UNA VARIANTE DEL METODO DE GENERACION DE COLUMNAS PARA EL PROBLEMA DE RUTEO VEHICULAR CON RESTRICCIONES MULTIPLES, RESULTADOS PRELIMINARES

Luis Contesse B., Mauricio Marambio C.
Departamento de Ingeniería Industrial y de Sistemas
Pontificia Universidad Católica de Chile, Casilla 306, correo 22, Santiago, Chile
fax: (56-2) 5521608; e-mail: lcontess@ing.puc.cl, e-mail: mmarambi@puc.cl

Juan Carlos Muñoz A.
Departamento de Ingeniería de Transporte, Pontificia Universidad Católica de Chile
Casilla 306, código 105, Santiago 22, Chile
fax: (56-2) 5530281; e-mail: jcmunoz@ing.puc.cl

RESUMEN

En este trabajo, se propone un algoritmo que resuelve a optimalidad, al menos teóricamente, el problema de la asignación de rutas de costo mínimo, que se originan y terminan en un cierto depósito de entre un conjunto posible, para una flota heterogénea cuyo tamaño es una consecuencia de esta asignación. La flota debe servir a un conjunto determinado de clientes con demandas conocidas, distribuidos geográficamente, dentro de ciertas ventanas de tiempo que deben ser respetadas. Estas están definidas por el instante más temprano y el más tarde en que puede comenzar el servicio, de tal modo que los vehículos que lleguen demasiado temprano deben esperar. La flota está constituida por vehículos de distintas capacidades límite, existiendo suficientes vehículos de cada tipo.

El problema, modelado como uno de Partición de Conjuntos, es abordado mediante un enfoque de Generación Parcial de Columnas en el contexto del Método del Branch & Bound. En el problema satélite del método resultante, las columnas son generadas mediante un algoritmo de rutas mínimas modificado, sobre una red con posibles ciclos negativos, que considera las restricciones adicionales impuestas por las ventanas de tiempo y los límites de capacidad para los distintos vehículos.

Se entregan resultados preliminares en algunos problemas clásicos de la literatura, en los que se observan promisorios avances y la versatilidad del método.

1. INTRODUCCION

El problema de ruteo vehicular (VRP) consiste en diseñar un programa de rutas de costo mínimo, que comiencen y terminen en un depósito central, para un flota de vehículos que deben servir a clientes distribuidos geográficamente con necesidades conocidas. Las restricciones usuales son que cada cliente debe ser asignado a sólo un vehículo y que la capacidad de los vehículos no puede ser excedida.

El problema de ruteo vehicular con ventanas de tiempo (VRPTW) es una extensión del problema VRP, que adicionalmente considera que cada cliente define un intervalo de tiempo dentro del cual está dispuesto a recibir al vehículo que lo servirá. Así, el servicio al cliente, que puede ser despachar o recoger, debe comenzar en un instante posterior al comienzo de la ventana de tiempo y antes del cierre de la misma. De este modo, el instante en que cada cliente es servido es una variable de decisión adicional. Este trabajo trata el problema con ventanas de tiempo duras (hard), es decir, tales que si un vehículo llega a visitar a un cliente demasiado temprano para los requerimientos del cliente, este simplemente debe esperar para comenzar su servicio. Por el contrario, no está permitido que un vehículo visite a un cliente fuera de la ventana de tiempo.

Si a este problema se agrega la posibilidad de que las rutas puedan originarse en un depósito cualquiera de entre un conjunto dado de depósitos, se está entonces ante un problema conocido en la literatura como "Multiple Depot Vehicle Routing Problem with Time Windows" (MD-VRPTW). En este caso, se supone que cada vehículo comienza y termina su ruta en el mismo depósito. Igualmente se supone en este caso, que cada vehículo realiza sólo una vuelta (ruta) en cada programa.

En este trabajo se propone un algoritmo exacto para resolver este último problema en el caso que se disponga de una flota de vehículos heterogénea en capacidad. El tamaño de la flota define una variable de decisión implícita cuyo valor es determinado simultáneamente con el programa óptimo de rutas.

