

ANALISIS DE LA PERFORMANCE DE HEURISTICAS EN PROBLEMAS DE RUTEO

Elsa S. de Wladislawowsky
Profesora Titular Investigación Operativa II

Luis Fornero
Profesor Titular Area Computación

I. PROBLEMAS DE RUTEO

Se los conoce como "vehicle routing problem (VRP)", "vehicle scheduling", "truck dispatching" o "delivery problem".

Son problemas de distribución donde vehículos con base en un depósito central deben abastecer, en un período dado, a un conjunto geográficamente disperso de clientes con demanda conocida de modo de minimizar algún objetivo (costo, combustible, horas-hombre, tamaño de la flota, etc.).

Estos problemas se aplican a:

- Empresas de mensajería o correos privados
- Recolección de basura
- Clearing bancario
- Transporte de personal o escolares

$$\sum_k Y_{i,k} = \begin{cases} k & i = 0 \\ 1 & i = 1, \dots, n \end{cases} \quad (2)$$

$$Y_{i,k} = 0 \text{ ó } 1 \quad \begin{matrix} i = 0, \dots, n \\ k = 1, \dots, k \end{matrix}$$

$$\sum_i X_{i,j,k} = Y_{j,k} \quad j = 0, \dots, n \quad (3)$$

$$\sum_j X_{i,j,k} = Y_{i,k} \quad i = 0, \dots, n \quad (4)$$

$$\sum_{(i,j) \in S \times S} X_{i,j,k} \leq |S| - 1 \quad S \subseteq \{1, \dots, n\} \quad (5)$$

$$X_{i,j,k} = 0 \text{ ó } 1 \quad \begin{matrix} i = 0, \dots, n \\ j = 0, \dots, n \\ k = 1, \dots, k \end{matrix}$$

La restricción (5) es para evitar los sub-tours o ciclos intermedios y ha sido formulada de diversas maneras, incluso por los mismos autores, por lo cual hemos elegido la que nos pareció más sencilla.

K = número de vehículos

$i = 0$ es el depósito

$i = 1, \dots, n$ clientes

B_k = capacidad del vehículo k

A_i = tamaño del pedido del cliente i .

$C_{i,j}$ = costo directo de viajar del cliente i al cliente j .

Los problemas pequeños pueden ser resueltos utilizando esta formulación, branch and bound, branch and cut, programación dinámica, etc. Dada la estrecha relación que tiene este problema con el del viajante de

comercio, las dificultades para obtener una solución óptima a medida que aumenta su tamaño son las mismas.

Es bueno recordar que el problema del viajante de comercio es NP-completo. También es NP-completo el problema de obtener una solución aproximada para la cual se conozca el ϵ .

Por estas razones es muy importante contar con métodos aproximados eficientes.

III. HEURISTICAS QUE PERMITEN OBTENER OPTIMOS LOCALES EN PROBLEMAS DE RUTEO

A continuación damos un listado de las heurísticas desarrolladas hasta el momento para resolver problemas de ruteo.

1. "Del vecino más cercano" (nearest neighbor). Depende del cliente inicial que se seleccione. Luego se visita el cliente más cercano al anterior y así sucesivamente hasta que se satura el cuello de botella que puede ser la capacidad del vehículo, el tiempo de recorrido, etc. Esto se hace para cada vehículo.
2. "De inserción". Parten del tour trivial (del depósito al depósito) y luego van agregando clientes según distintos criterios hasta saturar el cuello de botella, oportunidad en que se inicia un nuevo tour.
 - "Más cercana"(nearest): incluye al cliente más cercano a alguno del tour ya elegido.
 - "Más lejana"o "Más cara"(farthest): se incluye el cliente más lejano al tour ya elegido.
 - "Más barata"(cheapest): incluye el cliente que minimiza la distancia que se agrega al tour. Es decir, entre cada par de clientes (u, v) calcula el costo de insertar un nuevo cliente "x": costo de $(u, x) +$ costo de $(x, v) -$ costo de (u, v) . De todos estos costos (calculados para cada par de clientes contiguos) elige el cliente "x" que requiere el menor costo de inserción

