MATH 409 LECTURE 6
MINIMUM SPANNING TREES

REKHA THOMAS

In this lecture we look at the problem of finding a minimum cost spanning tree in a graph. You can read about this problem in either Chapter 6 of [2] or Chapter 2 of [1].

We start by establishing some basic properties of a spanning tree. Assume throughout the next few lectures that G is a connected graph. Recall that a spanning tree T of G is a spanning connected acyclic subgraph of G.

Exercise 1. Let T be a spanning tree of G, e be an edge of T and consider the graph $T\backslash\{e\}$ obtained by deleting the edge e from T. Prove that $T\backslash\{e\}$ is a disconnected graph consisting of two connected components $T_1 = (V_1, E_1)$ and $T_2 = (V_2, E_2)$ such that T_1 and T_2 are spanning trees on the subgraphs of G induced by V_1 and V_2 respectively.

Lemma 2. *Let T be a spanning connected subgraph of a graph G with n vertices. Then T is a spanning tree of G if and only if T has $n-1$ edges.*

Proof. This involves proving $P \Leftrightarrow Q$ where P is the statement T *is a spanning tree of* G and Q is the statement T *has* $n-1$ *edges*. In both statements we are allowed to add on the fact that T is a spanning connected subgraph of G. This addition is implicit in P but is a crucial addition needed for Q if the lemma is to be true.

($Q \Rightarrow P$): Suppose T is a spanning connected subgraph of G with $n-1$ edges. We need to show that T is acyclic which we will prove by the method of contradiction. So suppose there is a circuit C in T using the vertices $v_1, v_2, \ldots, v_k$. This circuit contains k edges which are all in T. There are $n-k$ vertices remaining in G and each of them is touched by at least one edge of T since T is a spanning and connected subgraph of G. None of these edges can come from the circuit C since the remaining $n-k$ vertices are not vertices of C. Therefore, T has at least $n-k$ more edges which implies that T has at least $k+(n-k) = n$ edges which contradicts our assumption Q. Therefore, T is acyclic and is a spanning tree of G.

Date: April 9, 2010.

($P \Rightarrow Q$): We prove this by induction on $n = |V(G)|$.

Base case: $n = 1$: If G has only one vertex, then $T = G$ and $|E(T)| = 0 = 1 - 1$. Therefore the implication is true when $n = 1$.

Induction hypothesis: Assume that $P \Rightarrow Q$ whenever $|V(G)| \leq k - 1$. (This is strong induction.)

Induction step: We need to argue that when T is a spanning tree in a graph G with $|V(G)| = k$ then T has $k - 1$ edges. Pick an edge $e \in T$ and consider the graph $T \backslash \{e\}$ obtained by deleting the edge e from T. By Exercise 1, $T \backslash \{e\}$ has two components $T_1 = (V_1, E_1)$ and $T_2 = (V_2, E_2)$ which are spanning trees on the subgraphs of G induced by V_1 and V_2 respectively. Since $|V_1|, |V_2| \leq k - 1$, by our induction hypothesis, T_1 has $|V_1| - 1$ edges and T_2 has $|V_2| - 1$ edges. Since T is obtained by connecting T_1 and T_2 by e, we conclude that T has $|V_1| - 1 + |V_2| - 1 + 1 = |V| - 1 = k - 1$ edges. $\square$

Exercise 3. Let $T = (V, E(T))$ be a spanning tree of G, let $e = \{i, j\}$ be an edge of G but not of T, and $f = \{k, l\}$ be an edge of a path in T from i to j. Then prove that $T' = (V, (E(T) \backslash \{k, l\}) \cup \{i, j\})$ is a spanning tree of G. **Hint:** Does there exist two different paths in a spanning tree T between two different vertices?

The number of spanning trees in K_n is n^{n-2}. How does this number compare to the number of traveling salesman tours in K_n. Which is larger?

Lemma 4. *Let T be a spanning tree in G and $e \in E(T)$. Let C be one of the components of $T \backslash \{e\}$ and define*

$$\delta(V(C)) := \{f = \{x, y\} \in E(G) : \text{either } x \text{ or } y \in V(C) \text{ but not both}\}.$$

In other words $\delta(V(C))$ is the collection of edges leaving C. By construction, $e \in \delta(V(C))$. Then $\delta(V(C))$ does not have any other edges of T besides e.

