Spring 2016, CS 583: Approximation Algorithms

Homework 4
Due: Friday 4/8/2016

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.

Read through all the problems and think about them and how they relate to what we covered in the lectures. Solve as many problems as you can. Please submit your solutions to at least 4 problems.

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 Problem 7.2 from Shmoys-Williamson book.

Problem 2 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 3 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.

Problem 4 Read about the prima-dual based augmentation framework for covering a proper function in the survey of Goemans and Williamson. This problem is designed to make you read the relevant parts. 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 $h(A - B), h(B - A) = 1$.

Assuming that the problem of covering a uncrossable function admits a 2-approximation, derive a $2k$-approximation for the problem of covering a proper function where $k = \max_S f(S)$ is the maximum requirement.

Problem 5 Consider the Steiner Network problem. Convince yourself that the requirement function is proper.

1. 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 $h(S) = 1$.

2. We discussed the iterative rounding approach for Steiner network. In each iteration the algorithm solves an LP for the residual problem given that it had already chosen a set of edges F. Describe a polynomial-time separation oracle for the LP that needs to be solved.

Problem 6 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.

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

1. Given a set of edges $F \subseteq E$ the minimal violated sets of f with respect to F are those sets S such that $f(S) = 1$ and $\delta_F(S) = \emptyset$. Prove that if f is uncrossable then for any F, minimal violated sets of f with respect to F are disjoint.

2. Suppose f is an uncrossable function. In general there cannot be a polynomial time algorithm to test if F is a feasible solution by simply using the value oracle for f. For instance the following function h is uncrossable: there is a single set A such that $h(A) = 1$ and for all other sets B, $h(B) = 0$. Using just a value oracle it is infeasible to find A without trying all possible sets. 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.

3. (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 7 We discussed the Point-to-Point connection problem and mentioned that it can be cast as a special case of covering a proper function. See survey of Goemans and Williamson on network design. Here we consider the *unbalanced* point-to-point connection problem. The input consists of an edge-weighted undirected graph $G = (V, E)$ and two disjoint sets of vertices S and T where $|S| \leq |T|$. The goal is to find a min-cost subgraph H of G such that in each connected component C of H there are equal number of nodes from S and T; that is $|V(C) \cap S| = |V(C) \cap T|$. When $|S| = |T|$ we have the Point-to-Point connection problem. Show that if $|S| < |T|$ the resulting problem is not necessarily a special case of covering a proper or the more general problem of covering an uncrossable function.