El algoritmo reposa sobre el método de generación de columnas y está vinculado a la aproximación desarrollada originalmente por Appelgren (1969) y, más tarde, por Desrochers et al. (1992), entre otros.

La principal diferencia entre nuestro enfoque y el de Desrochers et al. (1992), es la forma en que se explora el árbol de Branch & Bound. En nuestro algoritmo combinamos una generación parcial de rutas con una asignación óptima de las rutas generadas.

De este modo, luego de cada asignación óptima parcial, se obtiene una asignación factible para el problema original. Más aún, estas asignaciones sucesivas generan una sucesión de soluciones factibles cuyo costo converge monótonamente hacia el costo óptimo en un número finito de iteraciones.

El algoritmo propuesto tiene la importante ventaja de poder extenderse para resolver problemas más complejos o sofisticados de un modo simple y económico.

2. EL MODELO MATEMATICO

Sea N el conjunto de los clientes y D el conjunto de los depósitos. Separaremos cada depósito $d \in D$ en dos nodos; uno de inicio s_d y otro de término t_d. Sea S el conjunto de los nodos s_d, $d \in D$, y T el conjunto de nodos t_d, $d \in D$. Entonces, el grafo G (V,A), donde $V := N \cup S \cup T$ y $A = (S \times N) \cup (N \times N) \cup (N \times T) \setminus \mathrm{Diag}(N \times N)$, define la red.

Asociado a cada arco $(i, j) \in A$, existe un costo c_{ij} y un tiempo de duración t_{ij}. Asumiremos que el tiempo y el costo de servicio para el cliente i están incluidos, respectivamente, en la duración y el costo de cada arco (i, j) que emerge de i.

El costo fijo por utilizar cada vehículo se carga a los arcos que salen de cada bodega, esto es, a los arcos del conjunto S x N.

A cada vehículo se asigna un determinado índice k, que identifica completamente su capacidad Q_k como también la subred en que el vehículo podrá moverse G (V_k, A_k), con su respectivo nodo de comienzo s_k y de término t_k, los clientes que puede visitar N_k, con sus respectivas demandas y ventanas de tiempo, y los arcos por los que le está permitido moverse con sus respectivos tiempos y costos operacionales. Sea K el conjunto de todos los vehículos disponibles.

Para representar el modelo, se requerirán dos tipos de variables: las variables enteras x_{ij}^k, $(i, j) \in A_k$, $k \in K$, que tomarán el valor 1 si el arco $(i, j) \in A_k$ es utilizado por el vehículo k y 0 si no, y las variables continuas t_i, $i \in N$ que definen el instante de inicio del servicio al cliente i.

Así, el conjunto óptimo de rutas que satisfacen todas las restricciones consideradas, es solución al siguiente problema de minimización:

$$Min \sum_{k \in K} \sum_{(i,j) \in A_k} c_{ij}^k x_{ij}^k \tag{1}$$

$$\sum_{j \in N_k \cup \{t_k\}} x_{ij}^k = \sum_{j \in N_k \cup \{s_k\}} x_{ji}^k \qquad \forall i \in N_k, k \in K \tag{2}$$

$$\sum_{k \in K} \sum_{j \in N_k \cup \{t_k\}} x_{ij}^k = 1 \qquad \forall i \in N_k \tag{3}$$

$$x_{ij}^k (t_i + t_{ij}^k - t_j) \le 0 \qquad \forall (i,j) \in A_k, k \in K \tag{4}$$

$$a_j \le t_j \le b_j \qquad \forall j \in N_k \tag{5}$$

$$\sum_{(i,j) \in A_k} x_{ij}^k q_i^k \le Q_k \qquad \forall k \in K \tag{6}$$

$$x_{ij}^k \quad binary \qquad \forall (i,j) \in A_k, k \in K \tag{7}$$

Las restricciones (2) establecen continuidad en los flujos y las restricciones (3), que cada cliente sea servido exactamente una vez. Las restricciones (4) y (5) establecen la compatibilidad que debe existir entre las rutas y los instantes de inicio del servicio para cada cliente. Las restricciones (6) imponen que la capacidad Q_k, $k \in K$, de los distintos vehículos no sea sobrepasada.

