Lecture 8

CSE 522: Advanced Algorithms

October 25, 2004
Lecturer: Kamal Jain
Notes: Ning Chen

1 Prize-Collecting Problem Review

Given a graph $G = (V, E)$, a non-negative cost function $c : E \to Q_+$, and a non-negative penalty function $\pi : V \times V \to Q_+$, our goal is minimum-cost way of buying a set of edges and paying the penalty for the edges which are not connected via bought edges.

For a set $S \subset V$, denote $|S \cap \{i, j\}| = 1$ by $S \odot (i, j)$. Denote a family of subsets of V by $\mathcal{S} = \{S_1, \ldots, S_k\}$. For a family $\mathcal{S}$, we say $\mathcal{S} \odot (i, j)$ if there is an $S \in \mathcal{S}$ such that $S \odot (i, j)$. Let $\delta(S) = \{(u, v) \mid u \in S \text{ or } v \in S\}$. Therefore, our problem can be written as the following LP relaxation:

$$
\begin{aligned}
\min \quad & \sum_{e \in E} c_e x_e + \sum_{i,j \in V} \pi_{i,j} z_{i,j} \\
s.t. \quad & \sum_{e \in \delta(S)} x_e + z_{i,j} \geq 1, \quad \forall\ S \subset V, (i, j) \in V \times V, \mathcal{S} \odot (i, j) \\
& x_e \geq 0, \quad \forall\ e \in E \\
& z_{i,j} \geq 0, \quad \forall\ (i, j) \in V \times V
\end{aligned}
$$

The dual program of the above LP is:

$$
\begin{aligned}
\max \quad & \sum_{S \subset V, S \odot (i,j)} y_{S_{i,j}} \\
s.t. \quad & \sum_{S : e \in \delta(S), S \odot (i,j)} y_{S_{i,j}} \leq c_e, \quad \forall\ e \in E \\
& \sum_{S : S \odot (i,j)} y_{S_{i,j}} \leq \pi_{i,j}, \quad \forall\ (i, j) \in V \times V \\
& y_{S_{i,j}} \geq 0, \quad \forall\ S \subset V, S \odot (i, j)
\end{aligned}
$$

2 Dual Linear Program

Recall that in the last lecture, we showed the following program was equivalent to the the above dual program:

$$
\begin{aligned}
\max \quad & \sum_{S \subset V} y_S \\
s.t. \quad & \sum_{S: e \in \delta(S)} y_S \leq c_e, \quad \forall\, e \in E \\
& \sum_{S \in \mathcal{S}} y_S \leq f(\mathcal{S}), \quad \forall\, \mathcal{S} \subseteq 2^V \\
& y_S \geq 0, \quad \forall\, S \subset V
\end{aligned}
$$

where $f(\mathcal{S}) = \sum_{(i,j) \in V \times V, \mathcal{S} \odot (i,j)} \pi_{ij}$ is a function from 2^{2^V} to $\mathcal{R}^+$.

Lemma 2.1 *f is submodular, that is, for any $\mathcal{S}_1, \mathcal{S}_2$, $f(\mathcal{S}_1) + f(\mathcal{S}_2) \geq f(\mathcal{S}_1 \cap \mathcal{S}_2) + f(\mathcal{S}_1 \cup \mathcal{S}_2)$.*

Proof: The lemma follows from the facts that for any $(i,j) \in V \times V$, $\mathcal{S}_1 \odot (i,j)$ or $\mathcal{S}_2 \odot (i,j)$ is equivalent to $\mathcal{S}_1 \cup \mathcal{S}_2 \odot (i,j)$, and $\mathcal{S}_1 \cap \mathcal{S}_2 \odot (i,j)$ implies $\mathcal{S}_1 \odot (i,j)$ and $\mathcal{S}_2 \odot (i,j)$. □

We say an edge $e \in E$ is *tight*, if the first constraint of the above dual program holds with equality for e. We say a family $\mathcal{S} \subseteq 2^V$ is *tight*, if the second constraint of the above dual program holds with equality for $\mathcal{S}$.

Lemma 2.2 *Take any feasible solution y_S, suppose the constraints corresponding to $\mathcal{S}_1$ and $\mathcal{S}_2$ are tight, then the constraints corresponding to $\mathcal{S}_1 \cup \mathcal{S}_2$ and $\mathcal{S}_1 \cap \mathcal{S}_2$ are also tight.*

Proof: According to the above lemma, we have

$$f(\mathcal{S}_1) + f(\mathcal{S}_2) \geq f(\mathcal{S}_1 \cap \mathcal{S}_2) + f(\mathcal{S}_1 \cup \mathcal{S}_2).$$

And furthermore, since y_S is feasible,

$$\sum_{S \in \mathcal{S}_1 \cap \mathcal{S}_2} y_S + \sum_{S \in \mathcal{S}_1 \cup \mathcal{S}_2} y_S = \sum_{S \in \mathcal{S}_1} y_S + \sum_{S \in \mathcal{S}_2} y_S.$$

The lemma follows from the above two inequalities. □

3 Algorithm and Analysis

The details of the algorithm are referred to Jain and Hajiaghayi's paper, The Prize-Collecting Generalized Steiner Tree Problem via a New Approach of Primal-Dual Schema.

Initially, all connected components (vertices) are *active*. Then we raise the dual variables of all active components at a uniform rate until one of edges e or families $\mathcal{S}$ become tight. Then we set each element in the tight family set to be *inactive*, and repeat the above process until there is no active connected component. Note that in each iteration, there are exponentially possible family sets. However, we can find a tight set for the next iteration in polynomial time (details omit here).

Let the output of the algorithm be a forest F' and a set of pairs $\Gamma \subseteq V \times V$ not connected via F'.

Lemma 3.1 $\sum_{e \in F'} c_e \leq (2 - \frac{2}{n}) \sum_{S \subset V} y_S$.

Lemma 3.2 *The sum of penalties of marked pairs in* Γ *is at most* $\sum_{S \subset V} y_S$.

Therefore, we get the following conclusion:

Theorem 3.1 *The algorithm outputs a forest* F' *and a set of pairs* Γ *which are not connected via* F' *such that*

$$\sum_{e \in F'} c_e + \sum_{(i,j) \in \Gamma} \pi_{i,j} \leq (3 - \frac{2}{n}) \sum_{S \subset V} y_S \leq (3 - \frac{2}{n}) OPT.$$