3. "De la cápsula convexa" (convex hull): encuentra la cápsula convexa como tour parcial (si no hay un cuello de botella que lo invalide, en cuyo caso no puede usarse esta heurística) y agrega los restantes utilizando la "cheapest".
4. "Clarke and wright savings"(1964): es la más antigua y la más conocida. Consiste en crear un tour separado para cada cliente (del depósito al domicilio del cliente). De ese modo se inicia con la peor solución posible. A partir de ese momento se van uniendo tours aprovechando el ahorro de volver al depósito cada vez. El ahorro del cliente "i" al "j" sin pasar por el depósito "d" cada vez es:

$$S_{i,j} = C_{i,d} - C_{i,j} + C_{j,d}$$

Luego se ordenan los ahorros en forma decreciente y se van haciendo las inserciones de acuerdo al orden establecido respetando factibilidad.

5. "Clustering sweep" Gillet y Miller 1974. Es más intuitiva que las anteriores y de tipo geométrico. Se ubican los clientes en el plano y hace pasar un arco imaginario del depósito a uno de los clientes. Luego "gira" el arco abarcando la mayor cantidad posible de clientes hasta agotar la capacidad de cada vehículo y así sucesivamente.
6. "K-óptima" (Lin 1965; Lin y Kernighan 1973). Necesita una solución inicial. Parece dar buenos resultados tomando como solución inicial alguna heurística "greedy" (voraz o glotón). Se aplica sólo al caso simétrico. Para el caso 2-óptimo selecciona dos aristas no adyacentes (x, y) y (u, v) y las reconecta obteniendo (x, u) y (y, v) cambiando el sentido cuando corresponde y obteniendo un ahorro de:

$$c(x, u) + c(y, v) - c(x, y) - c(u, v)$$

en cada iteración selecciona las dos aristas que producen el mayor ahorro. Para el caso 3-óptimo elige 3 aristas no adyacentes las que pueden ser reconectadas de ocho maneras distintas y así sucesivamente. Es evidente que a medida que k aumenta, los cálculos son cada vez más engorrosos.

Como puede verse, la mayoría de los métodos presentados han sido desarrollados para resolver el problema del viajante de comercio. Esto es

razonable ya que cada vehículo enfrenta este problema, sin embargo hay un detalle de la mayor importancia que excede al caso del viajante de comercio y es la selección de los clientes que debe visitar cada vehículo.

IV. HEURISTICAS SELECCIONADAS

De acuerdo a lo expuesto anteriormente hemos decidido trabajar con heurísticas greedy dado que son muy rápidas a pesar de que no son muy eficientes para resolver el problema del viajante de comercio.

Sin embargo, pueden usarse como soluciones iniciales en algoritmos "de mejora" como son los k-óptimos. Además, de los diversos métodos que se han desarrollado, los que obtienen mejores resultados son muy engorrosos de implementar y requieren mucho tiempo de procesamiento. Por tal motivo es importante conocer la performance de las distintas heurísticas. Nos hubiera gustado trabajar con más heurísticas, pero distintas limitaciones nos llevaron a decidimos por las siguientes:

1. De inserción: Se inicia con el tour trivial para cada vehículo. Luego le asigna clientes al primero hasta saturar el cuello de botella, oportunidad en que pasa al segundo y así sucesivamente. Termina cuando el último cliente es asignado.

- De inserción más cercana: En cada iteración incluye al cliente más cercano a alguno del "tour" ya elegido.
- De inserción más lejana o más cara: Incluye al cliente más lejano al "tour" ya elegido.
- De inserción más barata (cheapest): Incluye al cliente que minimiza la distancia que se agrega al tour. Es decir, entre cada par de clientes (u, v) calcula el costo de insertar un nuevo cliente "x" haciendo:

$$\text{costo de } (u, x) + \text{costo de } (x, v) - \text{costo de } (u, v).$$

De todos estos costos (calculados para cada par de clientes contiguos) elige el cliente "x" que da el menor costo de inserción.

