**6.891 Approximation Algorithms** April 21, 2000

# Problem Set 4 Solutions

1. Consider the following problem:

**Minimum multicut problem in trees**

- **Input:**
  - A tree $T = (V, E)$
  - $k$ pairs of vertices $(s_i, t_i), i = 1 \ldots k$
  - Costs $c_e \geq 0$ for each edge $e \in E$
- **Goal:** Find a minimum-cost set of edges $F$ such that for all $i$, $s_i$ and $t_i$ are in different connected components of $G' = (V, E - F)$.

Obviously we can formulate this as a hitting set problem, where for each pair of vertices $(s_i, t_i)$ we have a set $T_i = \{e : e \text{ on unique path from } s_i \text{ to } t_i\}$ that must be hit. Suppose we root the tree at an arbitrary vertex $r$. Let $depth(v)$ be the number of edges on the path from $v$ to $r$. Let $lca(s_i, t_i)$ be the vertex $v$ on the path from $s_i$ to $t_i$ whose depth is minimum. Suppose we use the primal-dual method to solve this problem, where the violated set $T_i$ we choose in each iteration is the one that maximizes $depth(lca(s_i, t_i))$.

Prove that this gives a 2-approximation algorithm for the minimum multicut problem in trees.

**Solution:** We have the following primal and dual formulations of the problem:

$$\text{Min} \quad \sum_{e \in E} c_e x_e$$

subject to:

$$\sum_{e \in T_i} x_e \geq 1 \qquad \forall i$$

$$x_e \geq 0$$

$$\text{Max} \quad \sum_i y_i$$

subject to:

$$\sum_{i: e \in T_i} y_i \leq c_e \qquad \forall e \in E$$

$$y_i \geq 0 \qquad \forall i$$

Consider the following straightforward use of the primal-dual method:

$y \leftarrow 0$

$A \leftarrow \emptyset$

$l \leftarrow 0$

While $A$ is not feasible

$l \leftarrow l+1$

Choose $i$ such that $T_i$ violated, $depth(lca(s_i, t_i))$ maximized

Increase $y_i$ until $\sum_{j:e\in T_j} y_j = c_e$ for some $e_l \in T_i$

$A \leftarrow A \cup \{e_l\}$

For $j \leftarrow l$ downto 1

If $A - \{e_j\}$ feasible

$A \leftarrow A - \{e_j\}$,

Return $A' \leftarrow A$

By standard analysis, we have that

$$\sum_{e\in A'} c_e = \sum_{e\in A'} \sum_{i:e\in T_i} y_i = \sum_i |A' \cap T_i| y_i.$$

If we can show that $|A' \cap T_i| \leq 2$ for all $y_i > 0$, we will have a 2-approximation algorithm.

Pick an iteration $l$ in which the set $T_i$ is chosen. Let $A_l$ be the elements in $A$ at this iteration, and let $B = A' - A_l$. By arguments in class (Lecture 8), it must be the case that for any $e \in B$, $A_l \cup B - e$ is not a feasible solution. Note that $A_l \cap T_i = \emptyset$ (since $T_i$ is violated in this iteration), so that $A' \cap T_i = B \cap T_i$. So we need to show that $|B \cap T_i| \leq 2$. Recall that $T_i$ is a path from $s_i$ to $t_i$ in the tree; let $a_i = lca(s_i, t_i)$. Let $T_1$ be the path from $s_i$ to $a_i$, and $T_2$ the path from $a_i$ to $t_i$. We will show that $|B \cap T_1| \leq 1$, and the proof that $|B \cap T_2| \leq 1$ is identical; these two statements imply the theorem. Suppose that $|B \cap T_1| \geq 2$. We claim that removing all edges in $B \cap T_1$ from $B$ except the edge closest to $a_i$ is still a feasible solution, contradicting the minimality of $B$. To see this, notice that by the choice of $T_i$, for any other violated sets $T_j$ such that $T_i \cap T_j \neq \emptyset$, the set $T_1 \cap T_j$ is a path from some vertex in $T_1$ to $a_i$; if not, $T_j$ would have a least common ancestor of depth greater than the depth of $a_i$, which contradicts the choice of $T_i$. Therefore, if $T_j$ contains any edge in $B \cap T_1$, it contains the edge in $B \cap T_1$ closest to $a_i$.

2. The minimum-cost branching problem is as follows:

- **Input**:
  - An directed graph $G = (V, A)$
  - A root vertex $r \in V$
  - Weights $w_{ij} \geq 0$ for all $(i, j) \in A$
- **Goal**: Find a minimum-cost set of arcs $F \subseteq A$ such that for every $v \in V$, there is a directed path in $F$ from $r$ to $v$.

