Solución de problemas de ruteo de vehículos con restricciones de capacidad usando la teoría de grafos

Solving capacitated vehicles routing problems using graph theory

Alexander Correa Espinal, Ph. D., Juan Cogollo Flórez, M. Sc. & Juan Salazar López, M. Sc. (c)
Escuela Ingeniería de la Organización, Facultad de Minas, Universidad Nacional de Colombia, Medellín.
alcorrea@unal.edu.co; jmcogollof@unal.edu.co; jcsalazarl@unal.edu.co

Recibido para revisión 07 de marzo de 2011, aceptado 18 de octubre de 2011, versión final 27 de octubre de 2011

***Resumen* — La teoría de grafos es una herramienta para la solución de problemas de ruteo de vehículos con restricciones de capacidad (Capacitated Vehicle Routing Problem, CVRP). En este artículo se muestra la aplicación de una herramienta informática basada en la teoría de grafos para analizar y resolver un CVRP en una empresa de transporte de carga a nivel nacional. El análisis se realizó en una flotilla de 13 vehículos con las mismas especificaciones técnicas y capacidad de carga similar (15 Toneladas). El origen escogido fue Medellín y los municipios de su área metropolitana. Los destinos se concentraron en 4 ciudades: Cartagena, Bogotá, Buenaventura y Cúcuta.Se obtuvieron dos rutas óptimas: la primera ruta es cubierta por un vehículo con un recorrido de 2,347 Km y la segunda ruta es cubierta por un vehículo con un recorrido de 1,761 Km. Es posible obtener un ahorro de 21.9% en los recursos utilizados dado que las rutas pueden ser cubiertas usando 2 vehículos menos.**

***Palabras clave* —Teoría de grafos, ruteo, algoritmos, costos, mejora, distribución.**

Abstract— Graph theory is atool for solving Capacitated Vehicle Routing Problem (CVRP). This paper describes the application of software based on graph theory to analyze and solve CVRP in a nationwide freight company.The analysis was performed on a fleet of 13 vehicles with similar technical specifications and load capacity (15 tons). Medellin and its metropolitan area municipalities were the origin chosen. The destinations were 4 cities: Cartagena, Bogota, Buenaventura and Cucuta.Two optimal routes were obtained. The first one is covered using a vehicle with a distance of 2,347 Km. The second one is covered using a vehicle with a distance of 1,761 Km.Saving of 21.9% in used resources was obtained as routes can be covered using 2 vehicles less.

***Keywords* — Graph theory, routing, algorithms, costs, improvement, distribution.**

I. INTRODUCCIÓN

El estudio de los problemas de ruteo surge a mediados del siglo pasado con la proposición del modelo matemático del problema del agente viajero (Traveling Salesman Problem, TSP), a partir del cual muchas investigaciones se han dedicado de lleno a estos problemas en todos los casos particulares y con aplicaciones en el mundo real. En las últimas décadas ha aumentado el desarrollo de herramientas informáticas para la resolución de problemas reales de diseño de rutas de transporte, basados en modelos conceptuales inspirados en sistemas biológicos, inteligencia artificial, teorías matemáticas, entre otros, a los cuales se les programan e incorporan algoritmos y funciones dependiendo del problema a solucionar.

En este artículo se desarrolla la aplicación de una herramienta informática basada en la teoría de grafos para el análisis y solución de un CVRP en una empresa de transporte de carga pesada a nivel nacional. Inicialmente, en la sección 2, además de presentar una revisión conceptual de la teoría de grafos y del CVRP, se hace una breve descripción del software utilizado (*Grafos*), sus características, algoritmos y análisis implementados. En la sección 3 se describe el problema a solucionar y se explica detalladamente el procedimiento para su solución. En la sección 4 se muestran los resultados computacionales y se hace una discusión de los mismos. Finalmente, en la sección 5 se enuncian las conclusiones producto de este trabajo.

II. TEORÍA DE GRAFOS Y RUTEO DE VEHÍCULOS

2.1 Teoría de grafos

Muchos problemas de planificación de rutas de distribución han encontrado alternativas de solución en la teoría de grafos

Revista Avances en Sistemas e Informática, Vol.8 No. 3, diciembre de 2011 - Medellín. ISSN 1657-7663

dado que se facilita su modelamiento por la semejanza conceptual de las estructuras. Al igual que las rutas de distribución, los grafos son estructuras discretas que constan de vértices conectados mediante arcos.Un grafo dirigido (Figura 1) se denota por $G = (V, A)$, donde V es un conjuntono vacío de elementos denominados vértices y A es un conjuntode arcos. Cada $a \in A$ tiene asociados dos vértices de V, i y j, $i \neq j$; a $i$se le denomina origen del arco y a j, destino del arco. El arco $a$también sedenota por (i, j), de esta forma se hace referencia al vértice origen y al vértice destino del arco [1].

