Chapter 11

Minimum Spanning Tree Recitation.

11.1 The MST Problem.

Definition 11.1.1. A spanning tree T of a graph $G = (V, E)$ is a subset of edges in E such that T is a tree (having no cycles), and the graph (V, T) is connected.

Problem 11.1.1 (MST). Let $G = (V, E)$ be a weighted graph with weight function $w : E \to \mathbb{R}$. We extend the weight function to subsets of E by defining the weight of $X \subset E$ to be $w(X) = \sum_{e \in X} w(e)$. The minimum spanning tree (MST) of G is the spanning tree of G that has the minimal weight according to w. Note that in general, there might be more than one MST for G.

Definition 11.1.2. Let $U \subset V$. We define the cut associated with U as the set of outer edges of U, namely all the edges $(u, v) \in E$ such that $u \in U$ and $v \notin U$. We use the notation $X = (U, \bar{U})$ to represent the cut. We say that $E' \subset E$ respects the cut if $E' \cap X = \emptyset$.

Lemma 11.1.1 (The Cut-Lemma)**.** *Let T be an MST of G. Consider a forest $F \subset T$ and a cut X that respects X (i.e. $F \cap X = \emptyset$). Then $F \cup$ arg$\min_e w(e)$ is also contained in some MST. Note that it does not necessarily have to be the same tree T.*

Proof. If $e \in T$, then $F \cup \{e\} \subset T$ and we are done. Otherwise, consider the second case where $e \notin T$. This means that $T \cup \{e\}$ has $|V|$ edges and therefore must have a cycle. Let $\Gamma = T \cup \{e\}$ and let x and y be the endpoints of e (namely $e = (x, y)$). Denote the subset of vertices defining the cut X by U. Without loss of generality, let's assume $x \in U$ and $y \in \bar{U}$.

Since T is connected, there is a path $x \rightsquigarrow y$ in T, denote it by $\mathcal{P}$. Additionally, because $e \notin T$, we have that $e \notin \mathcal{P}$. This means that there must be another edge in $\mathcal{P}$ connecting a vertex in U to a vertex in $\bar{U}$[1]. Let e' be that edge, we have:

1. Both $e', e \in X$ So $w(e) \leq w(e')$.

2. $e \cup \mathcal{P}$ is a cycle in Γ.

[1]Otherwise, walking along $\mathcal{P}$ cannot take one out of U, leading to a contradiction as $\mathcal{P}$ leads to y.

By using the fact that subtracting an edge from a cycle doesn't harm connectivity (see Claim 11.2.2), we can conclude that $\Gamma/\{e'\}$ is connected. Since it has $|V|-1$ edges, it must be a spanning tree. On the other hand, by:

$$w\left(\Gamma/\{e'\}\right) = w\left(T\right) + \overbrace{w(e) - w(e')}^{\leq 0} \leq w\left(T\right)$$

So $\Gamma/\{e'\}$ is an MST. Finally, to close the proof, observe that $F \cup \{e\} \subset \Gamma/\{e'\}$. This means that, we have found an MST that contains $F \cup \{e\}$.

□

11.2 Kruskal Algorithm.

This algorithm constructs the MST iteratively by holding a forest F contained in an MST and then looking for the minimal edge in a cut that it respects. Note, that since F has no cycles, any edge $e \subset E$ that does not create a cycle in F must belong to a cut X that is respected by F. By ensuring that the edges are examined in increasing weight order, we can determine that the first edge that does not create a cycle is also the one with the minimum weight among them. Therefore, according to Lemma 11.1.1, we can conclude that the forest obtained by adding e into F is contained in an MST.

Result: Returns MST of given $G = (V, E, w)$
1 sorts the E according to w
2 define $F_0 = \emptyset$ and $i \leftarrow 0$
3 **for** $e \in E$ *in sorted order* **do**
4   **if** $F_i \cup \{e\}$ *has no cycle* **then**
5         $F_{i+1} \leftarrow F_i \cup \{e\}$
6         $i \leftarrow i + 1$
7   **end**
8 **end**
9 **return** F_i

Algorithm 1: Kruskal alg.

Claim 11.2.1. *For any i, $F_i \subset$ of an MST.*

Proof. By induction.

1. Base. Let T be an arbitrary MST of G. $F_0 = \emptyset \subset T$.
2. Assumption. Assume correctness for any $j < i$.
3. Step. By the induction assumption there is an MST T such $F_{i-1} \subset T$. Denote by $e = (x, y)$ the edge for which $F_i = F_{i-1} \cup \{e\}$. By algorithm definition, $F_i = F_{i-1} \cup \{e\}$ has no cycles (line number (4)). That means that in respect to F_{i-1} x and y belongs to two different connected components, Denote x connected component by U, and the cut it defines by $X = (U, \bar{U})$. Clearly F_{i-1} respects[2] X.

[2] Otherwise U would not be a connected component of F_{i-1}

On the other hand, $w(e) \leq \min_{e' \in X} w(e')$. Any other e' with $w(e') < w(e)$ is either already in F_{i-1} and therefore cannot be in X, or it closes a cycle in F_j for some $j < i$. Since $F_j \subset F_{i-1}$, it also closes a cycle in F_{i-1}. Therefore, it cannot be an edge connecting between U and $\bar{U}$ and does not belong to X.

□

Correctness of Algorithm 1. □

Claim 11.2.2. *Let G be a connected graph containing a cycle C. Then the subtraction of any an edge in C gives a connected graph.*

Proof. Assume, by contradiction, that a graph $G' = G/\{e\}$, where $e \in C$, is not connected. This means that there are two vertices u and v that have a path between them in G, but no such path exists in G'. Denote this path by $\mathcal{P}$ and observe that $e \in \mathcal{P}$, otherwise, $\mathcal{P}$ would also be a path from u to v in G'.

Denote the ends of e by $(x, y) = e$. Also, denote C by $\langle x_0, x_1, ..x_i, x, y, y_0, .., y_j \rangle$, where $y_j = x_0$ and there is an inequality for any other pair of vertices (we used the cycle definition). Then, there is a path $x \rightsquigarrow y$ in C, defined by

$$\langle x_i, x_{i-1}, .., x_1, x_0, y_{j-1}, y_{j-2}, .., y_0, y \rangle$$

We denote this path by $\mathcal{P}'$. By replacing e in $\mathcal{P}$ with $\mathcal{P}'$, we obtain a path $u \rightsquigarrow x \rightsquigarrow^{\mathcal{P}'} y \rightsquigarrow v$, which is a path between u and v that does not contain e. This contradicts the assumption that there is no path between u and v in G'. □