(a) Formulate the minimum-cost branching problem as a hitting set problem.

(b) Use the primal-dual method and the analysis given in class to give an optimal algorithm for this problem.

(a) For a directed graph $G = (V, A)$, define $\delta^-(S) = \{(u, v) \in A : u \notin S, v \in S\}$. Then we claim that the minimum-cost branching problem can be formulated as the following hitting set problem:

- Ground set: $A$
- Cost on ground elements: $w_{ij}$ for $(i, j) \in A$
- Sets to hit: $T_i = \delta^-(S)$ for $S$ s.t. $r \notin S$.

To see this, for a set $F$ of arcs, let $W$ be the set of all vertices reachable from $r$ in $(V, F)$. If $W \neq V$, then note that $r \notin V - W$. Also note that $\delta^-(V - W) = \emptyset$, since if $(i, j) \in \delta^-(V - W)$, it would be the case that $i \in W$ is reachable from $r$, $j \notin W$ is not reachable from $r$, yet $(i, j) \in F$, which is a contradiction. If $W = V$, then note that for any $S \neq \emptyset$ such that $r \notin S$, it must be the case that $\delta^-(S) \neq \emptyset$ since otherwise the vertices in $S$ would not be reachable from $r$.

(b) The primal and dual for this problem are:

$$\text{Min} \quad \sum_{e \in E} c_e x_e$$

subject to:

$$\sum_{e \in \delta^-(S)} x_e \geq 1 \qquad \forall S : r \notin S$$
$$x_e \geq 0$$

$$\text{Max} \quad \sum_{S} y_S$$

subject to:

$$\sum_{S : e \in \delta^-(S)} y_S \leq c_e \qquad \forall e \in E$$
$$y_S \geq 0 \qquad \forall S : r \notin S$$

Given an infeasible solution $A$, we have the algorithm choose the violated set $T(A) = \delta^-(S)$ for some strongly connected component $S$ of $(V, A)$, such that $r \notin S$. This leads to the following algorithm:

$y \leftarrow 0$

$A \leftarrow \emptyset$

$l \leftarrow 0$

While $A$ is not feasible

$l \leftarrow l + 1$

Choose strongly connected component $S$ of $(V, A)$, $r \notin S$.

Increase $y_S$ until $\exists e_l \in \delta^-(S) : \sum_{T:e\in\delta^-(T)} y_T = c_{e_l}$.

$A \leftarrow A \cup \{e_l\}$

For $j \leftarrow l$ downto 1

If $A - \{e_j\}$ feasible

$A \leftarrow A - \{e_j\}$,

Return $A' \leftarrow A$

By standard analysis, we have that

$$\sum_{e\in A'} c_e = \sum_{e\in A'} \sum_{S:e\in\delta^-(S)} y_S = \sum_S |A' \cap \delta^-(S)| y_S.$$

If we can show that $|A' \cap \delta^-(S)| = 1$ for all $y_S > 0$, we will have an optimal algorithm.

Pick an iteration $l$ in which the set $S$ is chosen. Let $A_l$ be the elements in $A$ at this iteration, and let $B = A' - A_l$. By arguments in class (Lecture 8), it must be the case that for any $e \in B$, $A_l \cup B - e$ is not a feasible solution. Note that $A_l \cap \delta^-(S) = \emptyset$ (since $S$ is violated in this iteration), so that $A' \cap \delta^-(S) = B \cap \delta^-(S)$. So we need to show that $|B \cap \delta^-(S)| = 1$. We give a proof by contradiction: suppose that $|B \cap \delta^-(S)| > 1$; in particular, suppose that both $(i, j) \in \delta^-(S) \cap B$ and $(i', j') \in \delta^-(S) \cap B$, where $(i, j) \neq (i', j')$. We know vertex $i$ can be reached from $r$ in $(V, A_l \cup B)$; without loss of generality, we assume that $i$ can be reached from $r$ in $(V, A_l \cup B)$ without using edge $(i', j')$. Since $S$ is a strongly connected component in $(V, A_l)$, and $j \in S$ and $j' \in S$, it must be the case that $j$ can reach $j'$ using edges only in $A$. Therefore, in $(V, A_l \cup B)$, $j$ can reach any vertex reachable from $j'$. Thus if we delete $(i', j')$ from $B$, the solution is still feasible, since (as we will argue), we can reach the same set of vertices from $r$ as before, and this will be a contradiction. By deleting $(i', j')$ the only problem might arise from vertices which could previously be reached only from $j'$ (including $j'$ itself). However, from $r$ we can reach $i$ (without using edge $(i', j')$) and therefore $j$, and therefore $j'$ (since the path from $j$ to $j'$ consists of edges in $A_l$), and therefore any vertex reachable from $j'$ in $(V, A_l \cup B)$.