2. Del vecino más cercano: Su eficiencia depende del cliente inicial que se seleccione, luego se visita el más cercano al anterior y así sucesivamente hasta que se satura el cuello de botella que puede ser la capacidad del vehículo, el tiempo de recorrido, etc. Por supuesto debe tenerse en cuenta el tiempo necesario para volver del último cliente al depósito así como para ir del depósito al primer cliente. Una vez completado el primer vehículo se pasa al segundo y así sucesivamente hasta que no queden clientes sin asignar.

3. Original: Esta heurística fue diseñada por nosotros para otro trabajo donde aprovechamos las características muy particulares de un problema dado. Sin embargo, pensamos incluirla en el proyecto de investigación a efectos de verificar su comportamiento respecto de las más generales. Consiste, sencillamente, en la del vecino más cercano pero iniciando cada tour con el cliente más lejano al depósito entre todos los no asignados. La lógica de este criterio (que resultó eficiente como se verá más adelante) es que parece razonable asignar a cada vehículo el cliente más alejado del depósito y luego asignar los que puede visitar "de camino" que son los que menos aumentan la distancia recorrida.

V. CARACTERISTICAS DEL PROGRAMA

Hemos trabajado en Quick basic con una matriz que admite hasta 185 clientes donde se volcaron datos de distancias de diversos lugares de la provincia de Mendoza y hemos elegido al azar un domicilio de depósito situado en el lugar que nos pareció más razonable.

Hemos generado al azar subgrafos asociados a distintas submatrices para verificar la performance de las heurísticas en distintas condiciones.

También hemos hecho pruebas bajo el supuesto de que los clientes imponen condiciones de horario, es decir que hay lo que se llama "horario de ventanilla" (time window).

Para finalizar comentamos los resultados obtenidos bajo los distintos supuestos.

Al ingresar al programa se accede a un menú que tiene las siguientes opciones:

1. MODIFICAR

1.1 Lugares: Se refiere a domicilios de clientes. En pantalla aparecen los números de clientes y se indican las siguientes opciones:

- * F2 “Almacenar”
- * F3 “Activar”
- * F9 “Distancias”: con el cursor puesto sobre el número de cliente cuyas distancias respecto de los demás se quiere ingresar o modificar, se oprime F9 y se accede a un sencillo cuadro donde se colocan las distancias del cliente elegido a todos los demás.
- * F10 “Servicios”: con el cursor puesto sobre el número de cliente se oprime esta tecla y se accede a un cuadro encabezado por los siguientes ítems:
 - * “Tipo”: Se refiere a los que generan distintos tiempos de espera en el domicilio del cliente.
 - * “Hora 1”: es el límite inferior del horario de ventanilla.
 - * “Hora 2”: es el límite superior del horario de ventanilla.
 - “Meses”: permite incorporar un plan anual de visitas.
 - “Días del mes”: permite incorporar un plan mensual de visitas.
 - “Probabilidad”: es para generar los subgrafos para las distintas puebas. Dentro de esta opción están a su vez las siguientes:
 - *F2 “save and exit”
 - *F9 “Insert”

***F10 "Delete"**

1.2 Verificar distancias: esta opción permite controlar si se cumple la desigualdad triangular.

2. SERVICIOS

2.1 Servicios para un día: Genera al azar un subconjunto de clientes. Si no se desea trabajar con un subconjunto de clientes sino con el total, se debe poner probabilidad 1 en la opción "probabilidad" descrita anteriormente. De la lista resultante pueden agregarse o eliminarse clientes con las siguientes opciones:

- * F2 "save and exist"
- * F9 "insert"
- * F10 "Delete"

2.2 Selección de heurística: al oprimir esta opción aparece en pantalla el listado de heurísticas a seleccionar:

- * "Original"
- * "Vecino más cercano"
- * "Inserción más cercana"
- * "Inserción más lejana"
- * "Inserción más barata"

2.3 Cálculo de recorridos: al seleccionar esta opción, el programa calcula el recorrido según la heurística elegida indicando paso a paso el cliente que va incorporando. Al final da un listado ordenado de clientes con el horario de llegada a cada uno de ellos y la distancia total recorrida así como el tiempo de cómputo.

