevar a cabo las hormigas.

El procedimiento “Actualiza la memoria de la hormiga” se encarga de especificar el estado inicial desde que la hormiga comienza su camino y además almacena la componente correspondiente en la memoria de la hormiga L . En los procedimientos

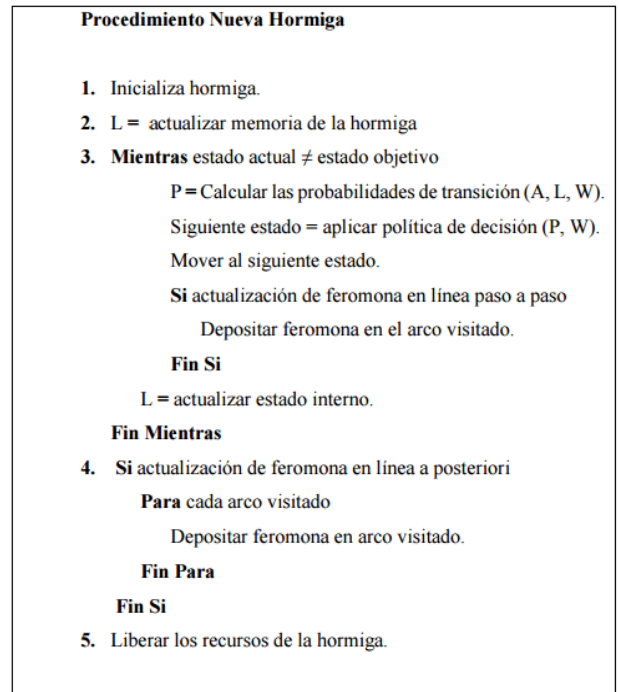


Figura 2: Procedimiento para genera una nueva hormiga

La “actualización de feromona en línea paso a paso” es el procedimiento donde se actualiza el rastro de feromona asociado a un arco, cuando la hormiga se mueve entre los nodos que este conecta. Una vez que la hormiga ha construido la solución puede reconstruir el camino recorrido y actualizar los rastros de feromona de los arcos visitados mediante el procedimiento llamado “Actualización de feromona en línea a posteriori”.

Al introducir el concepto de “Sistema” a la técnica común de Optimización Basado en Colonia de Hormigas, se consideran tres modificaciones importantes:

- Utiliza una regla de transición llamada Regla proporcional pseudo-aleatoria, que depende de un parámetro q . La regla tiene una doble intención:

Explotar el conocimiento disponible, eligiendo la mejor opción con respecto a la información heurística y los rastros de feromona. Y aplicar una exploración controlada.

- La evaporación de feromona y el depósito de feromona se aplica sólo a los arcos del mejor camino global. Para llevarla a cabo, el SCH sólo considera una hormiga concreta, la que generó la mejor solución global. Adicionalmente, se puede aplicar un algoritmo de búsqueda local para mejorar las soluciones de las hormigas antes de actualizar los rastros de feromona.
- Cada vez que una hormiga usa un arco (i, j) para moverse de la ciudad i a la ciudad j, disminuye cierta cantidad de feromona de ese arco [4].

2.4. Conjunto de Pareto

Para aportar el contexto multi-objetivo a la solución se emplea la técnica Conjunto de Pareto. Este es un concepto de la economía que tiene aplicaciones en ingeniería y diferentes ciencias sociales. Dada una asignación inicial de bienes entre un conjunto de individuos, un cambio hacia una nueva asignación que al menos mejora la situación de un individuo sin hacer que empeore la situación de los demás se denomina mejora de Pareto. Una asignación se define como "Pareto eficiente" o "Conjunto Pareto" cuando no pueden lograrse nuevas mejoras de Pareto. La definición técnica podría ser la siguiente: sea P un problema de optimización multi-objetivo. Se dice entonces que una solución S1 es pareto-óptima cuando no existe otra solución S2 tal que mejore en un objetivo sin empeorar al menos uno de los otros.