3. Consider the following problem:

**Partition into $k$-paths**

- **Input:**
  - An unweighted graph $G = (V, E)$
  - Edge costs $c_e \geq 0$ for all $e \in E$ that obey the triangle inequality (i.e. $c_{(u,v)} \leq c_{(u,w)} + c_{(w,v)}$ for all $u, v, w \in V$).
  - Parameter $k$ such that $|V|$ is a multiple of $k$.
- **Goal:** Find a minimum-cost collection of paths of $k$ edges each such that each vertex is on exactly one path.

Consider also the related problem:

**Partition into $0(\textbf{mod}\ k)$-trees**

- **Input:**
  - An unweighted graph $G = (V, E)$
  - Edge costs $c_e \geq 0$ for all $e \in E$
  - Parameter $k$ such that $|V|$ is a multiple of $k$.
- **Goal:** Find a minimum-cost collection of trees such that each tree has $0(\text{mod}\ k)$ vertices, and each vertex is in exactly one tree.

(a) Given an $\alpha$-approximation algortithm for the second problem, produce a $2\alpha\left(1 - \frac{1}{k}\right)$-approximation algorithm for the first.

(b) Use the primal-dual method to give a 2-approximation algorithm for the second problem.

(c) Give a $4\left(1 - \frac{1}{k}\right)$-approximation algorithm for the problem of partitioning a graph into *cycles* of length exactly $k$.

**Solution:**

(a) Given a solution $A$ for the second problem, we double the edges for every tree in the solution, and shortcut each tree to a cycle to obtain a cycle on every connected component of $(V, A)$. Let $C$ be the set of edges in the solution obtained. It is easy to see that $\sum_{e \in C} c_e \leq 2 \sum_{e \in A} c_e$. Now each cycle contains $0 (\mathrm{mod}\ k)$ vertices. Given some starting edge in each cycle, consider removing every $k$th edge; this partitions the cycle into paths of $k-1$ edges each. Some choice of starting edge on each cycle must account for at least $\frac{1}{k}$ of the weight of that cycle; so we remove the edges for that choice of starting edge, to obtain a set of edges $C'$. Thus $C'$ is a solution to the first problem of cost no more than $\sum_{e \in C'} c_e \leq 2(1 - \frac{1}{k}) \sum_{e \in C} c_e \leq 2(1 - \frac{1}{k}) \sum_{e \in A} c_e$. Note that any solution to the first problem is also a valid solution to the second problem, so $OPT_1 \geq OPT_2$, where $OPT_1$ represents the value of an optimal solution to the first problem and $OPT_2$ the value of an optimal solution to the second problem. We know that $\sum_{e \in A} c_e \leq \alpha OPT_2 \leq \alpha OPT_1$, so that $\sum_{e \in C'} c_e \leq 2\alpha(1 - \frac{1}{k}) OPT_1$.

(b) We model the problem as a hitting set problem in which the ground elements are edges $e \in E$, the cost of the elements are $c_e$, and the sets to be hit are the sets $T_i = \delta(S_i)$ for all $\emptyset \neq S_i \subseteq V$ such that $|S_i| \not\equiv 0 (\mathrm{mod}\ k)$. Any solution $F$ to the problem is a solution to the hitting set problem, since for any $S_i$ such that $\delta(S_i)$ must be hit, there must exist some tree $T$ of $(V, F)$ such that $S_i$ contains some but not all vertices of $T$. Thus $F \cap \delta(S_i) \neq \emptyset$. Similarly, any solution to the hitting set problem contains a feasible solution to the tree-partitioning problem. Let $F$ be a subset of edges from a solution $A$ to the hitting set problem in which we take a tree from each connected component of $A$. Note that for every connected component $C$ of $A$, it must be the case that $|C| \equiv 0 (\mathrm{mod}\ k)$, since otherwise it is not a valid solution to the hitting set problem. Thus $F$ is a solution to the tree-partitioning problem.

