



Introducción

En optimización combinatoria se busca la mejor manera de resolver un problema entre un conjunto de posibles soluciones al mismo. Para hablar de la mejor manera, se debe asociar cada una de las posibles soluciones a una función objetivo que nos permita valuarlas y compararlas entre sí.

Estos problemas son de mucho interés en la práctica por que se pueden modelar situaciones reales en donde tenemos que tomar una decisión sobre una tarea y la valuación puede representar una métrica que nos resulte de interés como una ganancia asociada o un costo a pagar.

Existen una gran variedad de problemas reales que se pueden resolver mediante optimización combinatoria. En nuestro caso apuntaremos a los problemas de ruteo de vehículos (VRP, por sus siglas en inglés). Hay muchas situaciones, generalmente relacionadas a problemas logísticos, en donde se deben rutear vehículos con el propósito de transportar una mercadería por ejemplo.

Los problemas de tipo VRP pueden ser muy variados dependiendo del contexto donde se estén aplicando. Existen muchas variaciones de este tipo de problemas. En ciertos casos la flota que se cuenta para realizar el ruteo puede ser homogénea y en otros no. A veces existe un depósito central de donde deben comenzar y a donde deben terminar las rutas. Los vehículos de la flota pueden tener restricciones adicionales como una capacidad máxima de carga, una distancia máxima que puedan recorrer en una ruta, o un tiempo de servicio máximo. También puede haber condiciones adicionales del lado de los puntos a visitar o de los caminos a tomar. Por poner ejemplos, los puntos podrían representar clientes a visitar que solo manejan un cierto intervalo horario. Las aristas podrían tener un costo asociado que no es fijo sino que depende de factores como la congestión en la red de tráfico.

Dentro de este gran abanico de problemas sobre su ruteo, en este TP intentaremos resolver el Problema de Ruteo de Vehículos con Capacidad.

Problema

El Problema de Ruteo de Vehículos con Capacidad (CVRP por sus siglas en inglés) se puede definir formalmente de la siguiente manera. Sea G un grafo con V su conjunto de vértices y E su conjunto de aristas, que asumiremos no dirigidas. V representa los puntos a recorrer enumerados de 1 a n , siendo 1 el depósito central y el resto de los vértices todos los clientes que se deben visitar. Para cada par de vértices (i, j) se tiene un valor asociado en la arista que los conecta que representa qué tanto nos cuesta movernos entre esos dos vértices. A su vez, cada vértice i cuenta con una demanda asociada que representa el volumen de entrega que se debe satisfacer con cada cliente.

En el depósito se cuenta con una flota, a priori no acotada, de vehículos con capacidad uniforme C .

El problema consiste en encontrar un conjunto de rutas de mínimo costo asociado que cumplan:

- Cada vértice, sin contar al depósito, es visitado exactamente una vez por algún vehículo de la flota.
- Cada vehículo comienza y termina su ruta en el depósito.
- La suma de las demandas de los clientes visitados por cada vehículo no puede exceder la capacidad C del mismo.

El costo asociado a un conjunto de rutas es la suma de los costos de las rutas que lo componen, el cual a su vez es la suma de los costos de las aristas que se deben recorrer en dicha ruta.

Dado que el CVRP pertenece a la categoría de problemas \mathcal{NP} -hard, no buscaremos dar la solución óptima para cada instancia sino una de la mejor calidad posible.

Parámetros y formato de entrada/salida

Para este trabajo se utilizarán las instancias de trabajo [3] disponibles en <http://vrp.atd-lab.inf.puc-rio.br/index.php/en/>.

Dichas instancias cumplen con el formato impuesto por la TSPLIB95. Se adjunta con el enunciado de este trabajo la especificación formal de los archivos de entrada, el cual se puede resumir de la siguiente manera.

Cada archivo contendrá una lista de secciones con información particular. Las instancias que utilizaremos presentan las siguientes secciones:

- NAME: indica el nombre de la instancia.
- COMMENT: un string para realizar algún comentario sobre la instancia.
- TYPE: indica el tipo de problema. En nuestro caso el valor de este campo siempre será CVRP.
- DIMENSION: indica la cantidad de puntos en el plano a visitar.
- EDGE_WEIGHT_TYPE: indica de que manera se entregarán las distancias entre los puntos. En nuestro caso el valor de este campo siempre será EUC_2D, lo cual significa que se tomará la distancia euclídea entre los puntos (más detalles en la especificación formal adjunta).
- CAPACITY: indica la capacidad de los vehículos.
- NODE_COORD_SECTION: es una lista con los puntos del problema. Cada fila tendrá la información de un punto que consta de su id y sus coordenadas x e y en el plano.
- DEMAND_SECTION: es una lista con la demanda de cada punto a visitar. Cada fila tendrá la información de un punto que consta de su id, seguido de su demanda.