3. MODELO DE PARTICION DE CONJUNTOS Y ENFOQUE DE GENERACION DE COLUMNAS

Este modelo equivalente considera como variables de decisión las rutas a utilizar por cada vehículo que entra en juego en vez de los arcos a utilizar por los mismos. Definiremos como ruta factible a toda ruta r_k que realiza un cierto vehículo k, que se origina en el nodo s_k, termina en el nodo t_k y que en el intertanto visita a un determinado sub-conjunto de clientes satisfaciendo las distintas restricciones de red, ventanas de tiempo y capacidad asociadas a dicho vehículo. Para efectos de esta modelación, cada ruta factible quedará completamente definida por el conjunto de clientes que visita y por el costo en que incurre el vehículo correspondiente para visitar estos clientes. Introduciremos la siguiente notación:

Ω :conjunto de todas las rutas factibles.
y_r :variable binaria cuyo valor es 1 si la ruta r es parte de la solución óptima y 0, en caso contrario.
δ_{ri} :parámetro que toma el valor 1 si la ruta r visita al nodo i y 0 en caso contrario.
c_r :costo asociado a la ruta r, que suponemos de valor entero, para todo r.

Nuestro objetivo consiste en encontrar un conjunto de rutas factibles de mínimo costo total, de modo que todos los clientes sean visitados exactamente por sólo un vehículo. De acuerdo a esto, formulamos el siguiente modelo de partición de conjuntos:

$$Min \sum_{r \in \Omega} c_r y_r \tag{8}$$

$$\sum_{r \in \Omega} \delta_{ri} y_r = 1 \qquad \forall i \in N \tag{9}$$

$$y_r \text{ binary} \qquad \forall r \in \Omega \tag{10}$$

No está de más decir, que dos rutas que visitan exactamente los mismos clientes, pero en distinto orden, no tienen necesariamente el mismo costo.

Al igual que en Desrosiers(1984), para abordar este problema, dentro del contexto del Método del Branch & Bound, haremos uso del clásico método de Generación de Columnas. En cada nodo del árbol, se resuelve el siguiente problema lineal reducido:

$$Min \sum_{r \in \Omega} c_r y_r \tag{11}$$

$$\sum_{r \in \Omega} \delta_{ri} y_r = 1 \qquad \forall i \in N \tag{12}$$

$$0 \le y_r \le 1 \qquad \forall r \in \Omega \tag{13}$$

$$y_r = 0 \qquad \forall r \in \Omega_0 \subset \Omega \tag{14}$$

$$y_r = 1 \qquad \forall r \in \Omega_1 \subset \Omega \tag{15}$$

en que las variables y_r, $r \in \Omega_0$ [resp. $r \in \Omega_1$] se fijan en 0 [resp. en 1]. A este problema le denominaremos el problema maestro reducido asociado a este nodo.

Para cualquier solución básica factible para el problema maestro, puede demostrarse que una nueva columna (ruta) de costo reducido mínimo se obtiene como solución óptima del problema de rutas mínimas para algún vehículo $k \in K$:

$$Min \sum_{(i,j)\in A_k} (c_{ij}^k - \sigma_i)\, x_{ij}^k \tag{16}$$

$$\sum_{j\in N_k \cup T_k} x_{ij}^k = \sum_{j \in N_k \cup S_k} x_{ji}^k \qquad \forall i \in N_k \tag{17}$$

$$\sum_{j\in N_k} x_{sj}^k = \sum_{j\in N_k} x_{jt}^k = 1 \tag{18}$$

$$x_{ij}^k > 0 \Rightarrow t_i + t_{ij}^k \le t_j \qquad \forall (i,j) \in A_k \tag{19}$$

$$x_{ij}^k > 0 \Rightarrow a_i \le t_i \le b_i \qquad \forall i \in N_k \tag{20}$$

$$\sum_{(i,j)\in A_k} x_{ij}^k q_i^k \le Q_k \tag{21}$$

$$x_{ij}^k \text{ binary} \qquad \forall (i,j) \in A \tag{22}$$

en que σ_i corresponde al multiplicador dual asociado al problema maestro relajado. Para un k dado, el problema es equivalente a encontrar una ruta de costo mínimo factible sobre la red $G(V_k,A_k)$, que se origina en el nodo s_k y termina en el nodo t_k, y donde los costos de los arcos (i,j) son de valor $(c_{ij} - \sigma_i)$. A estos subproblemas les denominaremos problemas satélites, y habrá tantos como vehículos k, k en K. Aunque algunos de estos problemas serán idénticos (por ejemplo, en caso de vehículos de idéntica capacidad que salen de un mismo depósito), éstos son considerados distintos para la aplicación del Método.