3. Modelo matemático

El problema a resolver considera 2 objetivos de igual importancia: Reducir la cantidad de vehículos y el recorrido total del viaje. De acuerdo a ello, se tiene las siguientes funciones objetivo:

$$f_1 = \min \sum_{k \in K} \sum_{(i,j) \in G} c_{ij} x_{ij}^k$$

$$f_2 = \min v$$

Asimismo, se presentan las restricciones asociadas a la naturaleza del problema:

- 1- El usuario debe ser atendido una sola vez:

$$\sum_{k \in K} y_{ik} = 1; \forall i \in U$$

$$\sum_{j \in U} x_{ij}^k = y_{ik}; \forall i \in U, k \in K$$

$$\sum_{j \in U} x_{ji}^k = y_{ik}; \forall i \in U, k \in K$$

- 2- Los vehículos solo pueden recoger la cantidad de usuarios que como máximo soporten

$$\sum_{i \in U} y_{ik} \leq c_k; \forall k \in K$$

- 3- Al inicio de cada ruta, cada vehículo recogerá al usuario disponible más lejano al destino y que se encuentre más cerca a este.

$$x_{0j}^k = 1; \forall k \in K, \text{ si } \sum_{j \in U} z_j^k = 1$$

Donde:

x_{ij}^k : Variable binaria igual a 1 si *arco(i, j)* es recorrida por un vehículo *k*.

y_{ik} : Variable binaria igual a 1 si el usuario *i* es recogido por el vehículo *k*.

c_{ij} : Variable positiva que indica el costo del viaje entre el punto *i* y el punto *j*.

q_k : Variable positiva que denota la capacidad máxima de usuarios que puede llevar el vehículo *k*.

v : Variable positiva que indica la cantidad de vehículos empleados para satisfacer a todos los usuarios.

z_j^k : Variable binaria igual a 1 si el cliente *j* está ubicado a una gran distancia respecto al destino y esté más cercano al vehículo *k* que al resto de los vehículos.

K : Universo de todos los vehículos.

U : Universo de todos los usuarios.

G : Universo de todos los arcos.

4. Adaptación del sistema basado en colonia de hormigas con múltiples objetivos

El algoritmo puede ser dividido en bloques según la estructura del algoritmo planteado en [1]; además se agrega un bloque opcional el cual ayuda a mejorar las soluciones obtenidas por el algoritmo.