2.4 Visualizar recorridos: Da un listado de clientes con indicación del recorrido asignado.

2.5 Imprimir recorridos

VI. EXPERIMENTOS REALIZADOS

Como se dijo anteriormente se han generado subconjuntos de clientes al azar, bajo distintas condiciones de horarios de entrega de mercadería y tiempos de demora en carga o descarga así como horarios de ventanilla, para conocer el comportamiento de las distintas heurísticas.

Los subgrafos tienen entre 70 y 80 nodos y se generan a partir del grafo de 185 nodos con datos de distancias de la provincia de Mendoza.

Los sub conjuntos de clientes incluyen siempre Gran Mendoza y agregan, además, clientes de varios departamentos.

Se han considerado los siguientes casos:

- * Horario continuo de entrega de 8 a 21 hs. con paradas de 10 minutos en cada cliente.
- * Horario discontinuo de 8 a 13 y 16.30 a 21 hs. con 10 minutos de espera en cada cliente.
- * Horario continuo de 8 a 18 hs. con 30 minutos de espera en cada cliente.
- * Horario continuo de 8 a 21 hs. con 30 minutos de espera en cada cliente.
- * Horario continuo de 8 a 20 hs. con 30 minutos de espera en cada cliente.

VII. RESULTADOS OBTENIDOS

Tiempo de computación

En cuanto a tiempo de computación las heurísticas "original" y "del vecino más cercano" por sus características requirieron tiempos muy similares entre ellas y sensiblemente menores que las otras tres llamadas "de inserción" que son más elaboradas.

En la mayoría de los casos el menor tiempo fue requerido por la heurística "original" siguiéndole la del "vecino más cercano" por diferencias de hasta el 6%.

La tercera en orden creciente es la de "inserción más cercana" que da un gran salto respecto de las anteriores siendo entre 40 y 50% superior a la del "vecino más cercano".

A continuación se sitúa la heurística "de inserción más cara" con tiempos hasta 5% superiores a su inmediata predecesora.

La heurística que requiere mayor tiempo de cómputo es la de "inserción más barata" que tarda entre 30 y 40% más que la anterior, lo que es razonable dado que efectúa más cálculos y comparaciones.

Distancia recorrida

Para poder juzgar mejor los resultados obtenidos en cuanto a distancias recorridas, se presenta para cada uno de los cinco casos descriptos anteriormente una table de frecuencias del orden obtenido al generar los recorridos correspondientes en cada una de las corridas. Es decir, se indica cuántas veces una heurística ha sido primera, segunda, etc, en cada corrida.

Consideramos que una heurística es mejor que otra según este criterio cuando genera un recorrido más corto para el mismo grupo de clientes.

Las tablas de frecuencia para cada caso son las siguientes:

Tabla 1: Horario continuo de 8 a 21 horas. Parada de 10 minutos en cada cliente.

Orden	Heurística N°				
	1	2	3	4	5
1	2	3	1		2
2	3	2			1
3			5		1
4	1			4	2
5	1	2	1	3	1

Tabla 2: Horario discontinuo (con ventanilla) de 8 a 13 y de 16.30 a 21 hs. Parada de 10' en cada cliente.

Orden	Heurística N°				
	1	2	3	4	5
1	4	1	3		2
2	6	1	1		2
3		4	1		5
4		1	4	4	1
5		3	1	6	

Tabla 3: Horario continuo 8 a 18 hs. Paradas de 30' en cada cliente.