This leads to the following primal and dual:

$$\text{Min} \quad \sum_{e \in E} c_e x_e$$

subject to:

$$\sum_{e \in \delta(S)} x_e \geq 1 \qquad \forall S : \emptyset \neq S \subseteq V, |S| \not\equiv 0 (\mathrm{mod}\ k)$$
$$x_e \geq 0$$

$$\text{Max} \quad \sum_{S} y_S$$

subject to:

$$\sum_{S : e \in \delta(S)} y_S \leq c_e \qquad \forall e \in E$$
$$y_S \geq 0 \qquad \forall S : \emptyset \neq S \subseteq V, |S| \not\equiv 0 (\mathrm{mod}\ k)$$

We now use an algorithm much like the one for the generalized Steiner tree problem:

$y \leftarrow 0$

$A \leftarrow \emptyset$

$l \leftarrow 0$

While $A$ is not feasible

$l \leftarrow l+1$

$\mathcal{C}_l \leftarrow \{S : S$ a connected component of $(V, A) : |S| \not\equiv 0 (\text{mod } k)\}$

Increase $y_S$ for all $S \in \mathcal{C}_l$ uniformly until $\exists e_l : \sum_{T:e\in\delta(T)} y_T = c_{e_l}$.

$A \leftarrow A \cup \{e_l\}$

For $j \leftarrow l$ downto 1

If $A - \{e_j\}$ feasible

$A \leftarrow A - \{e_j\}$,

Return $A' \leftarrow A$

Pick some iteration $l$, and let $A_l$ be the elements of $A$ in this iteration. Let $B = A' - A_l$. By the same arguments as above, we know that for any $e \in B$, $A_l \cup B - e$ is not feasible. By the same argument as was used in class for the generalized Steiner tree problem, we need only show that in every iteration $l$,

$$\sum_{C\in\mathcal{C}_l} |B \cap \delta(C)| \leq 2|\mathcal{C}_l|,$$

and this will be enough to show that

$$\sum_{e\in A'} c_e \leq 2\sum_S y_S \leq 2OPT.$$

The proof will be very similar to the argument used for the generalized Steiner tree.

Suppose we contract every connected component of $(V, A_l)$. In this contracted graph, call the nodes corresponding to the connected components in $\mathcal{C}_l$ red and the rest blue. Now consider the graph $G' = (V', B)$ where $V'$ is the vertex set. We note that $G'$ must be a forest, since if it had a cycle we could remove an edge of from $B$ and maintain feasibility, which is a contradiction.

How does the inequality we wish to prove translate to the graph $G'$? Note that $|B \cap \delta(C)|$ in $G$ for a connected component $C$ is equal to $deg(v)$ in $G'$ for the vertex $v$ corresponding to $C$. Similarly, $|\mathcal{C}_l|$ in $G$ is simply the number of red vertices in $G'$. We let $Red$ and $Blue$ represent the sets of red and blue vertices in $G'$, so that we can rewrite the above inequality as

$$\sum_{v \in Red} deg(v) \leq 2|Red|.$$

We will need the following claim.

**Claim 1** *If $v \in Blue$ then $deg(v) \neq 1$.*

**Proof:** If $deg(v) = 1$ then we claim $A_l \cup B - e$ is feasible for $e \in B$ and adjacent to $v$, which is a contradiction. Let $C$ be the connected component in $(V, A_l)$ that corresponds to the vertex $v$ in $G'$. Let $C'$ be the connected component of $(V, A_l \cup B)$ that contains $C$. Note that since $v$ is blue, it must be the case that $|C| \equiv 0 (\text{mod } k)$, and since $A_l \cup B$ is a feasible solution, it must be the case that $|C'| \equiv 0 (\text{mod } k)$. Thus by removing $e$ we split $C'$ into two components, $C' - C$ and $C$, each of which has $0 (\text{mod } k)$ vertices. Thus removing $e$ gives a feasible solution, a contradiction. □

To complete the proof, we first discard all blue nodes with $deg(v) = 0$. Then

$$\begin{aligned}
\sum_{v \in Red} deg(v) &= \sum_{v \in Red \cup Blue} deg(v) - \sum_{v \in Blue} deg(v) \\
&\leq 2(|Red| + |Blue|) - 2|Blue| \\
&= 2|Red|
\end{aligned}$$

The inequality follows since the sum of the degrees of nodes in a forest is at most twice the number of nodes, and since every blue node has degree at least two.

