CSOR W4231 (sec. 001, 002, H02): Analysis of Algorithms Nov 21

Homework 6: Due on Dec 5th by 12:01am

Instructors: *Alex Andoni, Cliff Stein*

Instructions. Please follow the homework policy and submission instructions in the Course Information handout. A few highlights to remember:

- follow the collaboration and academic honesty policies;
- write your name, uni, and collaborators on the top page;
- submission is via GradeScope (you are encouraged to use LaTeX, using the provided template);
- if you don't know how to solve a particular part of a problem, just write "*empty*", for which you will get 20% of the points of that part (in contrast, note that non-sensical text may get 0%).

Note that, for each bullet item of a problem, you can use the previous bullets as "black box", even if you didn't manage to solve them. Similarly, you can use anything from the lectures as black-box.

Problem 1

Let G be a graph and f be a maximum flow, which is *acyclic* (no unit of flow ever comes back to a node already visited). Give an algorithm that outputs a series of at most $|E|$ augmenting paths that, when augmented along would give rise to the flow f.

Note that you are not asked to give a new maximum flow algorithm, you asked how, given the maximum flow f, you can recreate a series of augmenting paths.

Problem 2

Suppose we want to solve the $s-t$ shortest path in a weighted directed graph G with the extra condition that the number of edges must be at most k, where k is the smallest number of edges in any path from s to t (ie, length of the shortest path in the unweighted version of the graph). You are to solve this problem by formulating it as an LP. You can assume that all weights are positive.

a) For a given graph $G = (V, E, w)$, nodes s, t, and a bound k, formulate the problem of finding the $s-t$ shortest path with at most k edges as an LP program with $e = |E|$ variables. *Hint:* first think of how you would formulate the program if you could set the unknowns x_i to be from $\{0, 1\}$, and then relax to $0 \leq x_i \leq 1$.

b) Show that the (optimal) value of the obtained LP is upper bounded by the shortest $s-t$ path with at most k edges.

c) Show that if the value of your LP is v, then there exists a shortest path of k edges and of weight v. *Hint:* use flow decomposition theorem. Can any path (with non-zero flow) have length more than k ?

Problem 3

Show that for any decision problem in NP, there is an algorithm that can solve it that runs in time $2^{O(n^k)}$, for some contant $k > 0$.

Problem 4 (optional for sections 002 and H02)

Given an integer $m \times n$ matrix A and an integer m-vector b, the *0-1 integer-programming problem* asks whether there exists an integer n-vector x with elements in the set $\{0, 1\}$ such that $Ax \leq b$. Prove that 0-1 integer programming is NP-complete. (Hint: Reduce from 3-SAT).