MA3705 Algoritmos Combinatoriales
Profesor: Iván Rappaport
Auxiliares: Benjamín Jauregui, Antonia Labarca

Auxiliar 4: Árboles Generadores de Costo Mínimo v2

8 de septiembre de 2023

P1. Considere la siguiente variante del algoritmo de Kruskal, denotado como *Kruskal reverso*. Dada una entrada (G, ω), con $G = (V, E)$ grafo y ω función de pesos inyectiva sobre las aristas, el algoritmo comienza con $F = E$ y ordenas las aristas de mayor a menor peso. Luego, iterativamente, en orden decreciente de peso, verifica si la arista e_i participa de un ciclo en F. Si es que participa del ciclo, saca a e_i de F y sigue analizando a la arista e_{i+1}, que es la $(i+1)$-esima arista de mayor peso en G. El algoritmo retorna (V, F).

a) Escriba el algoritmo de *Kruskal reverso* en pseudocódigo.

b) Analice la complejidad del algoritmo.

c) Pruebe que si C es un ciclo de G y $e_0 \in E$ es tal que $e_0 \in E(C)$ y $e_0 = \arg\max_{e \in E(C)} \omega(e)$, entonces e_0 no puede participar de un MST T de G.

d) Concluya que el algoritmo es correcto, es decir, que devuelve un MST de G.

P2. Sea $G = (V, E)$ grafo, $\omega \colon E \to \mathbb{R}_+$ función inyectiva de pesos sobre las aristas y T_0 un MST de (G, ω). Si $\mathcal{ST}_G$ es el conjunto de todos los árboles generadores de G, se define el *segundo MST* T' como $T' = \arg\min_{T \in \mathcal{ST}_G - T_0} \omega(T)$. Diseñe un algoritmo a tiempo polinomial, que dado un grafo G y una función de pesos inyectiva ω sobre las aristas, devuelva un segundo MST de G.

Hint: Pruebe primero que T' es segundo MST ssi difiere en exactamente una arista del MST T_0

P3. Propuesto: Considere el siguiente algoritmo, denominado algoritmo de *Boruvka*.

Algoritmo 1: Algoritmo de Boruvka

Input : Grafo $G = (V, E)$ simple y conexo, función de pesos $\omega \colon E \to \mathbb{R}_+$
Output: Árbol $H = (V, F)$
$F = \emptyset$;
completado $= false$;
mientras *not completado* **hacer**
  1. Calcular componentes conexas de (V, F);
  2. Para toda componente conexa C de (V, F), inicializar $c(C) = \emptyset$;
  3. Calcular la arista de peso mínimo de $\delta(C)$ para toda componente conexa C de (V, F) si existe, y almacenarla en $c(C)$;
  si *toda componente conexa C tiene* $c(C) = \emptyset$ **entonces**
    completado $= true$;
  fin
  en otro caso
    añadir a F todas las aristas $c(C)$ con $c(C) \neq \emptyset$;
  fin
fin
devolver $H = (V, F)$.

a) Escriba el algoritmo descrito con palabras como pseudocódigo.

b) Estudie la complejidad del algoritmo.

c) Pruebe que el grafo devuelto por el algoritmo es un MST de (G, ω). Es decir, pruebe que H es un árbol y que es el menor peso entre todos los posibles árboles generadores. Para esta última parte, razone por contradicción.