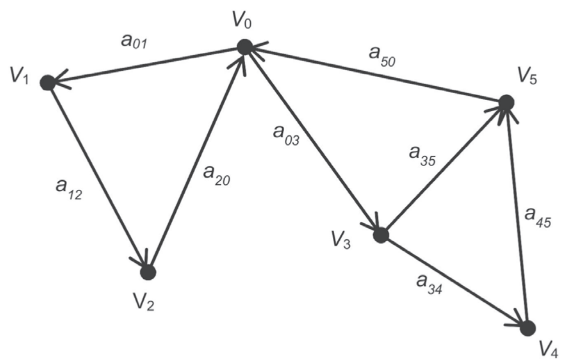


Figura 1. Ejemplo de un grafo dirigido

2.2 *Problema de ruteo de vehículos con restricciones de capacidad (CVRP)*

El CVRPes un problema fundamental de optimización combinatoria con aplicaciones en la práctica logística. En las últimas dos décadas se han logrado grandes avances en su solución debido, principalmente, a mejores algoritmos y al aumento creciente de las capacidades de los equipos de cómputo [2]. En el CVRP se da un conjunto finito de ciudades y los costos de viajes entre ellas; una ciudad específica es identificada como el depósito de los vehículos y el resto como los clientes. Cada cliente corresponde con una localización donde se entrega una cantidad de un único producto. Las cantidades demandadas por los clientes están determinadas previamente y no se pueden dividir, es decir, que tienen que ser entregadas a un vehículo de una sola vez. En la versión más simple se supone que los vehículos son homogéneos y, por lo tanto, tienen la misma capacidad máxima [3].

El CVRP también se formula como un problema de teoría de grafos. Se considera un grafo completo $G = (V, E)$, donde $V := \{0, 1, \ldots, n\}$ es el conjunto de vértices y E el conjunto de aristas entre cada dos vértices. Se denota por 0 el vértice que corresponde con el depósito de los vehículos y los vértices en$\{1, \ldots, n\}$ los distintos clientes. Para una arista $e = [i, j]$ denotamos por c_e elcosto de ir de i a j. Hay una flota de Kvehículos, cada uno de capacidad Q.Finalmente, se denota por d_i la demanda del cliente i. Una variable binariax_e indica si la arista $e$está en la ruta de un vehículo o no. Su formulación matemática es [3,4]:

$$\min \sum_{e \in E} c_e x_e \qquad (1)$$

Sujeto a:

$$x(\delta(\{i\})) = 2\, paratodo\ \mathrm{i} \in \mathrm{V}\backslash\{0\} \qquad (2)$$

$$x(\delta(\{0\})) = 2K \qquad (3)$$

$$x(\delta(S)) \geq \frac{2}{Q} \sum_{i \in S} d_i\, paratodoS \subset V\backslash\{0\} \qquad (4)$$

$$x_e \in \{0,1\} = 2\, paratodo\ \mathrm{e} \in \mathrm{E} \qquad (5)$$

La familia de igualdades (2) impone que el grado de cada vértice correspondiente a cada cliente es exactamente 2, es decir, que cada cliente sea visitado exactamente una vez por un vehículo. La igualdad (3) impone que el grado del depósito sea $2K$. Las desigualdades (4) y (5) fuerzan la biconexidad de una solución entera y que un conjunto de clientes que supera la capacidad máxima Q no pueda ser visitado por el mismo vehículo [3].Las soluciones de este problema son todas las K rutas que verifican la restricción de capacidad de los vehículos [5].

2.3 *Proyecto Grafos*

Grafos es unsoftware libre para la construcción, edición y análisis de grafos de utilidad para la docencia, aprendizaje y práctica de la teoría de grafos y otras disciplinas relacionadas como la investigación operativa, diseño de redes, ingeniería de organización industrial, la logística y el transporte, etc. Incorpora algoritmos y funciones que permiten modelar, diseñar y analizar problemas reales [6].El proceso del software tiene como pilares el desarrollo de un interfaz para la construcción y edición de grafos y el desarrollo de una estructura de clases y librerías *.dll* con algoritmos de resolución yanálisis de problemas de teoría de grafos.

