In the last lecture a primal-dual analysis was used to show a 2-approximation for the VERTEX COVER problem. Here we extend these ideas to show a 2-approximation for the STEINER FOREST problem using a primal-dual analysis. In the STEINER FOREST problem there is a graph $G = (V, E)$ where each edge $e \in E$ has a cost $c_e \in \mathbb{R}$. We are given k pairs of terminals $s_1, t_1, s_2, t_2, \ldots s_k, t_k \in V$. The goal is to find the minimum cost set of edges F^* such that in the graph $G[F^*]$ the vertices s_i and t_i are in the same connected component for $1 \leq i \leq k$. Here $G[F^*]$ denotes the induced subgraph G on the set of edges F^*. Notice that $G[F^*]$ can contain multiple connected components.

We now want to transform this problem into a **IP**. First we set up some notation. In the **IP** we will have a variable x_e for each edge $e \in E$ such that x_e is 1 if and only if e is part of the solution. Let the set $\mathcal{S}$ be the collection of all sets $S \subset V$ such that S disconnects s_i and t_i for some $1 \leq i \leq k$. For a set $S \subset V$ let $\delta(S)$ denote the set of edges crossing the cut $(S, V \setminus S)$. The **IP** can be written as the following.

$$\begin{aligned} \min \quad & \sum_{e \in E} c_e x_e \\ \text{such that} \quad & \sum_{e \in \delta(S)} x_e \geq 1 \quad \forall S \in \mathcal{S} \\ & x_e \in \{0, 1\} \quad \forall e \in E \end{aligned}$$

We can obtain an **LP**-relaxation by changing the constraint that $x_e \in \{0, 1\}$ to $x_e \geq 0$. The dual of the **LP**-relaxation can be written as the following.

$$\begin{aligned} \max \quad & \sum_{S \in \mathcal{S}} y_S \\ \text{such that} \quad & \sum_{S : e \in \delta(S)} y_S \leq c_e \quad \forall e \in E \\ & y_S \geq 0 \quad \forall S \in \mathcal{S} \end{aligned}$$

Before we continue, some definitions will be stated which will help to define our algorithm for the problem.

Definition 1 *Given a set of edges $X \subseteq E$, a set $S \in \mathcal{S}$ is* violated *for X if $\delta(S) \cap X = \emptyset$.*

Definition 2 *Given a set of edges $X \subseteq E$, a set $S \in \mathcal{S}$ is* minimally violated *if S is violated for X and there is no $S' \subset S$ that is also violated for X.*

Next we show that any two minimally violated sets are disjoint.

Claim 3 *$\forall X \subseteq E$ if S and S' are minimally violated sets then $S \cap S' = \emptyset$, i.e. S and S' are disjoint.*

Proof: To prove this claim, we will show that the minimally violated sets are connected components in the graph $G[X]$. Consider a minimal violated set S; w.l.o.g., we may assume the set S contains s_i but not t_i for some i. If S is not connected then there must be some connected component S' of S that contains s_i. But $\delta(S') \leq \delta(S) = 0$, and hence S' is violated; this contradicts the fact that S is minimal. Therefore, if a set S is a minimal violated set then it must be connected in the graph $G[X]$.

Now suppose that S is a connected component of $G[X]$; it is easy to see that no proper subset of S can be violated. (Since S is connected, there must be an edge from any $S' \subset S$ to $S - S'$; therefore, $\delta(S') \geq 1$.) Hence, any two minimally violated sets S and S' are connected components in $G[X]$ and they must have an empty intersection. □

We now give our algorithm for Steiner Forest:

```
SteinerForest:
  F ← ∅
  while F is not feasible
      Let C_1, C_2, ..., C_h be minimally violated sets with respect to F
      Raise y_{C_i} for 1 ≤ i ≤ h uniformly until some edge e becomes tight
      F ← F + e
      x_e = 1
  Output F' = {e ∈ F |  F − e  is not feasible}
```

The first thing to notice about the algorithm above is that it is closely related to our solution to the Vertex Cover problem, however, there are two main differences. In the Vertex Cover we raised all of the dual variables uniformly, however, in this algorithm we were more careful on which dual variables were raised. In this algorithm, we chose to only raise the variables which correspond to the minimally violated sets. This is because we do not want to raise a dual variable y_S corresponding set S without knowing the value of the variable $y_{S'}$ for any $S' \subset S$. The other main difference is that when we finally output the solution, we *prune* F to get F'. This is done for technical reasons, but the intuition is that we should include no edge in the solution which is not needed to obtain a feasible solution. To understand this algorithm, there is a non-trivial example in the textbook [1] on page 200 that demonstrates the algorithm's finer points.