(c) Our algorithm will be to run the algorithm of part (a) to produce paths that have $k$ vertices each, then add the edge that joins the two endpoints of each path. This gives us a solution of cost at most twice the cost of the paths (by triangle inequality). Note that the algorithm of part (b) produces a solution $F$ for the tree-partitioning problem so that $\sum_{e \in F} c_e \leq 2 \sum_S y_S$, where the dual solution is to dual given in part (b). Thus by the reasoning of part (a), if we have a solution $C$ to the cycle-partitioning problem, the cost is $\sum_{e \in C} c_e \leq 8(1 - \frac{1}{k}) \sum_S y_S$.

Now observe that the optimal solution to the cycle partitioning problem is feasible for the same integer program in which we change the right-hand side of every inequality from 1 to 2 (this assumes $k > 2$):

$$\text{Min} \quad \sum_{e \in E} c_e x_e$$

subject to:

$$\sum_{e \in \delta(S)} x_e \geq 2 \qquad \forall S : S \neq \emptyset, |S| \not\equiv 0 (\text{mod } k)$$

$$x_e \in \{0, 1\}.$$

Thus the following dual is a lower bound on the cost of an optimal solution to the cycle-partitioning problem:

$$\text{Max} \quad 2 \sum_S y_S$$

subject to:

$$\sum_{S: e \in \delta(S)} y_S \leq c_e \qquad \forall e \in E$$

$$y_S \geq 0 \qquad \forall S : S \neq \emptyset, |S| \not\equiv 0 (\text{mod } k)$$

The dual produced by the primal-dual algorithm for the tree-partitioning problem is feasible for this dual LP, but has twice the value. Thus for the dual produced by the algorithm $2 \sum_S y_S \leq OPT_C$, where $OPT_C$ is the value of an optimal solution to the cycle-partitioning problem. Thus if our algorithm above produces a solution $C$ to the cycle-partitioning problem, we know that

$$\sum_{e \in C} c_e \leq 8(1 - \frac{1}{k}) \sum_S y_S \leq 4(1 - \frac{1}{k}) OPT_C,$$

and we have a $4(1 - \frac{1}{k})$-approximation algorithm for the cycle partitioning problem.

4. Consider the following problem:

**Prize-collecting Steiner tree problem**

- **Input:**
  - An undirected graph $G = (V, E)$
  - Edge costs $c_e \geq 0$ for all $e \in E$
  - Penalties $\pi_i \geq 0$ for all $i \in V$
  - A root vertex $r \in V$
- **Goal:** Find a tree $T$ containing the root $r$ that minimizes the sum of the costs of the edges in the tree and the penalties of the vertices not in the tree (that is, that minimizes $\sum_{e \in T} c_e + \sum_{i \notin V(T)} \pi_i$, where $V(T)$ is the set of vertices in tree $T$.)

(a) Formulate this problem as a hitting set problem.

(b) (Bonus problem) Using the primal-dual method, give a 2-approximation algorithm for this problem.

**Solution:**

(a) The ground set of items is the set of all edges $E$ together with the set of all subsets of $V$ not containing $r$; that is, $E \cup \{S : S \subseteq V - \{r\}\}$. The cost of a ground element $e \in E$ is $c_e$, while the cost of a ground element $S \subset V$ is $\sum_{i \in S} \pi_i$. The sets that must be hit are the sets $T_i = \delta(S_i) \cup \{S : S \supseteq S_i\}$ for all $\emptyset \neq S_i \subseteq V - \{r\}$.

To see that this models the prize-collecting Steiner tree problem, suppose we have some solution $T$ to the prize-collecting Steiner tree problem. Then for our hitting set problem we choose a solution $A$ consisting of all $e \in T$ plus the ground element $V - V(T)$. The cost of $A$ is exactly $\sum_{e \in T} c_e + \sum_{i \in V - V(T)} \pi_i$. To see that it solves the hitting set problem, choose any non-empty $S_i \subset V - \{r\}$. If $S_i$ is not disjoint from $V(T)$, then $\delta(S_i)$ contains an edge of $T$, and $T_i \cap A$ is non-empty. If $S_i$ is disjoint from $V(T)$, then $S_i \subseteq V - V(T)$, and thus $T_i \cap A$ is non-empty. Similarly, suppose that we have a solution $A$ to the hitting set problem; we show it gives a solution to the prize-collecting Steiner tree problem of cost at least as much. Let $T$ be a tree of edges of $A$ in the connected component containing the root (that is, a tree that spans all vertices in this connected component and is a subset of its edges). Since there is no edge of $A$ in $\delta(V - V(T))$, it must be the case that the solution $A$ contains some ground element $S$ such that $S \supseteq V - V(T)$. Thus the cost of $A$ must be at least the cost of the prize-collected Steiner tree $T$.

(b) See the attached pages for the solution.