Es claro que si la solución factible básica vigente no es óptima, existirá entonces al menos una ruta con costo reducido negativo que podrá generarse al resolver alguno de estos problemas satélites. Afortunadamente para que el método converja, sólo es suficiente generar una ruta de costo negativo para alguno de estos problemas satélites y, no necesariamente, la más negativa.

En los subproblemas habrá arcos con costo negativo $(c_{ij} < \sigma_i)$ y, muy probablemente entonces, ciclos de costo negativo. De cualquier modo, las rutas factibles y, en particular también las que formen la solución óptima, no admiten ciclos en ellas, debido a la restricción (19) para los instantes de servicio a los distintos clientes. Estos problemas de costo mínimo no son fáciles de resolver debido a la existencia de ciclos de costo negativo. Para resolverlos, se usa una extensión del método propuesto por Desrosiers (1984) que incorpora las restricciones de capacidad en los vehículos. Es evidente que, entre más severas sean las restricciones de ventanas de tiempo y de capacidad en las rutas, menor es la probabilidad de generar rutas con ciclos de costo negativo en la resolución de los problemas satélites. El no tener que resolver los problemas satélites a optimalidad, también va en esta misma dirección.

El método descrito en la literatura para resolver el problema maestro (ver Appelgren (1969), Desrosiers(1984), Desrochers (1992) y Desrosiers (1995)), consiste en generar columnas (rutas) de costo reducido negativo a través de la resolución sucesiva de un número finito de problemas satélites.

Cuando no se pueda generar más rutas de costo negativo en ninguno de los subproblemas, el método habrá alcanzado una solución óptima relajada. Entonces, en caso de que la solución no sea una solución entera, deberá ramificarse en el árbol de B&B. La estrategia usual de ramificación encontrada en la literatura se basa en las siguientes consideraciones:

i) El número total de rutas activas debe ser un número entero
ii) El costo total debe ser entero

Cuando ambas condiciones han sido satisfechas y, no obstante, la solución vigente aún no es entera, entonces una tercera condición debe ser impuesta. Esta es, impedir que una determinada ruta sea parte de la solución óptima o, alternativamente, imponer que sea parte de ella. En términos del modelo, significa fijar una determinada variable y_r en 0 ó en 1, respectivamente. Algorítmicamente, este procedimiento no es fácil de ejecutar. De hecho, citando a Desrochers (1992), siempre es posible fijar una variable (fraccional) y_r en el valor 1. Esta información es fácilmente transferida al subproblema, simplemente eliminando de los problemas satélites, todos los nodos que la ruta r visita. Por el contrario, resulta prácticamente imposible fijar directamente la variable y_r en el valor 0.

En la figura 1, se describe esquemáticamente el proceso iterativo clásico.

