Fall 2013, CS 583: Approximation Algorithms

Homework 4

Due: 11/4/2013

Instructions and Policy: Each student should write up their own solutions independently. You need to indicate the names of the people you discussed a problem with; ideally you should discuss with no more than two other people.

Solve as many problems as you can. I expect at least four.

Please write clearly and concisely - clarity and brevity will be rewarded. Refer to known facts as necessary. Your job is to convince me that you know the solution, as quickly as possible.

Problem 1 Consider the Steiner network problem and let $r_{\max}$ be the maximum connectivity requirement over all the pairs. Each edge has a cost c_e. Suppose an edge can be bought multiple times where each copy costs c_e. Show how one can use the Cut-LP based 2-approximation for the Steiner forest problem as a black box to obtain an $4(\lfloor \log_2 r_{\max} \rfloor + 1)$ approximation for this problem. *Hint:* Suppose all pairs had the same connectivity requirement h. Show how to obtain a 2-approximation in this case.

Problem 2 Problem 7.2 from Shmoys-Williamson book.

Problem 3 Recall the augmentation framework for covering a proper function. This problem is designed to fill in the details that were not covered formally during the lecture. Let $f : 2^V \to \mathbb{Z}_+$ be a proper function. Recall that f is proper if it is symmetric, $f(V) = 0$, and it is maximal, i.e., $(f(A \cup B) \leq \max\{f(A), f(B)\}$ for disjoint A, B).

- Prove that if A, B, C is a partition of V then the maximum of f over these three sets cannot be attained by only one of them.
- Let p be an integer and define $g_p : 2^V \to \mathbb{Z}_+$ as $g_p(S) = \min\{p, f(S)\}$. Show that g_p is proper.
- Supose $G = (V, E)$ is a graph and $F \subseteq E$ such that for all $S \subset V$ $|\delta_F(S)| \geq \min\{p - 1, f(S)\}$. Define $h : 2^V \to \{0, 1\}$ as $h(S) = 1$ if $g_p(S) = p$ and $|\delta_F(S)| = p - 1$, otherwise $h(S) = 0$. Prove that h is uncrossable. That is, if $h(A) = h(B) = 1$ then $h(A \cap B), h(A \cup B) = 1$ or $h(A - B), h(B - A) = 1$.
- Prove that any proper function f is skew-supermodular.

Problem 4 Consider the Steiner Network problem. Convince yourself that the requirement function is proper. Now consider the p'th phase of the augmentation algorithm and the function h defined as in the previous problem. Describe how to efficiently compute the minimal sets S such that $h(S) = 1$.

Problem 5 Let $G = (V, E)$ be an undirected graph and let f be an *uncrossable* function on the vertex set, i.e., a $\{0, 1\}$ valued function that satisfies

- $f(V) = 0$; and
- if $f(A) = f(B) = 1$ for any $A, B \subseteq V$, then either $f(A \cup B) = f(A \cap B) = 1$ or $f(A - B) = f(B - A) = 1$.

Recall that the primal-dual algorithms require the ability to answer the following questions.

1. Given $F \subseteq E$, is F a feasible solution for f?
2. Given $F \subseteq E$ what are the minimal violated sets with respect to F?

Also recall that if f is uncrossable, the minimial violated sets are disjoint.

1. Suppose f is an uncrossable function. We do not know a polynomial time algorithm to test if F is a feasible solution by simply using the value oracle for f. However, suppose you have an oracle that given $F \subseteq E$ returns whether F is feasible or not. Show how you can use such an oracle to compute in polynomial time the minimal violated sets with respect to a collection of edges A. First prove that if $S \subset V$ is a *maximal* set such that $A \cup \{(i, j) : i, j \in S\}$ is not feasible then $V \setminus S$ is a minimal violated set for A. Then deduce that the set of minimial violated sets can be obtained by less than $|V|^2$ calls to the feasibility oracle.
2. (Extra credit:) Suppose f is a proper function, i.e., $f : 2^V \to \mathbb{N}$ is an (non-negative) integer valued function satisfying,
   - $f(V) = 0$,
   - f satisfies maximality (see Problem 3),
   - f is symmetric.

   Suppose f is accessible by a value oracle which when given a set $S \subset V$ returns the value $f(S)$; Show that there is a polynomial time algorithm to determine if F is a feasible solution.
   Hint: Consider the cuts in the Gomory-Hu tree T for the graph $G[F]$.

Problem 6 Let $G = (V, E)$ be a *directed graph* with non-negative edge costs $c : E \to \mathbb{R}_+$. Consider the problem of finding the min-cost strongly connected sub-graph problem. That is, we want to find $E' \subseteq E$ of smallest cost such that $G(V, E')$ is strongly connected.

- The following problem can be solved in polynomial time. Given an edge-weighted directed graph $G = (V, E)$ find the min-cost *arborescence* rooted at a given node $r \in V$. Using this describe a 2-approximation for the min-cost strongly connected subgraph problem by computing an in-arborescence and an out-arborescence.

- Now we will consider the unweighted case of the problem, that is, each edge $e \in E$ has weight 1. Suppose that the longest simple cycle in G has at most k edges. Show that the optimum must contain at least $\frac{k}{k-1}(n-1)$ edges. Now consider the following greedy algorithm. Find a simple cycle C of length at least 3 if it exists; otherwise C is any cycle of length 2. Contract the vertices of the cycle C into a vertex and recurse on the remaining graph. Formalize this algorithm and show that this algorithm gives a 1.75 approximation.

Problem 7 Let $G = (V, E)$ be an undirected graph with non-negative edge-weights. We will be interested in finding the min-cost k-edge-connected subgraph problem — the goal is to find a min-cost set $E' \subseteq E$ such that the graph (V, E') is k-edge-connected. Now consider the rooted counterpart where we are given a specified root node $r \in V$ and the goal is to find a min-cost $E' \subseteq E$ such that for each $v \in V$ the edge-connectivity from r to v is at least k in (V, E').

- Prove that the k-edge-connected subgraph problem is the same as its rooted counterpart in undirected graphs. (The problems are NP-Hard for $k \geq 2$).

- In directed graphs the rooted version is solvable in polynomial time — that is the min-cost set $A' \subseteq A$ (here $H = (V, A)$ is a directed graph) such that in (V, A') there are k edge-disjoint paths from r to v for each $v \in V$. We will use this directed graph rooted result to obtain a 2-approximation for the unweighted k-connected-subgraph problem. Given undirected graph $G = (V, E)$, obtain a directed graph $H = (V, A)$ by replacing each undirected edge $uv \in E$ by directed edges (u, v) and (v, u) with the same cost as that of uv. Pick an arbitrary root r and solve the rooted k-connectivity version of the problem in H. Let $A' \subseteq A$ be the directed edges chosen by the algorithm. Obtain $E' \subseteq E$ by choosing uv to be included in E' if (u, v) *or* (v, u) is in A'. Argue why E' is feasible. Argue that there is an optimum solution $A^* \subseteq A$ of cost at most twice the cost of the optimum solution for the original problem in the undirected graph G. Put things together to prove that the algorithms gives a 2-approximation.