- DEPOT_SECTION: una lista de cuales puntos son considerados depósitos. La lista tiene un id por fila y culmina con un -1 .
- EOF: indica el fin del archivo de entrada.

Para la salida se debe imprimir una primera línea que contenga un entero C , que representa la cantidad de vehículos utilizados en la solución. Luego continuarán C líneas. En cada línea se debe especificar la ruta recorrida por el vehículo i . Una ruta consiste de una lista de enteros separados por un espacio, representando el orden de los puntos que dicho vehículo debe visitar. Por último, se debe imprimir una línea final con el valor de la función objetivo, es decir el costo total de todas las rutas presentadas.

Enunciado

En el presente trabajo práctico se pide:

1. Describir el problema de CVRP dando ejemplos y soluciones
2. Describir situaciones de la vida real que puedan modelarse utilizando CVRP.
3. Diseñar e implementar para el CVRP las siguientes soluciones aproximadas:
 - Heurística constructiva de *savings* mediante el método de Clark-Wright [1].
 - Al menos una heurística constructiva golosa adicional.
 - Heurística constructiva de *cluster-first, route-second*. Clusterizando mediante *sweep algorithm* y luego resolviendo los TSPs restantes de manera heurística.
 - Heurística constructiva de *cluster-first, route-second* con otra alternativa de clusterización y resolución de TSPs posteriores.
 - Metaheurística de Simulated Annealing [2].
4. Para los métodos implementados, desarrollar los siguientes puntos:
 - a) Explicar detalladamente el algoritmo implementado.
 - b) Calcular el orden de complejidad temporal de peor caso del algoritmo.
 - c) Describir (si es posible) instancias de CVRP para las cuales el método no proporciona una solución óptima. Indicar (si es posible) qué tan mala puede ser la solución obtenida respecto de la solución óptima.
 - d) Realizar una experimentación que permita observar la performance del algoritmo comparando la calidad de las soluciones obtenidas y los tiempos de ejecución en función de la entrada (y de otros parámetros de ser apropiado). Dentro de los casos de prueba se deben incluir también, como casos patológicos, aquellos descritos en el ítem 4c. En caso de que el algoritmo tenga algún parámetro configurable que determine su comportamiento (la metaheurística por ejemplo, aunque queda abierto a los demás también), se debe experimentar variando los valores de los parámetros y elegir, si es posible, la configuración que mejores resultados provea para el grupo de instancias utilizado. Presentar los resultados obtenidos mediante gráficos adecuados.

5. Una vez elegidos los mejores valores de configuración para cada heurística implementada, realizar una experimentación **sobre un conjunto nuevo de instancias** para observar la performance de los métodos comparando nuevamente la calidad de las soluciones obtenidas y los tiempos de ejecución en función del tamaño de entrada. Presentar todos los resultados obtenidos mediante gráficos adecuados y discutir al respecto de los mismos.

Fechas de entrega

- Formato Electrónico: Jueves 21 de Noviembre de 2019, hasta las **23:59 hs**, enviando el trabajo (informe + código) a la dirección `algo3.dc@gmail.com`. El subject del email debe comenzar con el texto [TP3] la lista de apellidos de los alumnos.
- Formato físico: Viernes 22 de Noviembre de 2019, a las 18 hs. en la clase de laboratorio.

Importante: El horario es estricto. No se considerarán los correos recibidos después de la hora indicada.

Referencias

- [1] G. Clarke and J. W. Wright. Scheduling of vehicles from a central depot to a number of delivery points. Operations Research, 12(4):568–581, 1964.
- [2] Ibrahim Osman. Meta-strategy simulated annealing and tabu search algorithms for the vehicle routine problem. 41:421–451, 12 1993.
- [3] Diego Pecin, Artur Alves Pessoa, Marcus Poggi, and Eduardo Uchoa. Improved branch-cut-and-price for capacitated vehicle routing. In Integer Programming and Combinatorial Optimization - 17th International Conference, IPCO 2014, Bonn, Germany, June 23-25, 2014. Proceedings, pages 393–403, 2014.