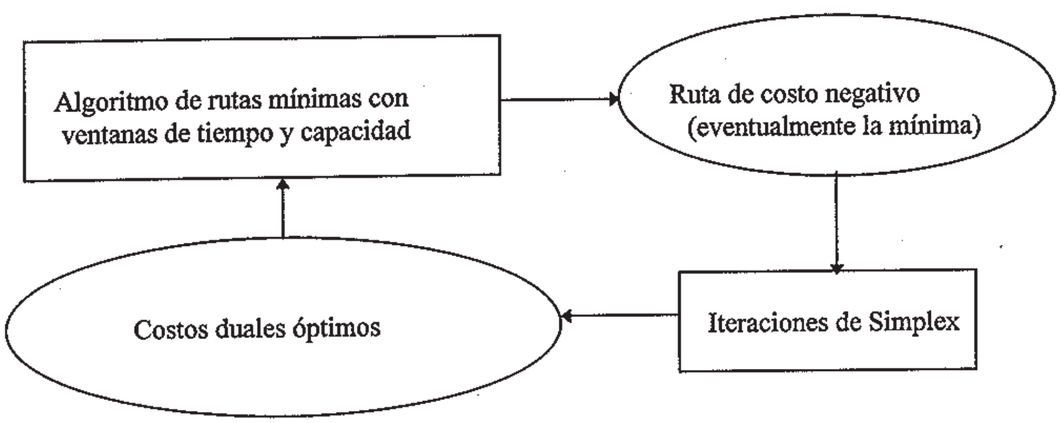

Figura 1

4. EL METODO PROPUESTO

El método propuesto en este trabajo, aborda el problema cambiando la forma en que el problema maestro y el problema satélite interactúan. En adelante supondremos, sin pérdida de generalidad, que existe un solo tipo de subproblema, con un único depósito, un único tipo de vehículo y una única red.

El procedimiento propuesto consiste en, dado un conjunto de rutas factibles, realizar una asignación óptima de éstas de modo de servir a todos los clientes. Una solución entera (eventualmente óptima) para este problema de Partición de Conjuntos Reducido, se obtiene por medio del Método de B&B, usualmente combinado con alguna heurística especializada. A partir de esta solución óptima entera que es obviamente factible para el problema maestro original se construye una solución básica factible entera de igual valor y se realiza el cálculo del vector de costos duales correspondiente. Con este vector, se resuelve el problema satélite para obtener nuevas columnas (rutas) de costo reducido negativo. Estas nuevas rutas son agregadas al conjunto reducido actual de rutas factibles y así se define un nuevo problema de Partición de Conjuntos Reducido. Resolviendo este problema por medio del B&B (eventualmente a optimalidad), se obtiene una nueva solución entera que es mejor o igual que la anterior y así sucesivamente. Procediendo de esta forma, se obtiene una secuencia de problemas cuyo valor óptimo es monótonamente no creciente. Este proceso iterativo sólo se detiene cuando no es posible generar nuevas rutas de costo negativo en el problema satélite. En esta última situación, podemos afirmar que la solución entera es óptima para el problema maestro original. En la Figura 2, se muestra el proceso iterativo clásico.

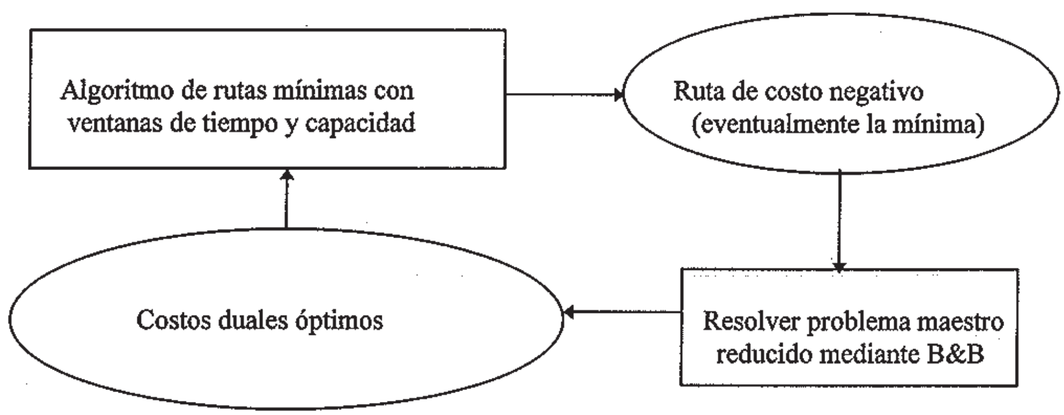

Figura 2

4.1. Análisis de convergencia finita

Si la solución entera calculada, óptima o no, para el problema reducido actual no es una solución óptima para el problema original, se agregarán entonces nuevas columnas de costo reducido negativo y se definirá así un nuevo problema maestro reducido. Como existe un número finito de rutas factibles, habrá una secuencia finita de problemas reducidos. Cada vez que se encuentran nuevas rutas de costo reducido negativo, al menos una de estas rutas es agregada al problema reducido. El valor óptimo entero del nuevo problema reducido será menor o igual que el actual.