Lemma 4 *At the end of the algorithm, F' and $\mathbf{y}$ are primal and dual feasible solutions, respectively.*

Proof: In each iteration of the while loop, only the dual variables corresponding to connected components were raised. Therefore, no edge that is contained within the same component can become tight, and, therefore, F is acyclic. To see that none of the dual constraints are violated, observe that when a constraint becomes tight (that is, it holds with equality), the corresponding edge e is added to F. Subsequently, since e is contained in some connected component of F, no set S with $e \in \delta(S)$ ever has y_S raised. Therefore, the constraint for e cannot be violated, and so $\mathbf{y}$ is dual feasible.

As long as F is not feasible, the while loop will not terminate, and there are some minimal violated sets that can have their dual variables raised. Therefore, at the end of the algorithm F is feasible. Moreover, since F is acyclic (it is a forest), there is a unique s_i-t_i path in F for each $1 \leq i \leq k$. Thus, each edge on a s_i-t_i path is not redundant and is not deleted when pruning F to get F'. □

Theorem 5 *The primal-dual algorithm for* Steiner Forest *gives a 2-approximation.*

Proof: Let F' be the output from our algorithm. To prove this theorem, we want to show that $c(F') \leq 2\sum_{s\in\mathcal{S}} y_S$ where y_S is the feasible dual constructed by the algorithm. It follows from this that the algorithm is in fact a 2-approximation. First, we know that $c(F') = \sum_{e\in F'} c_e = \sum_{e\in F'}\sum_{S\in\mathcal{S}:e\in\delta(S)} y_s$ because every edge picked is tight. Let $\deg_{F'}(S)$ denote the number of edges of F' that cross the cut $(S, V\setminus S)$. It can be seen that $\sum_{e\in F'}\sum_{S\in\mathcal{S}:e\in\delta(S)} y_s = \sum_{s\in\mathcal{S}} y_S \deg_{F'}(S)$.

Let A_i contain the minimally violated sets in iteration i and let Δ_i denote the amount of dual growth in the ith iteration. Say that our algorithm runs for α iterations. We can then rewrite $\sum_{s\in\mathcal{S}} y_S \deg_{F'}(S)$ as the double summation $\sum_{i=1}^{\alpha}\sum_{S\in A_i}\Delta_i \deg_{F'}(S)$. In the next lemma it will be shown for any iteration i that $\sum_{S\in A_i}\deg_{F'}(S) \leq 2|A_i|$. Knowing this we can prove the theorem:

$$\sum_{S\in\mathcal{S}} y_s \deg_{F'}(S) = \sum_{S\in\mathcal{S}}\sum_{i:S\in A_i}\Delta_i \deg_{F'}(S) \quad = \quad \sum_{i=1}^{\alpha}\sum_{S\in A_i}\Delta_i \deg_{F'}(S) \leq \sum_{i=1}^{\alpha}\Delta_i\cdot 2|A_i| \leq 2\sum_{S\in\mathcal{S}} y_S.$$

□

Now we show the lemma used in the previous theorem. It is in this lemma that we use the fact that we prune F to get F'.

Lemma 6 *For any iteration i of our algorithm,* $\sum_{S\in A_i}\deg_{F'}(S) \leq 2|A_i|$

Proof: Consider the graph $G[F']$, and fix an iteration i. In this graph, contract each set S active in iteration i to a single node (call such a node an *active node*), and each inactive set to a single node. Let the resulting graph be denoted by H. We know that F is a forest and we have contracted connected subsets of vertices in F; as $F'\subseteq F$, we conclude that H is also a forest.

The degree in H of an active node corresponding to violated set S is $\deg_{F'}(S)$. The average degree of the vertices in a forest is less than 2. However, this does not complete the lemma because it maybe the case that the active nodes in H have average degree greater than 2 and the inactive nodes have degree less than 2. Fortunately, this cannot be the case and we will show that all inactive nodes have degree at least two. This will then show that the average degree of all active nodes is at most 2 and complete the lemma.

Suppose an inactive node of H corresponding to an inactive component C has degree less than 2; it must be a leaf of the forest H. There is a unique edge $e\in F'$ that crosses the cut $(C, V\setminus C)$. We know that F was pruned to get F' so that F' has no redundant edges, so there must exist two terminals s_i and t_i such that $s_i\in C$ and $t_i\in V\setminus C$. However, then component C is violated in iteration i, which is a contradiction. □

References

[1] V. V. Vazirani. *Approximation Algorithms*, Springer-Verlag, New York, NY, 2001.