El usuario puede dibujar libremente el grafo sin preocuparse del análisis o algoritmo que utilizará posteriormente. El programa le avisará en caso de no factibilidad o de cualquier otro requerimiento para un análisis en particular. Esta libertad de cara al usuario implica una mayor complejidad en el desarrollo del código fuente, ya que no sólo hay que contemplar los supuestos aplicables al tipo de análisis o problema a resolver, sino cualquier escenario que pueda generar la interacción con el usuario (incluyendo los de no factibilidad) [6].

Grafos permite la construcción tanto de grafos dirigidos como no dirigidos. Los arcos pueden tener valores asociados de costo o distancia, flujo mínimo y flujo máximo. Asimismo, los nodos además de tener una etiqueta de identificación pueden tener un valor asociado (peso del nodo, demanda o capacidad de producción). El usuario puede personalizar el grafo con estilos de arco, estilos de nodo, trazos y colores. La distribución del grafo la decide el usuario, aunque el programa le puede ayudar con funciones que dibujan el grafo automáticamente (formato de árbol, radial, orgánico *force directed*, flujo, aleatorio, etc.). También se pueden importar o exportar las coordenadas de los nodos, con la posibilidad de incorporar un mapa como fondo del grafo. El programa puede además calcular la distancia entre nodos e introducir este valor automáticamente (o un costo proporcional) en los arcos del grafo [6,7].

Dentro de la denominación de problemas de rutas o recorridos se engloba todo un amplio conjunto de variantes y personalizaciones de problemas. Desde aquellos más sencillos hasta algunos mucho más complejos que incluso hoy en día son materia de investigación. Todos ellos, sin embargo, además del reto computacional que representan, tienen en común su gran importancia en investigación operativa por su aplicación práctica en la realidad. Al igual que el TSP, la mayoría de los problemas VRP son de complejidad NP-completo, porque el número de posibles soluciones crece exponencialmente con el número de nodos del grafo (ciudades o puntos de paso), y rápidamente sobrepasa las capacidades de cálculo de los computadores más potentes**.** Para resolver éstos últimos, *Grafos* cuenta con la ayuda de *lp_solve*; un *solver* de programación lineal entera mixta de licencia libre (*LGPL - GNU lesser general public license*), el cual resuelve modelos de programación lineal (mixta) puros, con variables enteras/binarias, conjuntos semicontinuos y *special ordered sets* (SOS) [6, 8,9].

A continuación se detallan los algoritmos y análisis implementados [6]:

Algoritmo de Dijkstra (camino mínimo): el problema de la ruta más corta se puede resolver utilizando programación lineal; sin embargo, debido a que el método simplex es de complejidad exponencial, se prefiere utilizar algoritmos que aprovechen la estructura en red que se tiene para estos problemas. Para ello, el algoritmo mantiene un conjunto *S* de nodos cuyos pesos finales de camino mínimo desde el nodo origen ya han sido determinados.

