# Kruskal's Algorithm for Minimal Spanning Tree

Given a simple, undirected, connected, weighted graph $G = (V, E, w)$, where $V$ is the vertex set, $E$ is the edge set, and $w : E \longrightarrow \mathbb{R}$ is the weight function which assigns a positive weight to each of the edges, the following is a proof that Kruskal's algorithm produces a minimal spanning tree (MST).

**Theorem.** Any spanning tree $T$ for $G$ constructed by Kruskal's algorithm is a minimal spanning tree.

**Proof.** The proof is by contradiction.

Let $T$ be a subgraph of $G$ constructed by Kruskal's algorithm, since the algorithm constructs a subgraph $T$ with $n$ vertices and $m$ edges such that $T$ has no cycles, and $m = n - 1$, then $T$ is a spanning tree for $G$.

Suppose that the edges in $T$ are $\{e_1, e_2, \ldots, e_{n-1}\}$, and the algorithm places the edges in the tree $T$ in that order, so that

$$w(e_1) \leq w(e_2) \leq \cdots \leq w(e_{n-1}).$$

Suppose also that $T$ is **not** a minimal spanning tree for $G$.

Among all the minimal spanning trees for $G$, choose one, say $H$, which has a maximum number of edges in common with $T$. The trees $H$ and $T$ are not identical, so $T$ has at least one edge that does not belong to $H$.

Let $e_k$, where $1 \leq k \leq n - 1$, be the first edge of $T$ that is not in $H$, and define $G_0 = H + e_k$.

Then $G_0$ has exactly one cycle, since if $G_0$ contained two cycles $C_1$ and $C_2$, then we could remove an edge from one of the cycles to get a connected graph $G_0'$ which contains a cycle such that $|V(G_0')| = n$ and $|E(G_0')| = n - 1$, but this implies that $G_0'$ is a tree, which is a contradiction. Thus, $G_0$ contains exactly one cycle $C$.

Since $T$ has no cycles, there is an edge $e_0$ of $C$ that is not in $T$. The graph $T_0 = G_0 - e_0$ is also a spanning tree for $G$. To see this, note that the graph $T_0 = G_0 - e_0$ is connected, since $G_0$ is connected and the edge $e_0$ is removed from the cycle. Also, $|V(T_0)| = n$ and $|E(T_0)| = |E(G_0)| - 1 = n - 1$, and therefore $T_0$ is connected and satisfies the tree formula: $|V(T_0)| = |E(T_0)| + 1$, thus, $T_0$ is a tree. It is a spanning tree for $G$ since it contains the same vertices as $G_0$, which are the same vertices as those of $H$, which are the same as those of $G$. Also,

$$w(T_0) = w(H) + w(e_k) - w(e_0).$$

Now, since $H$ is a minimal spanning tree for $G$, then $w(H) \leq w(T_0)$, and it follows that

$$w(e_0) \leq w(e_k).$$

By Kruskal's algorithm, $e_k$ is an edge of minimum weight such that the subgraph consisting of the edges $\{e_1, \ldots, e_{k-1}\} \cup \{e_k\}$ is acyclic. But the subgraph consisting of the edges $\{e_1, \ldots, e_{k-1}\} \cup \{e_0\}$ is a subgraph of the minimal spanning tree $H$, and is therefore acyclic, and since Kruskal's algorithm added $e_k$ instead of $e_0$, then

$$w(e_k) \leq w(e_0),$$

and so

$$w(e_k) = w(e_0).$$

Thus, $w(T_0) = w(H)$, and $T_0$ is also a minimal spanning tree for $G$.

However, $T_0$ has more edges in common with $T$ than $H$ has in common with $T$, which contradictions our original assumption. Therefore, the spanning tree constructed by Kruskal's algorithm is a minimal spanning tree.

$\square$