CSCI 6964   Fall 2007   Advanced Algorithm Design

Problem Set 6 – LP and Randomized Rounding

Due November 16

(1) Consider the following generalization of the maximum flow problem.

You are given a directed network $G = (V, E)$ with edge capacities $\{c_e\}$. Instead of a single (s, t) pair, you are given multiple pairs $(s_1, t_1), (s_2, t_2), \ldots, (s_k, t_k)$, where the s_i are sources of G and the t_i are sinks of G. You are also given k *demands* $d_1, \ldots, d_k$. The goal is to find k flows $f^{(1)}, \ldots, f^{(k)}$ with the following properties:

- $f^{(i)}$ is a valid flow from s_i to t_i.
- For each edge e, the total flow $f_e^{(1)} + f_e^{(2)} + \ldots + f_e^{(k)}$ does not exceed the capacity c_e.
- The size of each flow $f^{(i)}$ is at least the demand d_i.
- The size of the *total* flow (the sum of the flows) is as large as possible.

Show how to solve this problem in polynomial time.

(2) We are given a directed graph $G = (V, E)$, with special vertices s and t and capacities $\{c_e\}$ on the edges.

Let P be the set of paths from s to t. Consider the following linear program, with a variable x_e for every $e \in E$:

$$\begin{aligned} &\min \sum_{e \in E} c_e x_e \\ \forall p \in P, \quad &\sum_{e \in P} x_e \geq 1 \\ &0 \leq x_e \leq 1 \end{aligned}$$

(a) What does this LP find? Give an English explanation of the meaning of its objective, variables, and constraints. (Hint: "Fractional min-cut" will not be accepted as an answer. What happens if you think of x_e values as distances?)

(b) It can be shown that this LP always has an optimal integral solution (you do not have to prove this), and therefore would give us the minimum *s*-*t* cut. Notice, however, that this LP has an exponential number of constraints. Can we use the ellipsoid algorithm to solve this LP in time polynomial in $|V|, |E|$, and $\max\{c_e\}$?

(c) Write down the dual of this LP. Give an English explanation of the meaning of its objective, variables, and constraints.

(3) The following problem arises in both VLSI design (wiring together terminals on a chip) and in network routing (reserving bandwidth for a set of network connections). You are given a graph $G = (V, E)$ with integer edge capacities c_e (number of wires per channel on a chip, or number of virtual circuits through a link in a network) and a collection of pairs of vertices $\{(s_i, t_i)\}$. Each pair must be connected by a single path, such that the number of paths through an edge is less than its capacity. Finding a feasible solution is NP-Complete, so instead we seek an almost-feasible solution, in which the usage of each edge is close to its capacity (this is easily turned into a solution that works if every edge has a little more capacity than is needed for a feasible solution).

(a) Devise an integer linear program (IP) capturing this routing problem. Write it in terms of f_{ij}^k, an indicator variable for whether the path from between the k^{th} demand pair uses edge (i, j). *Hint:* think of a unit-value flow between each demand pair.

(b) Consider the LP relaxation of this IP. Argue that it can be seen as defining a collection of fractional paths between each demand pair, of total capacity 1.

(c) Given these fractional paths, devise a randomized rounding scheme that gives an integer solution. Argue that if each edge has capacity 1, and the path-finding problem has a feasible solution, then the rounding scheme can find a solution in polynomial time such that every edge carries $O(\log n)$ paths.

(d) *Extra Credit.* Generalize, and argue that if a solution exists in which every edge carries only w paths, then in polynomial time a solution can be found in which each edge of capacity w carries only $w + O(\sqrt{w \log n})$ paths (so for $w > \log n$, we get a "constant factor approximation").