- ***Algoritmo de Floyd-Warshall*:** el problema que intenta resolver este algoritmo es el de encontrar el camino más corto entre todos los pares de nodos o vértices de un grafo. Esto es semejante a construir una tabla con todas las distancias mínimas entre pares de ciudades de un mapa, indicando además la ruta a seguir para ir de la primera ciudad a la segunda. Existen varias soluciones a este problema y los algoritmos a aplicar dependen también de la existencia de arcos con pesos o costos negativos en el grafo. En el caso de no existir pesos negativos, sería posible ejecutar *V* veces el Algoritmo de Dijkstra para el cálculo del camino mínimo, donde *V* es el número de vértices o nodos del grafo.
- ***Algoritmo de Bellman-Ford***: soluciona el problema de la ruta más corta o camino mínimo desde un nodo origen, de un modo más general que el Algoritmo de Dijkstra, ya que permite valores negativos en los arcos. El algoritmo devuelve un valor booleano si encuentra un circuito o lazo de peso negativo. En caso contrario calcula y devuelve el camino mínimo con su coste. Para cada vértice *v* perteneciente a *V*, se mantiene el atributo *d*[*v*] como cota superior o costo del camino mínimo desde el origen *s* al vértice *v*.
- ***Algoritmo de Dijkstra (árbol mínimo)***: es un algoritmo de trayectoria más corta, rutea cada vehículo a lo largo de la trayectoria de longitud mínima (ruta más corta) entre los nodos origen y destino. Hay varias formas posibles de seleccionar la longitud de los enlaces. La forma más simple es que cada enlace tenga una longitud unitaria, en cuyo caso, la trayectoria más corta es simplemente una trayectoria con el menor número de enlaces. De una manera más general, la longitud de un enlace puede depender de su capacidad de transmisión y su carga de tráfico. La solución es encontrar la trayectoria más corta que contenga pocos enlaces no congestionados; de esta forma los enlaces menos congestionados son candidatos a pertenecer a la ruta.
- ***Algoritmo de Kruskal***: el objetivo del algoritmo de Kruskal es construir un árbol (subgrafo sin ciclos) formado por arcos sucesivamente seleccionados de mínimo peso a partir de un grafo con pesos en los arcos. Un árbol (spanning tree) de un grafo es un subgrafo que contiene todos sus vértices o nodos. Un grafo puede tener múltiples árboles. La aplicación típica de este problema es el diseño de redes telefónicas. Otra aplicación menos obvia es que el árbol de coste total mínimo puede ser usado como solución aproximada al problema del viajante de comercio. La manera formal de definir este problema es encontrar la trayectoria más corta para visitar cada punto al menos una vez. Si se visitan todos los puntos exactamente una vez, lo que se tiene es un tipo especial de árbol. Si se tiene una trayectoria que visita algunos vértices o nodos más de una vez, siempre se puede soltar algunos nodos del árbol. En general, el peso del árbol total mínimo es menor que el del viajante de comercio, debido a que su minimización se realiza sobre un conjunto estrictamente mayor.
- ***Algoritmo de Prim***: consiste en dividir los nodos de un grafo en dos conjuntos: procesados y no procesados. Al principio, hay un nodo en el conjunto procesado que corresponde al equipo central; en cada interacción se incrementa el grafo de procesados en un nodo (cuyo arco de conexión es mínimo) hasta llegar a establecer la conexión de todos los nodos del grafo a procesar. De la misma manera, se puede calcular el árbol de costo máximo.
- ***Algoritmo de Ford-Fulkersson***: se puede considerar un grafo como una red de flujo, donde un nodo fuente produce o introduce en la red cierta cantidad de algún tipo de material, y un nodo sumidero lo consume. De igual modo que en redes eléctricas (Ley de Kirchhoff), la suma de flujos entrantes a

Revista Avances en Sistemas e Informática, Vol.8 No. 3, diciembre de 2011 - Medellín. ISSN 1657-7663

un nodo debe ser igual a la suma de los salientes (principio de conservación de energía), excepto para el nodo fuente y el nodo sumidero. Por tanto, el problema de flujo máximo se enuncia como: ¿cuál es la tasa a la cual se puede transportar el material desde el nodo fuente al nodo sumidero, sin violar las restricciones de capacidad? Este algoritmo se puede usar para resolver modelos de: transporte de mercancías (logística de aprovisionamiento y distribución), flujo de gases y líquidos por tuberías, componentes o piezas en líneas de montaje, corriente en redes eléctricas, paquetes de información en redes de comunicaciones, tráfico ferroviario, sistema de riego y otros.

III. APLICACIÓN DE LA TEORÍA DE GRAFOS EN LA SOLUCIÓN DEL CVRP

En las empresas de transporte de carga, el proceso de asignación de rutas se realiza, generalmente, sin utilizar herramientas científicas, sólo bajo el criterio de personas“expertas” en el tema y de acuerdo a los requerimientos de carga de los clientes. Dado que los costos de transportes representan entre uno y dos tercios de los costos logísticos totales, el mejoramiento de su eficiencia mediante la máxima utilización del equipo de transporte y de su personal, es un asunto importante para la empresa. Un problema frecuente en la toma de decisiones es reducir los costos de transporte y mejorar el servicio al cliente encontrando las mejores rutas a seguir por los vehículos que minimicen el tiempo o la distancia total del recorrido.

El problema modelado consistió en una flotilla de 13 vehículos con las mismas especificaciones técnicas y capacidad de carga similar (15 Toneladas). El origen escogido fue Medellín y los municipios de su área metropolitana. Los destinos se concentraron en 4 ciudades: Cartagena, Bogotá, Buenaventura y Cúcuta.