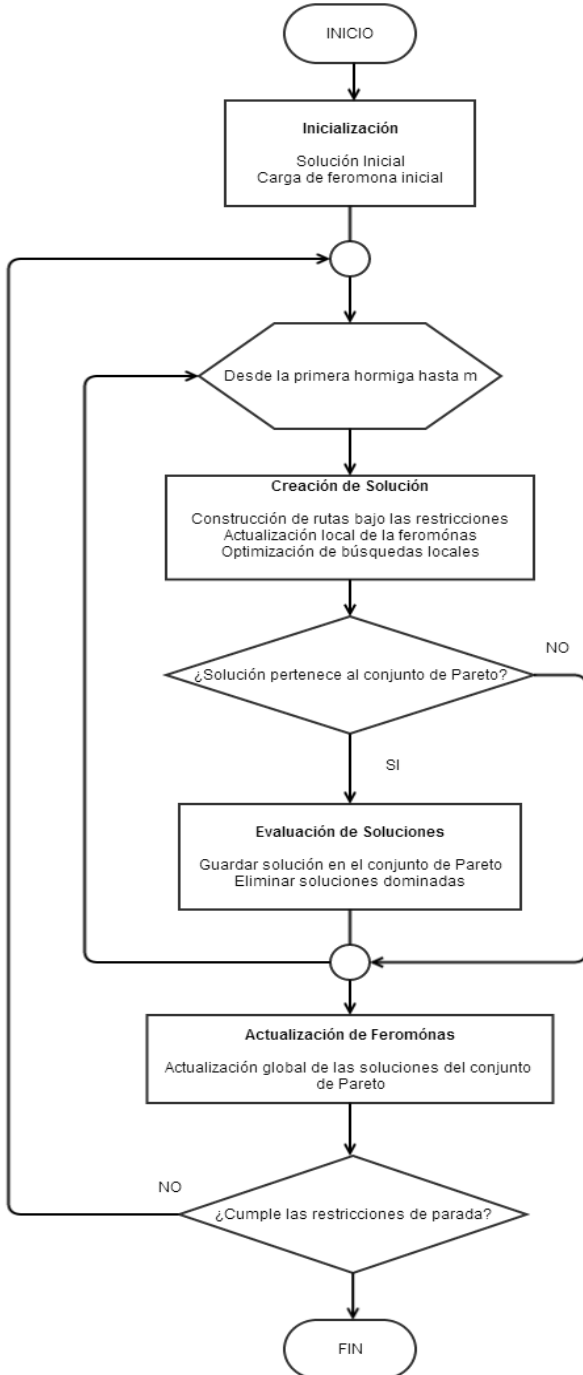


Figura 3: Diagrama de flujo del algoritmo.

En el **Bloque de Inicialización** se realizan los ajustes para que el algoritmo pueda iniciar sus funciones,

incluye la lectura del valor de los parámetros que define el usuario. Se construye una solución inicial mediante la heurística del vehículo más cercano, del cual se obtiene el valor del feromona inicial puesta en todos los arcos.

En el **Bloque de creación de soluciones**, a cada vehículo se le va asignando un nuevo usuario a recoger, que cumpla con las restricciones del problema, dependiendo de los criterios de explotación y exploración propuestos por una variante especificada en [1].

El **Bloque de actualización de la feromona** se encuentra en dos etapas del algoritmo: actualización local, que ocurre cada vez que una solución factible es hallada por las hormigas, y actualización global, la cual se aplica cuando una solución que mejora las contenidas en el conjunto de Pareto es encontrada.

El **Bloque de optimización de Búsquedas Locales** ayuda a mejorar las soluciones que por la naturaleza del algoritmo están sujetos entre los nodos más cercanos (óptimos locales). Para solucionar este problema, se genera soluciones vecinas a la presente, para la cual, se intercambian los arcos de sus rutas. Si la solución vecina es mejor que la actual, se hace una actualización de la solución. El operador de intercambio implementado, es el Crossover, el cual combina dos rutas de una solución sin perder la orientación de las rutas. Este bloque ayuda a mejorar la distancia total recorrida por los vehículos, siendo únicamente favorable para la primera función objetivo.

En el **Bloque de la evaluación de las soluciones multiobjetivo**, se utiliza el conjunto de Pareto en el cual inicialmente se agrega la primera solución encontrada. Si una mejor solución es hallada y es no dominada por las que se encuentran en el conjunto, se agrega al conjunto de Pareto.

Sistema de optimización de rutas

El sistema para resolver el problema de optimización de rutas y vehículos tiene el nombre de "Planificador MOACS", en el cual se implementa la metaheurística de optimización basada en colonia de hormigas multiobjetivo, con el algoritmo de mejora y la implementación del conjunto de Pareto para reunir las mejores soluciones factibles. Este presenta la siguiente arquitectura (Fig.4):

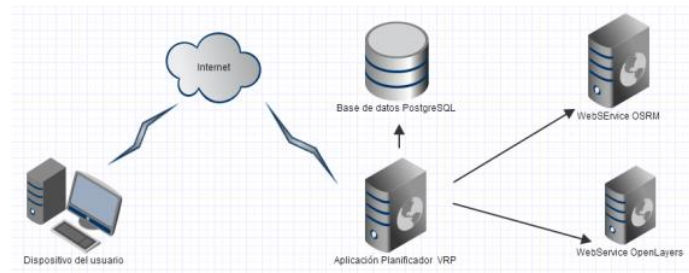


Figura 4: Arquitectura de la aplicación

Dispositivo del usuario: Representa el medio electrónico mediante el cual el usuario (planificador) accede a la aplicación. Puede utilizar cualquier equipo que tenga una conexión a Internet.

Base de Datos PostgreSQL: En esta base de datos son almacenados todos los datos que el usuario añade a través de la aplicación, los datos de distancias dados por el web service OSRM, además de las soluciones que resultan de la planificación de las rutas.

