CS60007 Algorithm Design and Analysis 2018
Supplementary for Lecture 3

Palash Dey
Indian Institute of Technology, Kharagpur

July 23, 2018

Proof of Correctness of the Kruskal's Algorithm for Finding an MST

The proof of correctness of the Kruskal's algorithm follows immediately from the following more general result.

Lemma 1. *Let* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ *be an undirected weighted graph,* $e_1, e_2, \ldots, e_{n-1}$ *the edges in* $\mathcal{G}$ *in the order they are picked by the Kruskal's algorithm. Then, for every* $1 \leqslant k \leqslant n-1$*, there exists an MST of* $\mathcal{G}$ *which includes* $e_1, e_2, \ldots, e_k$.

Proof. We prove it by induction on k. For $k = 1$, the statement follows from the home task given in the class. Let us assume the result for k. That is, let us assume that there exists an MST T_k of $\mathcal{G}$ which includes $e_1, e_2, \ldots, e_k$. If T_k already includes e_{k+1}, then we have nothing to prove. So let us assume without loss of generality that T_k does not include e_{k+1}. Let us consider the subgraph $\mathcal{H} = T_k \cup \{e_{k+1}\}$. Let $\mathcal{C}$ be the cycle in $\mathcal{H}$. The cycle $\mathcal{C}$ includes the edge e_{k+1} since T_k is a tree. The cycle $\mathcal{C}$ also includes an edge $f \in E[\mathcal{G}] \setminus \{e_1, e_2, \ldots, e_k\}$ by the definition of e_{k+1} (that the Kruskal's algorithm picks it in the $(k+1)$-th iteration). Also by the definition of e_{k+1}, we have $wt(e_{k+1}) \leqslant wt(f)$. Hence the weight of the spanning tree $T_{k+1} = \mathcal{H} \setminus \{f\}$ is at most the weight of T_k. Thus T_{k+1} is also an MST of $\mathcal{G}$ and it includes $e_1, e_2, \ldots, e_{k+1}$. This completes the proof of the Lemma. □