Proof. We will prove this lemma by the method of contradiction. Suppose $\delta(V(C))$ contains a second edge of T — call it h. Let $e = \{i, j\}$ and $h = \{k, l\}$. Let C and C' be the two components of $T \backslash e$. We may assume that $i, k \in C$ and $j, l \in C'$. Then there is an (i, k)-path P_1 in C since C is a spanning tree on the vertices of G in C and there is also a (j, l)-path P_2 in C'. This implies that P_1, e, P_2, h form a circuit of G that lies in T which contradicts that T is acyclic. Therefore we conclude that our assumptions are wrong. The only part of the assumption that could be wrong is our assumption that the conclusion of the lemma was false (since the hypothesis of the lemma cannot be false unless this lemma is actually entirely false). Thus there is only one edge of T in $\delta(V(C))$. $\square$

Suppose we have a connected graph $G = (V, E)$ and edge costs $c_e \in \mathbb{R}$ for all edges $e \in E$. The **cost of a subgraph** H of G is the sum $\sum\{c_e : e \in H\}$. We will denote this sum as $c(H)$.

Definition 5. The Minimum Spanning Tree (MST) Problem: Find the minimum cost spanning tree in a connected graph $G = (V, E)$ with edge costs $c_e \in \mathbb{R}$ for all $e \in E$.

Exercise 6. Show that any MST problem can be reduced to an MST problem with positive edge costs.

We will describe and analyze two algorithms to solve this problem. Both algorithms rely on the following theorem.

Theorem 7. *Let T be a spanning tree in a connected graph G with edge costs c_e for all $e \in E(G)$. Then the following statements are equivalent:*

(1) *T is a minimum spanning tree (MST).*
(2) *For all $f = \{x, y\} \in E(G)\backslash E(T)$ no edge on the (x, y)-path in T has higher cost than f.*
(3) *For all $e \in E(T)$, e is a minimum cost edge of $\delta(V(C))$ where C is a connected component of $T\backslash e$.*

Proof. We will show that (1) $\Rightarrow$ (2) $\Rightarrow$ (3) $\Rightarrow$ (1).

(1) $\Rightarrow$ (2) : We prove this by contradiction. Assume that T is a MST and that (2) is false – i.e., there exists an edge e in the (x, y)-path in T with $c_e > c_f$, where $f = \{x, y\} \in E(G)\backslash E(T)$. By Lemma 3, $T' = (V, E(T) \cup \{f\}\backslash\{e\})$ is also a spanning tree of T. However, T' has lower cost than T since $c_e > c_f$ which contradicts that T was an MST. Therefore, we conclude that (1) $\Rightarrow$ (2).

(2) $\Rightarrow$ (3) : We prove this by proving the contrapositive statement which is that $\sim$ (3) $\Rightarrow\sim$ (2). Suppose (3) is false — i.e., there exists $f = \{x, y\} \in \delta(V(C))$ such that $c_f < c_e$. By Lemma 4, $f \notin T$. (Otherwise, both e and f would be edges in T that lie in $\delta(V(C))$ which contradicts Lemma 4.) But e lies on the (x, y)-path in T and $c_e > c_f$ which violates (2). Thus $\sim$ (3) $\Rightarrow\sim$ (2) which is equivalent to (2) $\Rightarrow$ (3).

(3) $\Rightarrow$ (1) : We prove this last implication by a direct argument. So assume (3) is true for a spanning tree T. Let T^* be a MST of G

such that $E(T) \cap E(T^*)$ is as large as possible. We will show that $T = T^*$ which will establish (1).

Suppose there exists $f = xy \in E(T)\backslash E(T^*)$. Then adding f to T^* creates a circuit D. Let C be a connected component of $T\backslash f$. Then there exists some edge $g \in D \cap E(T^*)$ such that g also lies in $\delta(V(C))$. Since T^* is a MST, $c_g \leq c_f$. Since T satisfies (3), $c_f \leq c_g$ which implies that $c_f = c_g$. (Note that $g \notin T$.)

Consider $T' = (V(G), E(T^*) \cup \{f\}\backslash\{g\})$. By Lemma 3, T' is a spanning tree of G. Since $c(T') = c(T^*)$, T' is actually an MST of G. However, T' has one more edge in common with T than T^* which contradicts our choice of T^*. This implies that there does not exist $f \in E(T)\backslash E(T^*)$ which implies that $T = T^*$.

$\square$

References

[1] W. Cook, W. Cunningham, W. Pulleyblank, and A. Schrijver. *Combinatorial Optimization*. Wiley-Interscience Series in Discrete Mathematics, 1998.

[2] B. Korte and J. Vygen. *Combinatorial Optimization*. Springer, Berlin, 2000.