Si la solución entera actual no es una solución entera óptima para el problema maestro original, entonces después de un número finito de redefiniciones del problema reducido el procedimiento debe necesariamente llegar a un problema maestro reducido cuya solución entera tenga un valor óptimo entero estrictamente menor. Así, como de este modo sólo es posible generar una secuencia finita de problemas reducidos que termina, en el peor de los casos, con la generación del problema maestro original, al cabo de un número finito de iteraciones se llegará necesariamente a la generación de un problema reducido cuya solución óptima entera es también solución óptima entera al problema original. Esta solución entera, sólo será identificada como solución óptima del problema original cuando no puedan encontrarse rutas de costo reducido negativo en el problema satélite.

Debido a la monotonía de los valores óptimos de la secuencia de problemas reducidos, el costo de la solución entera vigente entrega la menor cota superior disponible para el valor óptimo entero del problema original. Este costo define claramente un valor de corte para la aplicación del método de B&B al siguiente problema reducido. Además, para garantizar la convergencia finita del método, no es necesario encontrar una solución óptima entera para el problema reducido. En realidad, podrá detenerse el B&B una vez que se haya encontrado una nueva solución entera factible con costo estrictamente menor que la actual. Por otro lado, para obtener una cota inferior del valor óptimo entero del problema original, se puede resolver el problema maestro original, relajando las restricciones de integralidad, mediante el método clásico de Generación de Columnas.

Para que el B&B pueda encontrar una solución entera mejor que la anterior, se debe estar seguro de que el nuevo problema reducido contiene al menos una ruta de costo reducido negativo adicional. Por otro lado, con el fin de no aumentar innecesariamente el tamaño del problema reducido, si dos rutas visitan exactamente a los mismos clientes, pero en distinto orden, bastará conservar la de menor costo. Más generalmente, para evitar acumular demasiadas columnas en el problema reducido, se requiere eliminar algunas de ellas, de entre las que no son parte de la solución óptima entera. Sin embargo, para garantizar la convergencia finita del método en este caso, la eliminación de rutas sólo podrá realizarse después de obtener una solución entera factible estrictamente mejor que la actual. Uno de los criterios propuestos para hacer esto, es eliminar rutas de entre aquéllas que tengan más altos costos reducidos. Esto no significa que las rutas eliminadas no puedan ser reconsideradas más adelante en el curso del algoritmo. Por esto se propone almacenar estas rutas en algún lugar y revisar si en algún momento vuelven a tener un costo reducido negativo.

Evidentemente, para que este método sea competitivo con otros métodos alternativos, es importante que el tamaño del problema reducido se mantenga razonablemente pequeño. Este aspecto no está garantizado con la forma en que se eliminan rutas. De hecho, si la solución vigente está ubicada en un lugar poco estratégico del árbol de B&B, se podrá aumentar considerablemente el tamaño del problema reducido, antes de conseguir una solución entera mejor. Deberemos por tanto pensar en nuevos criterios para eliminar rutas, sin destruir la convergencia del algoritmo. Este criterio debería evitar en lo posible resolver dos veces un mismo problema maestro reducido. El primer criterio propuesto es encontrar la solución óptima al problema reducido relajado. Si el valor óptimo asociado a esta solución es menor que el del problema precedente entonces, si eliminamos rutas cuidando de conservar tanto las rutas que conforman la solución óptima entera como la solución óptima relajada al problema maestro reducido actual, se estará seguro de que el problema previo no podrá ser resuelto nuevamente en el curso del algoritmo. Este criterio es siempre susceptible de ser aplicado si el valor óptimo para el problema reducido relajado actual es estrictamente mayor que el valor óptimo del problema relajado maestro original.

Alternativamente, en vez de resolver el problema reducido relajado actual, se podría resolver un problema reducido relajado aún más reducido, resultante de eliminar del primero algunas o incluso todas las rutas correspondientes a aquellas variables que fueron forzadas a tomar el valor cero en el B&B. El valor óptimo de este último problema reducido es obviamente mayor o igual al del problema relajado actual, de modo que el criterio explicado en el párrafo precedente, que es claramente válido para este problema también, se hace particularmente atractivo de aplicar.

