MIT 18.211: COMBINATORIAL ANALYSIS

FELIX GOTTI

Lecture 26: Spanning Trees and Kruskal's Algorithm

Definition 1. Let G be a graph. A subgraph T of G is called a *spanning tree* of G provided that T is a tree and $V(T) = V(G)$.

Every connected graph has a spanning tree.

Proposition 2. *Every connected simple graph has a spanning tree.*

Proof. Let G be a connected simple graph. Then the set $\mathscr{G}$ consisting of all connected subgraphs G' of G with $V(G') = V(G)$ is clearly nonempty (indeed, $G \in \mathscr{G}$). Among all the graphs in $\mathscr{G}$, let T be one minimizing the number of edges. Then T is minimally connected, and so a tree. Since $V(T) = V(G)$, we conclude that T is a spanning tree of G. $\square$

It follows from Cayley's theorem that, for every $n \in \mathbb{N}_{\geq 2}$, the number of spanning trees of the complete graph K_n is n^{n-2}.

Let G be a connected simple graph, and let $\omega\colon E(G) \to \mathbb{R}_{>0}$ be a map. For every subgraph G' of G, define the *weight* of G' to be

$$\omega(G') := \sum_{e \in E(G')} \omega(e).$$

Among all the spanning trees T of G, we would like to find one with minimum weight. To do so, we assume that every edge of G is initially unmarked, and we sequentially mark one of the minimum-weight unmarked edges that do not create any cycle in G with the already marked edges. More formally, we have the following algorithm/recipe.

Kruskal's Algorithm (G is a connected graph and $\omega\colon E(G) \to \mathbb{R}_{>0}$ is a function)

(1) Assume that the edges of G are initially unmarked.
(2) Let S be the subset of $E(G)$ consisting of all unmarked edges that do not create any cycles with the marked edges.
(3) If S is nonempty, then
  - take $e \in S$ with $\omega(e) = \min\{\omega(s) \mid s \in S\}$;
  - mark e;
  - return to step (2).

(4) Set T to be the subgraph of G whose edges are the marked edges.

It is clear that after repeating steps (1) and (2) finitely many times (indeed, $|V(G)|-1$ times), we will obtain a spanning tree T of G. However, it is not obvious at all that such a spanning tree will minimize the sum $\sum_{e \in E(T)} \omega(e)$. We will prove this in the next theorem. First, let us argue the following proposition.

Proposition 3. *Let F and F' be two forests with $V(F) = V(F')$. If $|E(F)| < |E(F')|$, then there is an edge $e \in E(F') \setminus E(F)$ such that the graph we obtain from F by adding the edge e is still a forest.*

Proof. Suppose, towards a contradiction, that if we add any edge $e \in E(F') \setminus E(F)$ to the forest F we produce a cycle. Then every edge in $E(F')$ connects two vertices that belong to the same connected component of F. This implies that the number of connected components k' of F' is at least the number of connected components k of F. Therefore $|E(F')| = |V(F')| - k' \le |V(F)| - k = |E(F)|$, which is a contradiction. $\square$

We are in a position to prove that Kruskal's algorithm yields a minimum spanning tree.

Theorem 4 (Kruskal's Algorithm Correctness)**.** *Let G and $\omega \colon E(G) \to \mathbb{R}_{>0}$ be as above. If T is a spanning tree obtained from successive iterations of steps (1) and (2) above, then $\sum_{e \in E(T)} \omega(e) \le \sum_{e \in E(T')} \omega(e)$ for any spanning tree T' of G.*

Proof. Set $n := |V(G)|$. Suppose, by way of contradiction, that there exists a spanning tree T' of G such that $\sum_{e \in E(T')} \omega(e) < \sum_{e \in E(T)} \omega(e)$. Let $e_1, \dots, e_{n-1}$ and $e'_1, \dots, e'_{n-1}$ be the edges of T and T' labeled in increasing order of weight, that is,

$$\omega(e_1) \le \omega(e_2) \le \cdots \le \omega(e_{n-1}) \quad \text{and} \quad \omega(e'_1) \le \omega(e'_2) \le \cdots \le \omega(e'_{n-1}).$$

Let j be the minimum index in $[\![1, n-1]\!]$ such that

$$\sum_{i=1}^{j} e'_i < \sum_{i=1}^{j} e_i.$$

The minimality of j guarantees that

$$\sum_{i=1}^{j-1} e'_i \ge \sum_{i=1}^{j-1} e_i.$$

Since $\omega(e_1) = \min\{\omega(e) \mid e \in E(G)\}$, we see that $j > 1$. Now consider the subforests F and F' of G determined by the edges $e_1, e_2, \dots, e_{j-1}$ and $e'_1, e'_2, \dots, e'_j$. Since $|E(F)| = j - 1 < j = |E(F')|$, Proposition 3 guarantees the existence of an index $k \in [\![1, j]\!]$ such that $e'_k \notin \{e_1, \dots, e_{j-1}\}$ and the subgraph of G obtained by adding the edge e'_k to F is a forest, and so contains no cycles. However, the fact that

$$\begin{aligned}\omega(e'_k) \leq \omega(e'_j) < \omega(e'_j) &+ \left(\sum_{i=1}^{j} \omega(e_i) - \sum_{i=1}^{j} \omega(e'_i) \right) \\ &\leq \omega(e_j) + \left(\sum_{i=1}^{j-1} \omega(e_i) - \sum_{i=1}^{j-1} \omega(e'_i) \right) \\ &\leq \omega(e)\end{aligned}$$

produces a contradiction as in the j-th step of the Kruskal's algorithm, choosing e'_k would produce a forest with less weight than the one obtained by choosing e_j. $\square$

Practice Exercises

Exercise 1. *Let G be a simple graph. Prove that G is a tree if and only if G does not contain any cycle but connecting any two not adjacent vertices will produce a cycle.*

Exercise 2. *Prove that a graph is a tree if and only if it is connected and has exactly one spanning tree.*

Exercise 3. *Let G be a connected simple graph. If the graph that we obtain from G by removing an edge e is disconnected, then we say that e is a bridge of G. Prove that an edge e of G is a bridge if and only if it is in every spanning tree of G.*

Department of Mathematics, MIT, Cambridge, MA 02139
Email address: fgotti@mit.edu