Math 425 Assignment 1

Max Horowitz-Gelb

1

2

Lemma A:

Statement: A connected subgraph of n nodes requires at least $n-1$ edges.

Proof: Consider a graph $G = (V, E)$.

Let G_n be the set of all connected subgraphs of G with n nodes. And let G'_n be a subgraph of G_n such that the number of edges in G'_n is minimized. Then let $E'_n = E(G'_n)$ and $V'_n = V(G'_n)$.

First we shall consider the base case of $n = 1$ nodes. Here it is trivial to see by definition of a graph that $E'_1 = 0$.

Now assume that $E'_b = \alpha$. Then it must be true that $E'_{b+1} \geq \alpha + 1$ since for any subgraph in G_b, by definition of a graph, requires at least one new edge to connect a new node.

Therefore $E'_n \geq n + 1$, hence proving our statement. ■

Lemma B:

Statement: For a spanning tree H, there exists a node v such that $v \in H$ and v has a degree of 1

Proof: Clearly no node can have a degree of 0 or else H would not be connected and therefore not be a tree. So then it is suffice to show that if all nodes in H have a degree greater than 1, then that implies that H has a cycle and therefore is not a tree.

To show this, consider a simple traversal algorithm. In this algorithm one starts at an arbitrary node in H and then traverses an unused edge until it can no longer move because it is stuck at a node with all used edges. If at any point in this algorithm it hits a node already visited then there must be a cycle in the graph since it would imply that there is a path using unique edges from a node back to itself. If this algorithm were run on H it would have to hit a node a second time. This is because for the algorithm to not find a cycle it would have to never hit a node twice and then stop at a final unvisited node because it was stuck. But it can't get stuck at such a node since the degree of all nodes is greater than 1. Therefore there must exist a node in H such that its degree is 1. ■

Lemma C:

Statement: If H is a tree with at least 2 nodes then there exists a node v such that $v \in H$, $H \backslash v$ is a tree and $|E(H)| = |E(H \backslash v)| + 1$

Proof: By Lemma B there must be a node $v \in H$ with a degree of 1. H is a tree and therefore has no cycles so $H \backslash v$ has no cycles either. And since v has a degree of 1, by definition there can be no path in H between any two nodes in $H \backslash v$ that can include the node v. Then since H is connected, by definition $H \backslash v$ is connected. Therefore since $H \backslash v$ is connected and has no cycles it is also a tree. And since v has a degree of 1, then clearly $|E(H)| = |E(H \backslash v)| + 1$. ■

Lemma 2.2

Statement: Let G be a connected graph with n nodes. Then G is a spanning tree if and only if it has exactly $n - 1$ nodes.

Proof: First we shall prove that if G is a spanning tree then it has 1 less edges than nodes.

Let S_n be the set of all spanning trees of all subgraphs of G containing n nodes where a spanning tree is possible. Then let $|E_n|$ be the number of edges of used for each subgraph in S_n assuming they are all the same.

Clearly $|E_1| = 0$ by definition of a graph.

Then if $S_n = n - 1$ then $S_{n+1} = n$. This is because Lemma C implies for all $H \in S_{n+1}$ there exists a node $v \in H$ such that $H \backslash v \in S_n$ and as well $|E_{n+1}| = |E_n| + 1$.

Therefore by induction G is a spanning tree of n nodes then it contains $n - 1$ edges.

Now we shall show that if G is a connected graph with n nodes and $n - 1$ edges then it is a tree.

If G were connected and not a tree then it must have a cycle. If it had a cycle then there must be an edge $e = (u, v)$ such that there is a path without e from u to v and clearly e can be removed from G without breaking the connectivity of G. But this would imply that there is a connected graph with $n - 2$ edges and n nodes with breaks Lemma A. Therefore G cannot have any cycles and must be a tree. ■

3

Lemma D

Statement: If H is a spanning tree of graph G then there is one unique simple path between all pairs of nodes in H.

Proof: First we shall show that all edge simple paths between nodes are simple paths. If there was an edge simple path between two nodes in H that was not simple, then there is a portion of that path $v_i, e_i, ...e_{j-1}, v_j$ such that $v_i = v_j$. This would imply that there is a cycle in H which cannot be true since H is a tree. Therefore all edge simple paths in H are also simple.

For any two nodes in H there is at least one path between them since H is a tree. If that path is not edge simple then it can be transformed into an edge simple path by removing all loops $e_i, ...e_{j-1}, v_j$ such that $v_i = v_j, e_i = e_{j-1}$. And as we have shown that edge simple path must be simple. Therefore there is at least one simple path between all pairs of nodes in H.

If there were more than one simple path between two nodes then select two of them and simply remove from both paths $e_k...v_n$ such that both removals are equal and remove $v_0...e_i$ such that i is maximized and less than k and v_{i+1} is the same for both paths. When concatenating both paths you would create a cycle which cannot be true since H is a tree.

Therefore for any pair of nodes in H there is only one edge simple path between them and it is simple.

■

Lemma 2.7

Statement: Let $H = (V, T)$ be a spanning tree of $G = (V, E)$. Let $e = vw$ be an edge in $E \backslash T$, and let f be en edge of a simple path in H from v to w. Then

(a) the subgraph H' obtained from adding e has a unique cycle containing e, and (b) the subgraph $H'' = (V, T \cup \{e\} \backslash \{f\})$ is a spanning tree of G.

Proof of (a): First we can clearly see that since H is a spanning tree then there is a path from v to w and then adding e creates a cycle by definition. If adding e created more than one unique cycle, that would imply that there exists at least two unique edge simple paths connecting v to w in H. This cannot be true since it would mean that there was a cycle in H which is a tree. Therefore e only introduces one unique cycle. ∎

Proof of (b)

First e cannot introduce a cycle because if it did that would imply that there was a simple path from v to w in $H \backslash \{f\}$. But this cannot be true since by Lemma D, removing f breaks the one and only simple path between v and w.

Again by Lemma D only all pairs of nodes in H connected by a simple path containing f will be disconnected in $H \backslash \{f\}$. Since v and w are connected in H using a simple path containing f, then all the disconnected nodes in $H \backslash \{f\}$ must be connected to either v or w. Then all nodes in $H \backslash \{f\}$ are connected to v and or w. Therefore since in $H \cup \{e\} \backslash \{f\}$ v and w are connected, then all pairs of nodes are connected.

Therefore H'' is a spanning tree. ∎

4

Lemma E

Statement: Let H be a spanning tree of some graph G. Then let $e = (u, v)$ be some edge in H. Finally let A be the set of all nodes in H connected by a simple path including e to u. Then A is disconnected from $H \backslash A$ in the subgraph $H \backslash \{e\}$.

Proof: If the above statement were not true, this would imply that the $|\delta(A)|$ of H was greater than 1. But this can't be true because then there would be more than one unique edge simple paths between pairs of nodes in H which breaks lemma D. ∎

Exercise 2.9

Statement: Let $G = (V, E)$ be a connected graph with costs $c_e, e \in E$. If $H = (V, T)$ is a MST of G, and $e \in T$ then there is a cut D of G with $e \in D$

and $c_e = min\{c_f : f \in D\}$.

Proof: Let $e = (u, v)$ be given as described in the statement. Then let A be the set of nodes connected to u in H via a simple path using e. Then let $D = \delta_G(A)$. Then by Lemma E if $a \in D$ then either $a = e$ or $a \notin T$. Therefore $c_e = min\{c_f : f \in D\}$ since if this were not true, then we could create a spanning tree $H \cup \{f\} \backslash \{e\} : c_f < c_e$. This would be a spanning tree with less weight than H which cannot be true since H is a MST.