Aplicación Planificador VRP: Representa la aplicación web del proyecto.

Web Service OSRM: Este servicio web permite hallar la distancia mínima entre dos puntos geográficos. Se

emplea en el proyecto para realizar el cálculo de las distancias entre los nodos participantes (vehículos, destino, trabajadores), las cuales se toman como distancias base para las operaciones que realiza el algoritmo.

Web Service OpenLayers: Servicio que brinda toda la información referente a mapas geográficos, así como imágenes, nombres de las calles, ubicación gráfica de los puntos geográficos, entre otros, y que permite interactuar con estos mapas a través de la colocación de capas de información. Se utiliza en el proyecto como base para mostrar la ubicación geográfica de los nodos (vehículo, empresa y trabajadores) y para presentar gráficamente las soluciones resultado de la planificación.

Además, presenta los siguientes módulos:

- Módulo de empresa, en el cual se carga información de las empresas clientes, sus empleados, se realiza la planificación de las rutas y se almacenan sus soluciones (Fig.5).
- Módulo de vehículo, en el cual se carga información de los vehículos y los depósitos.



Figura 5: Visualización de Soluciones obtenidas por la planificación

5. Experimentación y validación

5.1. Calibración

La calibración nos permite hallar un rango de valores óptimos para cada parámetro del algoritmo que nos

asegure que se obtendrán los mejores resultados en cada iteración. Esta calibración se realizó mediante la creación de dos instancias de datos (Tabla 1 y Tabla 2). Para cada ejecución se realizaron 100 ejecuciones por instancia. Los resultados se muestran en (Tabla 3).

Tabla 1: Valores de la instancia 1

Instancia 1:

| | | | | | |
|---|----|---|---|---|---|
| Cantidad de usuarios | 20 | | | | |
| Capacidad de vehículos disponibles | 7 | 7 | 3 | 3 | 3 |

Tabla 2: Valores de la instancia 2

Instancia 2:

| | | | | | | | | |
|---|----|---|---|---|---|---|---|---|
| Cantidad de usuarios | 26 | | | | | | | |
| Capacidad de vehículos disponibles | 7 | 7 | 7 | 3 | 3 | 3 | 3 | 3 |

Tabla 3: Valores de los parámetros del resultado de la experimentación

Resultado:

| | Instancia 1 | Instancia 2 |
|---------------------------------|--------------------|--------------------|
| Evaporación | 0.2 | 0.4 |
| Ponderado de Visibilidad | 10 | 10 |
| Ponderado de Rastro | 0.5 | 0.5 |
| Cantidad de Hormigas | 200 | 200 |
| Cantidad de Iteraciones | 250 | 250 |

5.2. Validación

Para la validación de los objetivos se emplean 2 instancias descritas en Tabla 4 y Tabla 5. La comparación de los resultados obtenidos por el sistema versus los obtenidos por la planificación tradicional se visualizan en Tabla 6

Tabla 4: Resultado de la experimentación para la instancia 1

Instancia 1:

| Rutas | Capacidad | Recorrido | Clients | Distancia Total (m) | Errores +/-10 por nodo(m) |
|--------|-----------|--|---------|---------------------|---------------------------|
| Ruta 1 | 7 | Depósito - 6 - 1 - 4 - 3 - 2 - 5 - 7 - Empresa | 7 | 36437 | +/- 90 |
| Ruta 2 | 7 | Depósito - 9 - 10 - 11 - 19 - 18 - 20 - 12 - Empresa | 7 | 17001 | +/- 90 |
| Ruta 3 | 3 | Depósito - 8 - 17 - 16 - Empresa | 3 | 27991 | +/- 50 |
| Ruta 4 | 3 | Depósito - 15 - 14 - 13 - Empresa | 3 | 17675 | +/- 50 |
| | | | 20 | 99104 | |

