ADS Tutorial 4

Instructor: Aris Filos-Ratsikas TA: Kat Molinet

October 17, 2024

Problem 1

Let $G = (V, E)$ be a connected graph with positive edge costs, and let G_T be a Minimum Spanning Tree (MST) on G. Let G' be a connected graph that is the same as G, but with different edge costs c'_e rather than c_e for edge edge $e \in E$. In which of the following cases will G_T still be a spanning tree of the connected graph G'?

(a) When $c'_e = c_e + 17$.

(b) When $c'_e = 17c_e$.

(c) When $c'_e = c_e^2$.

(d) All of the above.

(e) None of the above.

Problem 2

In this problem we will consider the *uniqueness* of MSTs on connected graphs.

(a) First consider connected graphs in which the edge costs might not be distinct. Draw an example of such a graph that has two different MSTs.

(b) Now consider any connected graph G for which the edge costs are all distinct. Prove that there is a unique MST on G.

Problem 3

Suppose you are given a connected graph $G = (V, E)$ with edge costs which are all distinct, with $|V| = n$ and $|E| = m$, and a specified edge $e^* \in E$. Design an algorithm with running time $O(m + n)$ which returns "yes" if e^* is contained in a MST of G and "no" otherwise.

To prove the correctness of your algorithm, you may first want to prove the following statement:

> *Some edge $e = (u, v)$ does not belong to a MST of G if and only if there is a path from v to w which consists entirely of edges that have smaller cost than e.*

Hint: You may want to use the Cut Property and the Cycle Property to prove the statement.

Problem 4

Consider an alternative algorithm for computing an MST G_T on a connected graph $G = (V, E)$ with distinct edge weights. At iteration t of the algorithm, let G_T^t be the graph consisting of the set of nodes V and the set of edges that have been added by the algorithm to the developing MST in iterations $1, \ldots, t-1$. Let $\mathcal{C}^t$ be the set of connected components of G_T^t. For each connected component $C \in \mathcal{C}^t$, add to the MST the edge with the smallest cost $e = (v, u)$ where v is in C and u is not in C. Update the graph to G_T^t and the set of connected components to $\mathcal{C}^{t+1}$ and repeat. Before the first iteration G_T^0 contains no edges and $\mathcal{C}^0 = \{\{v_1\}, \{v_2\}, \ldots, \{v_n\}\}$, i.e., every node is a connected component by its own.

Show that this algorithm terminates in $O(m \log n)$ time and that in the end it produces a MST of G. It might be useful to first prove the following statement, using the Cut Property for the proof:

> *Let $u \in V$ be any node in G. The MST of G must contain edge (u, w), which has the minimum cost amongst all edges incident to u.*

Note: This algorithm was first proposed by Otakar Borůvka in 1926, in a response to a request to construct an electrical network connecting several cities using the least amount of wire. The algorithm was rediscovered multiple times in the late 30s, early 50s and early 60s.