CS880: Approximations Algorithms	
Scribe: Chi Man Liu	**Lecturer:** Shuchi Chawla
Topic: Primal-Dual Method: Steiner Forest and Facility Location	**Date:** 3/8/2007

Last time we discussed how LP-duality can be applied to approximation. We introduced the primal-dual method for approximating the optimal solution to an LP in a combinatorial way. We applied this technique to give another 2-approximation for the Vertex Cover problem. In this lecture, we present and analyze a primal-dual algorithm that gives a 2-approximation for the Steiner Forest problem. We also briefly talk about a primal-dual algorithm for Facility Location.

14.1 Steiner Forest

We restate the Steiner Forest problem here.

Definition 14.1.1 (Steiner Forest) *Given an undirected graph $G = (V, E)$, edge costs $c : E \to \mathbb{R}^+$, and disjoint subsets $S_i \subseteq V$, find a minimum-cost forest F such that for all i and $u, v \in S_i$, there exists a path connecting u to v in F.*

To formulate Steiner Forest as an ILP, we introduce an indicator variable x_e for each edge e. Let $\mathcal{S}$ be the collection of all sets $S \subseteq V$ that cut some S_i into two parts, i.e. $\mathcal{S} = \{S \in V \mid S \cap S_i \notin \{\emptyset, S_i\} \text{ for some } i\}$. Then from last lecture we have the following ILP formulation for Steiner Forest.

$$\begin{array}{rll} \text{minimize} & \sum_{e\in E} c_e x_e & \\ \text{subject to} & \sum_{e\in\delta(S)} x_e \geq 1 & \forall S \in \mathcal{S} \\ & x_e \in \{0,1\} & \forall e \in E \end{array}$$

where $\delta(S)$ denotes the boundary of S, i.e. the set of edges in E with exactly one endpoint in S. The LP relaxation for this problem is as follows. Note that we do not restrict x_e to be at most 1 since any optimal solution x^* to the LP must satisfy $x^*_e \leq 1$ for all $e \in E$.

$$\begin{array}{rll} \text{minimize} & \sum_{e\in E} c_e x_e & \\ \text{subject to} & \sum_{e\in\delta(S)} x_e \geq 1 & \forall S \in \mathcal{S} \\ & x_e \geq 0 & \forall e \in E \end{array}$$

We introduce a variable y_S for each $S \in \mathcal{S}$ and obtain the dual LP.

$$\begin{array}{rll} \text{maximize} & \sum_{S\in\mathcal{S}} y_S & \\ \text{subject to} & \sum_{S:e\in\delta(S)} y_S \leq c_e & \forall e \in E \\ & y_S \geq 0 & \forall S \in \mathcal{S} \end{array}$$

If $c_e = 1$ for all e and integer constraints are imposed, the dual LP can be interpreted as the problem of picking a largest collection $\mathcal{C} \subseteq \mathcal{S}$ such that no two sets in $\mathcal{C}$ have a common edge on their boundaries.

We start with the primal infeasible solution $x = (0, \ldots, 0)^T$ and the dual feasible solution $y = (0, \ldots, 0)^T$. We then raise some of the dual variables until some dual constraint goes tight. At this point, we put the edge corresponding to the constraint into F, and freeze all dual variables in the constraint. We repeat this process, all the time keeping the dual solution feasible, until all dual variables are frozen. A detailed description of our primal-dual algorithm is as follows.

1. Set $x_e = 0$ for all $e \in E$, and $y_S = 0$ for all $S \in \mathcal{S}$.
2. Raise uniformly the dual variables y_S's corresponding to all **minimal** unsatisfied sets.
3. When a dual constraint goes tight, assign the value 1 to the primal variable x_e corresponding to the constaint, and freeze all dual variables y_S in the constraint. Without loss of generality, we assume that at most one dual constraint goes tight at any point in time.
4. Repeat the above two steps until all y_S are frozen.

After the above primal-dual steps, we get a collection of edges F. Finally, we need a pruning step to remove extra edges from F.

5. For all edges $e \in F$ such that $F \backslash \{e\}$ is still feasible, remove e from F. We denote the resulting collection of edges by F'.

Intuitively, we can think of the algorithm as expanding minimal unsatisfied sets (*active sets*) until some of them touch on some edge, at which time we merge the active sets to form a larger set. The newly formed set may be active or inactive. If it is active, we start expanding it with other existing active sets. Initially, each terminal node forms an active set. The above process continues until all sets become inactive. Also note that when we expand an active set, it may touch a non-Steiner node. In such a case, a new set is formed by adding that node into the active set.

