TP 2: Recorridos y árbol generador mínimo

28 de Mayo, 2023 Algoritmos y Estructuras de Datos III

Integrante	LU	Correo electrónico
Zaid, Pablo	869/21	pablozaid2002@gmail.com
Arienti, Federico	316/21	fa.arienti@gmail.com

Facultad de Ciencias Exactas y Naturales

Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2610 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (++54 +11) 4576-3300

http://www.exactas.uba.ar

RESUMEN

En la teoría de grafos, el problema del *árbol generador mínimo*[1] —o *minimum spanning tree*, en inglés— se refiere al problema de encontrar, para un grafo conexo $G = (V,\ E)$ con función de peso $w : E \to \mathbb{R}$ asociada, un subgrafo conexo y acíclico de G —es decir, un árbol generador— que minimice la suma total del peso de sus aristas.

Existen diversos métodos para la resolución de este problema. Entre ellos, los algoritmos *golosos* de *Prim* y de *Kruskal*, que se basan en la selección de aristas de peso mínimo *seguras*[2] para la construcción de una solución.

El siguiente informe evalúa el problema de los *módems*, explicado en el próximo apartado, y lo reformula como un problema de *árbol generador mínimo* que aprovecha el invariante del algoritmo de *Kruskal*. Además, evalúa la eficiencia de la solución propuesta de manera empírica en función de la aplicación de diferentes estructuras de datos: *priority queue* y *disjoint set*; y heurísticas: *union by rank* y *path compression*[3].

Palabras clave: *árbol generador mínimo, algoritmo de Kruskal.*

Índice

[1] Ver Thomas H. Cormen; Charles E. Leiserson; Ronald L. Rivest y Clifford Stein. Introduction to algorithms. 2009. Sección 23: *Minimum Spanning Trees.*

[2] Una arista es *segura* si y sólo si al agregarla a un subgrafo de un árbol generador, el resultado sigue siendo un subgrafo de algún árbol generador.

[3] Ver nota al pie (1). Sección 21: *Data Structures for Disjoint Sets.*

1. El problema de los módems

El problema de los *módems* que consideraremos tiene la siguiente premisa. Dado un conjunto de N oficinas

$$O := \{o_1 \ ... \ o_N\}$$

donde, para cada $1 \leq i \leq N$, la oficina o_i se representa por su posición $(x_i, \ y_i)$ en el plano cartesiano, queremos encontrar el costo mínimo que deberemos pagar en cables para conectar todas las oficinas a internet[4].

Para ello, vamos a contar con $W < N$ módems a distribuir entre las oficinas e incurriremos en un costo de U o V pesos, $U \leq V$, respectivamente, por cada centímetro de cable UTP o de fibra óptica utilizado. Dado que los cables UTP tienen ciertas restricciones, estos se podrán utilizar si y sólo si la distancia entre dos oficinas es menor o igual a R centímetros.

Por ejemplo, si

$$O = \{(0, \ 0), \ (0, \ 1), \ (1, \ 0)\}, \ R = 2, \ W = 1, \ U = 1 \ \text{y} \ V = 1$$

luego una solución puede situar un módem en la oficina $(0, \ 0)$ y conectarla a cada una de las restantes con un centímetro de cable UTP, por un costo total de dos pesos.

1.1. Modelado como un problema de árbol generador mínimo. Dada la descripción anterior, vamos a mostrar que el problema se puede modelar como una variante del problema del árbol generador mínimo. Lo detallamos a continuación.

Sea G un grafo pesado completo cuyos vértices son el conjunto de oficinas O. Por cada par de vértices $o_i, \ o_j \in O$, definimos el costo de la arista $(o_i, \ o_j)$ como

$$w(o_i, \ o_j) = \begin{cases} d_{ij} \cdot U & \text{si } d_{ij} \leq R \\ d_{ij} \cdot V & \text{si no} \end{cases}$$

donde d_{ij} es la distancia euclideana entre ambas oficinas.

Luego, de haber, al menos, una oficina con un módem, G modela una solución no mínima al problema de los módems en la que todas las oficinas están conectadas entre si con la opción más barata disponible entre un cable UTP y uno de fibra óptica.

Si $W = 1$, sigue que, para mantener a todas las oficinas conectadas y minimizar el costo empleado en los cables, basta encontrar un árbol generador mínimo de G.

Sin embargo, si $W > 1$, podemos reducir aún más el costo si, en vez de encontrar un árbol generador mínimo, encontramos un bosque generador mínimo de W componentes. Es decir, que incluya todos los vértices de G y cuyo peso sea mínimo.

Vamos a demostrar que este bosque es óptimo y que basta modificar el algoritmo de *Kruskal* para que termine en la iteración $N - W$ para encontrarlo.

1.2. Demostración de optimalidad. Supongamos, por absurdo, que un bosque generador mínimo de W componentes B de G no es una solución óptima al problema de los módems. Luego, debe existir otro subgrafo B' de G que tiene menor peso y que, una vez dispuestos los W módems, provee de internet a todas las oficinas.

Notamos primero que B' también debe ser un bosque generador de W componentes. Esto se debe a que: si no fuera generador, entonces no estaríamos considerando todas las oficinas en nuestro modelo; y, si no fuera bosque de W componentes, entonces podríamos reducir su

[4] Notar que una oficina tendrá acceso a internet si y sólo si tiene un módem o está conectada a una oficina con acceso a internet.

peso si eliminamos suficientes aristas hasta formar uno. Notar, a su vez, que no puede tener más componentes. Si no, no habría suficientes módems para conectar todas las oficinas.

Sigue que B' es un bosque generador de W componentes conexas de G que tiene un peso menor que un bosque generador mínimo de W componentes de G. $\rightarrow\leftarrow$

1.3. Demostración de correctitud. Dicho esto, vamos a demostrar primero la siguiente proposición. Dado un grafo conexo G, un subgrafo B de un árbol generador mínimo T de G es un bosque generador mínimo de k componentes de G si se compone de las $n - k$ aristas de peso mínimo de T.

Por propiedad de árboles, está claro que si B tiene $n-k$ aristas, entonces B es un bosque generador de k componentes conexas. Vamos a demostrar entonces, por el absurdo, que es mínimo.

Supongamos que existe un bosque generador B' de k componentes de G que tiene menor peso que B. Luego, podemos construir un árbol generador de G tomando un conjunto E de $k-1$ aristas de T que unan a las k componentes conexas diferentes en B'. Esto lo podemos hacer ya que, si tal conjunto no existierá, entonces habría, al menos, un par de vértices ubicados en dos componentes conexas diferentes de B', para los cuales no existe un camino que los una en T. Lo que es absurdo, dado que T es un árbol generador.

Dicho esto, como estas aristas tienen, a lo sumo, el peso de las $k-1$ aristas máximas en T, sigue que $B' + E$ es un árbol generador de G con peso menor que el árbol generador mínimo T. $\rightarrow\leftarrow$

En consecuencia, dado que el invariante del algoritmo de *Kruskal* afirma que, para la k-ésima iteración, el grafo T_k construido por el algoritmo es un subgrafo de un árbol generador mínimo de G y, en cada iteración, el algoritmo agrega una arista segura de peso mínimo a T_k, para todo $1 \leq k \leq n-1$, entonces T_{n-k} es un subgrafo de un árbol generador mínimo de T compuesto por las $n-k$ aristas de peso mínimo de T. Sigue que T_{n-k} es un bosque generador mínimo de k componentes.

1.4. El algoritmo. Dicho todo esto, proponemos el siguiente algoritmo como solución al problema de los *módems*.

1.5. Complejidad temporal y espacial.

2. Evaluación empírica