

Trabajo práctico 3: El problema del *cartero chino*

Primera entrega: 2011, 25 de Noviembre hasta las 18:00 horas.

Segunda entrega: 2011, 7 de Diciembre (miércoles) hasta las 18:00 horas.

Sea $G = \langle V, E \rangle$ un grafo. Cada arista $e \in E(G)$ tiene asociado un peso positivo $p(e)$. Algunas aristas son dirigidas y otras no. Un *tour chino* es un circuito que pasa por todas las aristas al menos una vez, respetando la orientación. La *longitud* del circuito es la suma de los pesos de las aristas del tour (contando repeticiones).

El problema del *cartero chino* (el problema CC), consiste en encontrar el tour chino cuya longitud sea mínima.

1. Describir situaciones de la vida real que puedan modelarse utilizando CC.
2. Desarrollar e implementar un algoritmo exacto para CC.
3. Desarrollar e implementar una heurística constructiva para CC.
4. Desarrollar e implementar una heurística de búsqueda local para CC.
5. Desarrollar e implementar un algoritmo que use la metaheurística GRASP [1] para CC. En cada iteración de GRASP utilizar como primera fase alguna modificación de la heurística dada en 3, y como segunda fase la heurística dada en 4. *Fijar* todos los parámetros involucrados (tamaño de la lista restringida de candidatos) en la primera fase de GRASP, tamaño de la vecindad, cantidad de vecinos a visitar, cantidad de iteraciones máxima, cantidad de iteraciones sin mejorar, etc). *Justificar cada elección mediante experimentos prácticos. Justificar también la elección de los valores a ser considerados para esas pruebas.*
6. Para cada uno de los métodos de los ejercicios 2 a 5:
 - calcular la complejidad,
 - tratar de describir instancias de CC para las cuales el método no proporciona una solución, óptima? *Qué tan mala puede ser la solución obtenida respecto de la solución óptima?*,
 - aplicar el método a varias instancias de CC, respetando los formatos de archivos que se indican, más abajo,
 - analizar la calidad de las soluciones obtenidas y el tiempo de ejecución.
7. *Presentar los resultados obtenidos en el ejercicio anterior mediante gráficos adecuados, y utilizarlos para comparar los distintos métodos entre sí.*

Ej3.in: cada archivo de entrada contiene varias instancias. En la primera línea del archivo figurarán las cantidades n , m_1 y m_2 de vértices, aristas no orientadas y aristas orientadas respectivamente. Luego, en las primeras m_1 líneas, aparecen las aristas *no* orientadas como i, j, p (donde p es el peso de la arista) y en

las siguientes m_2 líneas aparecen las aristas orientadas como i, j, p (donde p es el peso de la arista y se la interpreta orientada desde i hacia j). Luego en las $m_1 + m_2$ líneas aparecen los pesos correspondientes a las aristas.

Ej1.out: para cada entrada, en el archivo de salida figurará, en la primera línea, la longitud del circuito y en las siguientes líneas, la lista de aristas ordenadas que lo conforman.

Referencias

- [1] Thomas A. Feo and Mauricio G. C. Resende. *Greedy Randomized Adaptive Search Procedures*. Journal of Global Optimization, pp. 1–27, 1995.