Lemma 14.1.2 *At termination, y is dual feasible.*

Proof: In our algorithm, whenever a dual constraint goes tight, all its variables are frozen, therefore no dual constraints can be violated throughout the algorithm. Since all dual constraints are satisfied at the beginning, they remain satisfied. ■

By weak duality, we have the following corollary.

Corollary 14.1.3 *Let OPT be the cost of an optimal Steiner forest. Then $\sum_{S \in \mathcal{S}} y_S \leq OPT$.*

Lemma 14.1.4 *At the end of the primal-dual steps, F is a forest and is primal feasible.*

Proof: Since we consider dual constraints one by one, and always add edges between two minimal unsatisfied sets, we never form a cycle, and thus F is a forest. If F was not feasible, then some $S \in \mathcal{S}$ remained unsatisfied upon termination. That means none of the edges in $\delta(S)$ were in F; equivalently, y_S had not been frozen. This contradicts step 4 of our algorithm. ■

Lemma 14.1.5 *F and y satisfy the primary complementary slackness conditions.*

Proof: We only raise the value of a primal variable in step 3 of our algorithm. Such an event happens only when the corresponding dual constraint goes tight. Hence, $x_e > 0$ implies that $\sum_{S: e \in \delta(S)} y_S = c_e$. ■

In view of Corollary 14.1.3 and Lemma 14.1.5, we may want to show that $y_S > 0$ implies $\sum_{e \in \delta(S)} x_e \leq 2$. Then this would give

$$\sum_{e \in F} c_e = \sum_{e \in F} \sum_{S: e \in \delta(S)} y_S = \sum_{S \in \mathcal{S}} \sum_{e \in F \cap \delta(S)} y_S = \sum_{S \in \mathcal{S}} y_S \sum_{e \in \delta(S)} x_e \leq 2 \sum_{S \in \mathcal{S}} y_S \leq 2OPT.$$

However, this is not true in general, because average degree of the sets can be high when the sets become inactive. That is why we need the pruning step to remove extra degrees. We now shift our attention to the pruned solution F'.

Lemma 14.1.6 *F' is a forest and is primal feasible.*

Proof: F' is a forest follows from the fact that its superset F is forest. To show that F' is feasible, pick an i and $u, v \in S_i$. Since F is feasible and is a forest, there exists a unique path in F connecting u to v. Removing any edge in the path would make the solution infeasible, therefore the whole path must be contained in F'. ■

Lemma 14.1.7 *F' and y satisfy the primary complementary slackness conditions.*

Proof: This is a direct corollary of Lemma 14.1.5. ■

Lemma 14.1.8 *Let OPT be the cost of an optimal Steiner forest. Then*

$$\sum_{e \in F'} c_e \leq 2 \sum_{S \in \mathcal{S}} y_S \leq 2OPT.$$

Proof: By Lemma 14.1.7,

$$\sum_{e \in F'} c_e = \sum_{e \in F'} \sum_{S: e \in \delta(S)} y_S.$$

Rewriting the summations on the right-hand side, we get

$$\begin{aligned} \sum_{e \in F'} c_e &= \sum_{S \in \mathcal{S}} \sum_{e \in F' \cap \delta(S)} y_S \\ &= \sum_{S \in \mathcal{S}} y_S \cdot deg_{F'}(S) \end{aligned}$$

where $deg_{F'}(S)$ is the number of edges in F' crossing the boundary of S, i.e. $|F' \cap \delta(S)|$. Our goal is to show that $\sum_{S \in \mathcal{S}} y_S \cdot deg_{F'}(S) \leq 2 \sum_{S \in \mathcal{S}} y_S$. We use an inductive argument to show that the inequality holds at any point in time. The base case ($t = 0$) is trivial since $\sum_{S \in \mathcal{S}} y_S = 0$. For the inductive part, we assume that the inequality is satisfied at time t. Let $\Delta > 0$ be an amount such that no dual constraint goes tight in the time interval $[t, t + \Delta)$. Let $\mathcal{A}(t)$ be the collection of all active sets at time t. Then, the increase in the left-hand side of the inequality within $[t, t + \Delta]$ is