Orden	Heurística N°				
	1	2	3	4	5
1	7		3		5
2	5	4	2		5
3	2	6	5		3
4	1	5	5		2
5				15	

Tabla 4: Horario continuo de 8 a 21 hs. Paradas de 30' en cada cliente

Orden	Heurística N°				
	1	2	3	4	5
1	4	4	1		2
2	6	2	1		2
3	1	1	7		2
4		4	2		5
5				11	

Tabla 5: Horario continuo de 8 a 20 hs. Paradas de 30' en cada cliente

Orden	Heurística N°				
	1	2	3	4	5
1	5	5	2		5
2	8	7	1		2
3	3	2	3		8
4	1	3	10	1	2
5			1	16	

VIII. ANALISIS DE RESULTADOS

El resultado más sorprendente para nosotros es el buen comportamiento de la heurística "original" que diseñamos para otro trabajo de la facultad y que será objeto de nuevos estudios con un subsidio para investigación.

A simple vista se detecta que la heurística N° 4 "de inserción más cara" o "de inserción más lejana" es la que revela peor desempeño en cuanto a distancia recorrida, además, requiere mucho tiempo de cómputo, superada en ese aspecto sólo por la número 5 "de inserción más barata".

Analizando la tabla 1 se ve prácticamente un empate entre las heurísticas 1 y 2 en la mejor situación, siguiéndoles la 5 y luego la 4.

En la tabla 2 se ve una clara ventaja de la heurística N° 1 seguida por un empate de las heurísticas 2 y 3. Quedaría como peor la número 5 si no tenemos en cuenta la 4 que por obvio mal comportamiento no necesitamos seguir considerando.

En la tabla 3 asignamos el siguiente orden a las heurísticas:

Orden	1	2	3	4
Heurística	1	5	3	2

En la tabla 4 estimamos que los resultados son:

Orden	1	2	3	4
Heurística	1	2	5	3

Por último la tabla 5 nos muestra un empate en el primer puesto entre las heurísticas 1 y 2 seguidas por la 5 y luego por la 3.

Teniendo en cuenta todo lo expuesto, consideramos que la heurística Nº 1 a la que llamamos "original" es la que presenta la mejor performance, ya que en distancias se mantiene en el primer puesto (sola o empatada) y es la más rápida en tiempo de cómputo.

La heurística 2 tiene un buen comportamiento pero, como se describe en la literatura, tanto puede dar una buena solución como una mala. Esto se ve mejor analizando los resultados de cada corrida.

En tercer lugar consideramos a la heurística número 5 y en cuarto lugar a la número 3.

BIBLIOGRAFIA

- CHRISTOFIDES, N., *Vehicle Routing, The travelling Salesman Problem*-Eds. Lawler et alii, Wiley 1985.
- CHRISTOFIDES, N., MINGOZZI, A. y TOTH, P., *Exact Algorithms for the Vehicle Routing Problem. Based on Spanning tree and Shortest Path Relaxations*, *Mathematical Programming* 20, 1981, pág. 255/282.
- DESROCHERS, M. y VERHOOG T., *A New Heuristic for the Fleet Size and Mix Vehicle Routing Problem*, *Computers Operations Research*, Vol. 18, Nº 3, 1991.
- FISHER, MARSHALL y JAIKUMAR, RAMCHANDRAN, *A Generalized Assignment Heuristic for Vehicle Routing*, *Networks*, Vol 11 1981, págs. 109/124.
- GILLET, B.E. y MILLER, L. R., *A Heuristic Algorithm for the Vehicle Dispatch Problem*, *Opns. Res.*, Vol 22, págs. 340/349.
- MORET, B.M.E. y SHAPIRO, H.D., *Algorithms from P to NP*, The Benjamin/Cummings, 1991.
- SALOMON, Marius et alii, *Vehicle Routing and Scheduling Problems with Time Window Constraints: Efficient Implementations of Solution Improvement Procedures*, *Vehicle Routing: Methods and Studies*, Golden and Assad Ed., Elsevier, 1988.
- WALUKIEWICZ, Stanislaw, *Integer Programming*, PWM, Warszawa, 1991.