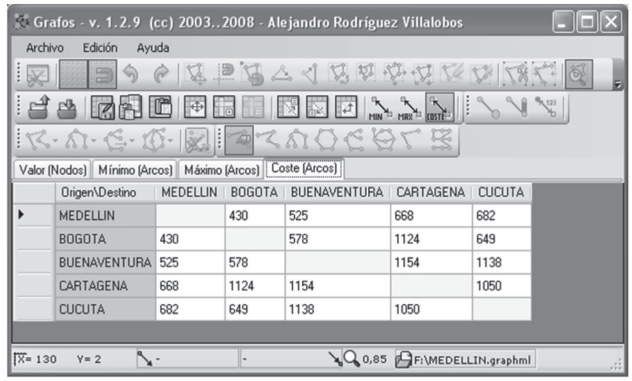
Las demandas de carga fueron:

- Cartagena :315 Ton.
- Cúcuta : 30 Ton.
- Bogotá: 345 Ton.
- Buenaventura: 60 Ton

El desarrollo de la solución del CVRP aplicando *Grafos* se hizo siguiendo los pasos sugeridos por Rodríguez [6]: ingreso de la matriz de distancias entre ciudades, generación de los nodos con sus respectivas oferta y demanda de carga y establecimiento de las restricciones y variables de operación de los vehículos.

3.1 Matriz de distancias

En la Figura 2se ilustra el procedimiento de ingreso de la matriz de distancias entre las ciudades origen y destino.



Origen\Destino	MEDELLIN	BOGOTA	BUENAVENTURA	CARTAGENA	CUCUTA
MEDELLIN		430	525	668	682
BOGOTA	430		578	1124	649
BUENAVENTURA	525	578		1154	1138
CARTAGENA	668	1124	1154		1050
CUCUTA	682	649	1138	1050	

Figura 2. Matriz de distancias.

3.2 Generación de nodos

La Figura 3 muestra el resultado de la fase de generación del mapa con los grafos correspondientes a la matriz de distancias origen – destino, con las cargas ofertadas (signo positivo) y demandadas (signo negativo)en cada ciudad y las distancias entre ellas (en kilómetros) establecidas en el paso anterior.

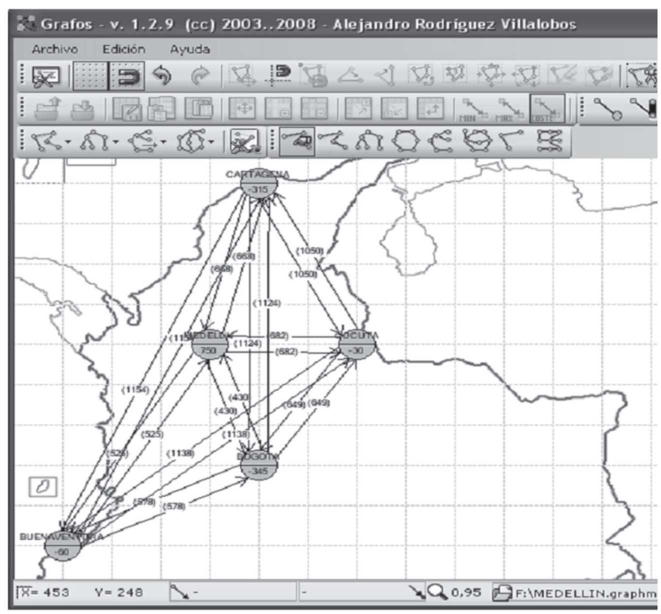


Figura 3. Esquema de los nodos del problema.

3.3 Restricciones

Las restricciones de los vehículos están dadas por las condiciones operacionales propias de la empresa, las cuales se rigen por aspectos legales e intrínsecos de este tipo de negocios. Estas restricciones se calcularon teniendo en cuenta que un vehículo trabaja 10 horas diarias, 6 días a la semana, 4 semanas por mes y a una velocidad promedio de 60 Km/h.

Las unidades utilizadas para el ingreso o definición de las restricciones en el modelamiento (Figura 4) son las siguientes:

- Demanda y oferta de carga,en Toneladas/mes.

- Costos fijos y variables, en $/mes
- Capacidad de carga de los vehículos, en Toneladas/mes.
- Máxima distancia de los vehículos, en Kilómetros/mes.