$$\Delta \cdot \sum_{S \in \mathcal{A}(t)} deg_{F'}(S),$$

whereas the increase in the right-hand side is

$$2\Delta \cdot |\mathcal{A}(t)|.$$

We claim that any time t, the average degree of all active sets is at most 2, i.e. $\sum_{S\in\mathcal{A}(t)} deg_{F'}(S) \leq 2\cdot|\mathcal{A}(t)|$. If that holds, then the increase in LHS is at most the increase in RHS, thus completing our induction. To prove the claim, we consider all components of F at time t. The average degree of these components under the final F is at most 2, implying that the average degree of these components under F' is at most 2. Some of these components are active sets while others are inactive. If we can show that all the inactive ones have degree (under F') not equal to 1, then we are done. We prove this by contradiction. Suppose that for some inactive component S, $deg_{F'}(S) = 1$. Let e be the only edge in $F' \cap \delta(S)$. Then e must lie on some path in F' connecting $u, v \in S_i$ for some i, otherwise it would have been removed from F'. But then $u \in S$ and $v \notin S$ (or vice versa), implying that $S \in \mathcal{S}$, which is a contradiction since we have assumed that S is inactive. ■

Remark. The above 2-approximation is due to Agarwal, Klein and Ravi [1]. Later, Goemans and Williamson [2] started to use the primal-dual method in solving many other approximation problems.

14.2 Facility Location

We restate the Facility Location problem here.

Definition 14.2.1 (Facility Location) *Given a set of facilities I, a set of customers J, facility opening costs $f : I \to \mathbb{R}^+$ and a metric c on $I \cup J$, find a subset $S \subseteq I$ that minimizes the total cost $C(S) = C_f(S)+C_r(S)$, where $C_f(S)$ is the total facility openning cost defined as $C_f(S) = \sum_{i\in S} f(i)$, and $C_r(S)$ is the total routing cost defined as $C_r(S) = \sum_{j\in J} \min_{i\in S} c(i, j)$.*

To formulate Facility Location as an ILP, we introduce an indicator variable x_i for each facility i. For each facility i and customer j, we introduce a variable y_{ij} which is set to 1 if and only if j is assigned to i. For notational convenience, we write f_i and c_{ij} instead of $f(i)$ and $c(i, j)$. Then we have the following ILP formulation for Facility Location.

$$\begin{array}{lll} \text{minimize} & \sum_{i\in I} f_i x_i + \sum_{i\in I, j\in J} c_{ij} y_{ij} & \\ \text{subject to} & \sum_{i\in I} y_{ij} \geq 1 & \forall j \in J \\ & x_i - y_{ij} \geq 0 & \forall i \in I, j \in J \\ & x_i, y_{ij} \text{ are non-negative integers} & \forall i \in I, j \in J \end{array}$$

By introducing a variable α_j for each customer j and β_{ij} for each facility-customer pair (i, j), we get the following dual of the LP relaxation.

$$\begin{array}{lll} \text{maximize} & \sum_j \alpha_j & \\ \text{subject to} & \alpha_j - \beta_{ij} \leq c_{ij} & \forall i \in I, j \in J \\ & \sum_j \beta_{ij} \leq f_i & \forall i \in I \\ & \alpha_i, \beta_{ij} \geq 0 & \forall i \in I, j \in J \end{array}$$

We can interpret the dual LP as follows. We are the owners of the facilities and are going to collect money from the customers in order to open some of the facilities. For each customer j, α_j is the amount paid by j. For each facility i and customer j, β_{ij} is the portion of α_j that pays for facility i. The constraint $\alpha_j - \beta_{ij} \leq c_{ij}$ states that the amount paid by customer j should not exceed the portion that goes to facility i plus the routing cost c_{ij}, for every facility i. The constraint $\sum_j \beta_{ij} \leq f_i$ states that no facility should overcharge its customers, i.e. the customers should not pay more than enough to open the facility. Under these constraints, we want to maximize $\sum_j \alpha_j$, the total amount collected from the customers. Note that each customer may be paying for several facilities. To deal with this, we close some of the facilities in our primal solution such that every customer pays for at most one facility, and reassign all unassigned customers to the remaining facilities. We will look at the primal-dual algorithm next lecture.

References

[1] A. Agrawal, P. Klein and R. Ravi. When Trees Collide: An Approximation Algorithm for the Generalized Steiner Problem on Networks. In *SIAM Journal on Computing*, 24, pp. 440–456, 1995.

[2] M. X. Goemans and D. P. Williamson. A General Approximation Technique for Constrained Forest Problems. In *SIAM Journal on Computing*, 24, pp. 296–317, 1995.