En el caso de que no sea posible definir un nuevo problema reducido con un valor óptimo estrictamente menor que el anterior, se deberá recurrir temporalmente a técnicas de ramificación que logren evitar que algunas rutas específicas sean generadas por el problema satélite. Ver Desrochers [1992], Desrosiers [1995].

Aplicar este método al problema con múltiples depósitos es bastante directo. En vez de tener un único problema satélite que provee rutas factibles al problema maestro, habrá tantos de éstos como depósitos hay en el problema. La Figura 3 ilustra el procedimiento para dos depósitos. Para seleccionar el orden de ejecución de los problemas satélites existirán variados criterios. Inclusive, ambos problemas satélites pueden ser abordados simultáneamente.

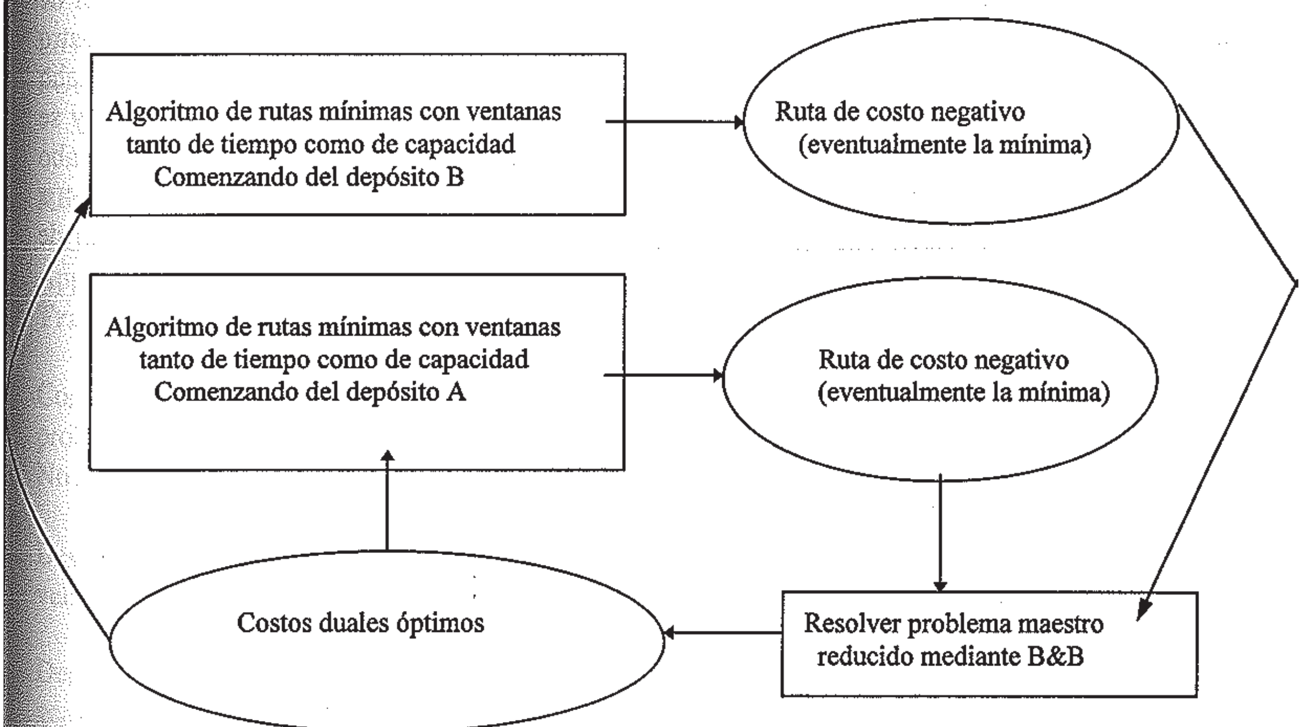

Figura 3

Adicionalmente si se dispone de vehículos de distintas capacidades, se podrá ocupar exactamente el mismo método. Por ejemplo, si existen tres tipos de vehículos diferentes, entonces se contará con seis problemas satélites distintos (uno por cada tipo de vehículo y de depósito).