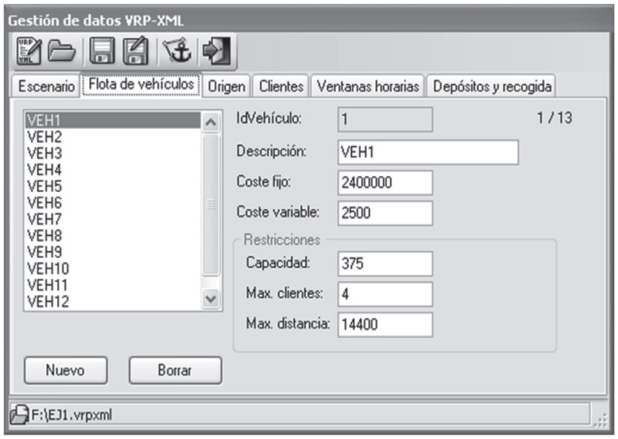


Figura 4. Configuración de restricciones.

Las restricciones finales fueron las siguientes:

- Costo Fijo = $ 2,400,000 /mes en cada vehículo.
- Costo Variable = $ 2,500/ Km en cada vehículo.
- Capacidad de carga = 375 Toneladas/mes
- Máxima distancia recorrida = 14,400 Km/mes

IV. RESULTADOS Y DISCUSIÓN

4.1 Resultados

Los resultados del modelamiento se resumen en forma gráfica en la Figura 5.

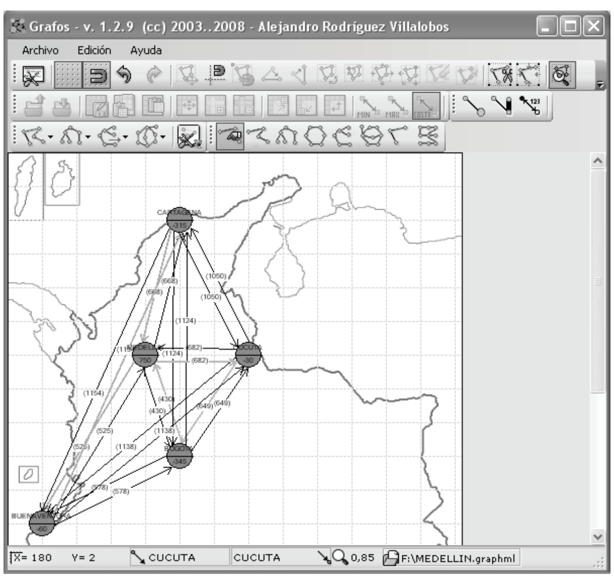


Figura 5. Solución gráfica del CVRP.

Los resultados detallados que arroja *Grafos*para cada una de las dos rutas obtenidas se presentan en la Tabla 1.

Tabla 1. Resultados de la solución del CVRP

RESULTADO	RUTA 1	RUTA 2
Recorrido 1	Medellín-Buenaventura	Medellín-Cúcuta
Recorrido 2	Buenaventura-Cartagena	Cúcuta-Bogotá
Recorrido 3	Cartagena-Medellín	Bogotá-Medellín
Distancia total	2,347 Km	1,761 Km
Costo fijo	$ 2,400,000	$ 2,400,000
Costo variable	$ 5,867,500	$ 4,402,500
Aprovechamiento (Servicio/ Capacidad)	375/375 = 100 %	375/375 = 100 %

4.2 Discusión

Los resultados obtenidos usando *Grafos*en la solución del CVRP propuesto están limitados a las restricciones del modelamiento definidas anteriormente. El modelo asume un vehículo con capacidad de carga 375 Toneladas/mes y una distancia máxima recorrida de 14,400 Km/mes, que son los datos de capacidades de trabajo promedio de los vehículos analizados en dicho periodo de tiempo.

Teniendo en cuenta estas limitantes de capacidad, se tienen, entonces, los siguientes resultados:

RUTA 1: IdVehículo 14: VEH13

Servicio/Capacidad = Aprovechamiento (%): 375/375 = 100 %

(Demanda) Cliente > Ubicación:

(60) BV1 > BUENAVENTURA

(315) CG1 > CARTAGENA

RUTA 2: IdVehículo 4: VEH4

Servicio/Capacidad = Aprovechamiento (%): 375/375 = 100 %

(Demanda) Cliente > Ubicación:

(345) B1 > BOGOTA

(30) CU1 > CUCUTA

Esto es, la ruta 1 puede cubrirse con un vehículo con aprovechamiento de capacidad del 100% (375 Ton/mes) y un recorrido de 2,347 Km/mes. De igual manera, la ruta 2 es cubierta por un vehículo con aprovechamiento de capacidad del 100% (375 Ton/mes) y un recorrido de 1,761 Km/mes.

