

TD n°9

Arbres couvrant minimum.

Dans ce TD, $G = (V, E)$ est un graphe fini, non orienté, sans boucle, dont les arêtes sont pondérées. La fonction $\omega : E \rightarrow \mathbb{R}$ est la fonction de poids des arêtes. Le poids de G , noté $\omega(G)$, est la somme des poids des arêtes de G (c'est-à-dire $\omega(G) = \sum_{(x,y) \in E(G)} \omega(x,y)$).

La Minute Historique

Le problème de trouver un arbre couvrant minimum (ACM) est un problème très ancien. Et aussi bizarre que cela puisse paraître, c'est un problème qui est même plus vieux que l'informatique elle-même. En effet en 1926, Otakar Borůvka fut un des premiers (si ce n'est le premier) à fournir un algorithme pour trouver "efficacement" un arbre couvrant minimum. La motivation d'alors était de concevoir un réseau de distribution électrique pour la Moravie du Sud. Avec la contrainte évidente : que cela coûte le moins possible. Ce n'est qu'au milieu des années 50 et l'avènement de l'informatique que ce problème est revenu à la mode. C'est à Joseph B. Kruskal (1956) et à Robert Prim (1957) que nous devons les algorithmes que nous allons étudier aujourd'hui.

Définition

un arbre T est un arbre couvrant de G si $T = (V, A)$ est un arbre dont l'ensemble des sommets V est celui de G et l'ensemble des arêtes A est un sous ensemble des arêtes de G ($A \subseteq E$). De plus, un arbre couvrant T est dit minimum (ou parfois : de poids minimum) si pour tout arbre couvrant T_i de G , on a $\omega(T) \leq \omega(T_i)$.

Exercice 1 [Exemple] Trouvez sur un arbre couvrant de poids minimal sur le graphe suivants en appliquant les deux algorithmes vus en cours.