Notar que si el problema dispone sólo de un tipo de vehículo, pero de muchos depósitos, y si está permitido que un vehículo comience una ruta en un depósito, pero la termine en otro, entonces si se agrega un nodo artificial que contacte a todos los depósitos, del que salgan y al que vuelvan los distintos vehículos, el problema puede modelarse con sólo un único problema satélite.

5. RESULTADOS DEL ALGORITMO

Durante el transcurso de la implementación de variadas versiones del sistema, se ha probado que éste presenta buenos resultados tanto en las soluciones entregadas como en los tiempos en alcanzarlas para cierto tipo de problemas. Se ha logrado determinar que la eficiencia del sistema con problemas cuyos clientes poseen ventanas de tiempo estrechas es bueno, lográndose soluciones por sobre un 2% con respecto a las mejores soluciones publicadas, obteniéndose en tiempos aceptables.

La eficiencia del sistema empeora drásticamente con problemas cuyos clientes poseen ventanas de tiempo anchas. Esto se debe principalmente a la lentitud del algoritmo de D'Esopo modificado para este problema. Es muy posible que ante ventanas de tiempo anchas una heurística alternativa podría entregar tiempos menores.

Los mejores resultados obtenidos se han logrado utilizando una idea denominada "Partición de los problemas", ésta consiste en dividir el espacio en que están ubicados los clientes en cuatro regiones simétricas y ejecutar el algoritmo en pares adyacentes de esas regiones. Una vez hecho esto para las cuatro posibilidades se reúnen todas las columnas generadas en cada proceso y se ejecuta el algoritmo en base a la región completa. La ejecución de los problemas se llevó a cabo en un equipo Pentium de 133 Mhz con 32 Mb en Ram. Los problemas escogidos corresponden a los presentados en Solomon (1987) que consisten en 100 clientes con ventanas de tiempo de distinta magnitud para cada problema.

Tabla 1
Capacidad camión = 200; 1 bodega

		Algoritmo			Literatura		
Problema	Clientes	Solución	N° rutas	Tpo (seg.)	Solución	N° rutas	Tpo (seg.)
R101	25	618,32	8	6	617,1	8	5,8
	50	1046,7	12	25	1035,2	12	66,7
	100	1642,87	20	472	1607.7	18	1064,2
R102	25	548,1	7	14	547,1	7	20,3
	50	911,44	11	83	904,6	11	67,8
	100	1473,98	18	1387	1434	17	756,9
R109	100	1219.67	14	26803	1205	12	

Tabla 2
Capacidad camión = 200; 2 bodega

Problema	Clientes	Solución	N° rutas	Tpo. (seg.)
R101	20	419,67	8	5
	50	895	14	27
	100	1392,78	21	173

Tabla 3
Capacidad camión = 200; 4 bodega

Problema	Clientes	Solución	N° rutas	Tpo. (seg.)
R101	20	357.97	9	5
	50	802.76	16	41
	100	1212.06	25	209

En la literatura no se ha reportado aún problemas de más de una bodega, por lo que sólo pueden contrastarse los resultados obtenidos para los casos de una bodega.

AGRADECIMIENTOS

Los autores expresan su agradecimiento al proyecto FONDECYT 1951224 y al proyecto FONDEF 55 por su apoyo en este trabajo.

REFERENCIAS

Appelgren, L.H. (1969) A column generation algorithm for a ship scheduling problem. **Transportation Science 5**, 64-78.

Contesse, L. y J.C. Muñoz (1996) An exact algorithm for the multiconstrained vehicle routing problem. **2nd ALIO/EURO Workshop on Practical Combinatorial Optimization.** 11-15 November, Valparaíso, Chile.

Desrochers, M., J. Lenstra and M.M. Solomon (1992) A new optimization algorithm for the vehicle routing problem with time windows. **Operations Research 40**, 342-354.

Desrosiers, J., F. Soumis y M. Desrochers (1984) Routing with time windows by column generation. **Networks 14**, 545-565.