Tabla 5: Resultado de la experimentación para la instancia 2

Instancia 2:

| Rutas | Vehículo Capacidad | Recorrido | Clients | Distancia Total (m) | Errores +/-10 por nodo(m) |
|--------|--------------------|--|---------|---------------------|---------------------------|
| Ruta 1 | 7 | Depósito - 14 - 15 - 16 - 18 - 17 - Empresa | 5 | 22436 | +/- 70 |
| Ruta 2 | 7 | Depósito - 10 - 6 - 3 - 7 - 1 - Empresa | 5 | 21131 | +/- 70 |
| Ruta 3 | 7 | Depósito - 2 - 20 - 4 - 9 - 11 - 13 - 19 - Empresa | 7 | 25980 | +/- 90 |
| Ruta 4 | 3 | Depósito - 21 - 22 - Empresa | 2 | 23392 | +/- 40 |
| Ruta 5 | 3 | Depósito - 8 - 5 - 12 - Empresa | 3 | 29020 | +/- 50 |
| Ruta 6 | 3 | Depósito - 26 - 25 - Empresa | 2 | 31087 | +/- 40 |
| Ruta 7 | 3 | Depósito - 23 - 24 - Empresa | 2 | 24612 | +/- 40 |
| | | | 26 | 177658 | |

Tabla 6: Comparación de resultados de las instancias
Comparación de resultados:

| Instancia | Real | | Propuesto | | Mejora (%) | |
|-----------|---------------|-----------|---------------|-----------|------------|-----------|
| | Distancia (m) | Vehículos | Distancia (m) | Vehículos | Distancia | Vehículos |
| I1 | 98824 | 4 | 93634 | 4 | 5.25 | 0.00 |
| I2 | 177258 | 7 | 139369 | 5 | 21.38 | 28.57 |

6. Conclusión y trabajos futuros

Las restricciones del modelo matemático se adecuan a la proporcionada por los expertos del negocio. Es por tal motivo, que ésta solución sólo es útil con empresas cuyos negocios son similares. Además, El algoritmo implementado da soluciones factibles desde la primera iteración, las cuales pueden mejorar según la cantidad de iteraciones. Además, cuenta con un módulo para evitar los óptimos locales y así tratar de buscar cada vez mejores soluciones. En cuanto a la implementación, se emplearon componentes externos de software con licencia GPL (Licencia Pública General).

Como trabajo futuro, se considerará la solución a eventos inesperados en tiempo real (calles cerradas, vehículos averiados, pasajeros que no puedan ser recogidos por algún imprevisto, etc.), pues la planificación actual es estática, es decir, si ocurre algún evento inesperado, se debe realizar nuevamente la planificación. Para esto se deberá obtener la información necesaria de repositorios externos y adaptar el algoritmo implementado a estos nuevos procesos. Además, como una forma de ampliar las funcionalidades del software, se propone como trabajo futuro el incluir al cliente como un usuario más del sistema, de tal manera que estos aprovechen la información proporcionada por el producto.

Referencias

- [1] B. Barán, M. Schaerer, in *The 21st IASTED International Multi-Conference on Applied Informatics*, Innsbruck, Austria, 2003.
- [2] M. Dorigo, T. Stützle, *Ant Colony Optimization* (Massachusetts Institute of Technology, 2004).
- [3] D. García, *Un Enfoque Metaheurístico para un Problema de Ruteo con flexibilidad en las fechas de entrega*. Tesis Doctoral, Universidad Autónoma de Nuevo León, 2010.
- [4] E. Menéndez, *Metaheurística de Optimización Mediante Colonia de Hormigas y Aplicaciones*, (Universidad Central "Marta Abreu" de las Villas, 2005).
- [5] J. Rodríguez, *Análisis de algoritmos basados en colonia de hormigas en problemas de camino mínimo*. Proyecto Final de Carrera, Universidad Carlos III de Madrid, España, 2010.
- [6] C. Zabala, *Implementación del Sistema de Colonia de Hormigas con Búsqueda Local al Problema de Ruteo de Vehículos con Capacidad y Ventanas de Tiempo (CVRPTW)*, Disponible en www.researchgate.net/publication/, 2005.
- [7] P. Zabala, *Problemas de Ruteo de Vehículos*. Tesis Doctoral, Universidad de Buenos Aires, Argentina, 2006.