El resultado consolidado es:

- Distancia total = 4,108 Km
- Costo Variable total = $ 10,270,000
- Costo Fijo total = $ 4,800,000
- Coste Total (CF+CV)= $ 15,070,000

Este resultado consolidado se interpreta operativamente así: se requieren recorrer 4,108 Km/mes con una capacidad de carga de 375 Ton/mes. Trasladando este resultado a la capacidad de los equipos se tiene:

375 Toneladas = 25 Viajes de 1 camión de 15 toneladas.

25 viajes * 4,108 Km = 102,700 Km

Para cubrir este recorrido se necesita:

(102,700Km)/(14,400 Km / Equipo) = **7.13 camiones**

El análisis se hizo en una flotilla de 13 camiones que están dedicados en un 70.25 % a las rutas analizadas, es decir, 9.13 camiones.

Por lo tanto, el ahorro se puede cuantificar así:

9.13 – 7.13 = 2 camiones

Ahorro = 21.9%

Este ahorro en el uso de 2 equipos menos se refleja directamente en una disminución de $4,800,000en los costos fijos y permite que los equipos estén disponibles para laborar en otras rutas.

Este ahorro se obtiene siguiendo literalmente las recomendaciones delos resultados computacionales, aunque se debe analizar, por ejemplo, que hay algunos desplazamientos vacíos innecesarios en los recorridos Cartagena-Medellín y Cúcuta-Medellín, que se podrían aprovechar en otras rutas, lo cual se debe a las restricciones que se establecieron para modelar el problema.

V. CONCLUSIONES

La aplicación desarrollada para la solución del CVRP permitió demostrar la importancia de la utilización de herramientas informáticas para resolver problemas de ruteo con el fin de optimizar costos para la obtención de ventajas competitivas en sistemas de distribución comercial caracterizados por altos costos operacionalese incremento permanente en los fletes.

El establecimiento de las restricciones para el modelamiento de problemas de ruteo de vehículos juega un papel importante en la obtención de una solución óptima en la medida que se pueden obtener resultados operativamente inviables o ineficientes. En estos casos, es preciso complementar la solución teórica con una adecuada planeación de rutas para evitar, por ejemplo, desplazamientos vacíos de los vehículos.

REFERENCIAS

[1] DiestelR., 2000. Graph theory. 2 ed. New York: Springer. 322 P.
[2] Chandran B. y Raghavan S., 2008.Modeling and Solving the Capacitated Vehicle Routing Problem on Trees. En: Golden B.; Raghavan S. y Wasil E. The vehicle routing problem: latest advances and new challenges. New York: Springer. pp. 239 – 274.
[3] Hernández H., 2004. Procedimientos exactos y heurísticos para resolver problemas de rutas con recogida y entrega de mercancía. Tesis doctoral (Doctor en Estadística e Investigación Operativa). La Laguna: Universidad de La Laguna. Facultad de Matemáticas. Departamento de Estadística, Investigación Operativa y Computación.133 P.
[4] GolumbicM., 2004. Algorithmic graph theory and perfect graphs.2 ed. London: Elsevier. 314 P.
[5] Toth P. y Vigo D., 2002. The vehicle routing problem. Philadelphia: SIAM. 367 P.
[6] Rodríguez A.Grafos. [En línea]. URL disponible en: <http://personales.upv.es/arodrigu/grafos/> [Citado en 10 de diciembre de 2010].
[7] Rodríguez A., 2006. Grafos: Herramienta informática para el aprendizaje y resolución de problemas reales de teoría de grafos. En: x congreso de ingeniería de organización (septiembre de 2006). Valencia: Universidad Politécnica de Valencia. pp. 1-8.
[8] ParedesC., 2008. Análisis del software Grafos. Cataluña: Universidad Politécnica de Cataluña. 102 P.
[9] Wang C-H.y Lu, J-Z., 2009. A hybrid genetic algorithm that optimizes capacitated vehicle routing problems. En: Expert Systems with Applications, N° 36, pp. 2921–2936.