CS 683 Spring 2001 – Problem Set 1

Due in class Thursday, February 22

(1) Consider the multicommodity flow problem on an undirected graph $G = (V, E)$, with pairs of terminals $(s_1, t_1), \ldots, (s_k, t_k)$. There is one unit of capacity on each edge, and we want to find the maximum value λ^* such that at least λ^* units of demand can be sent between each pair of terminals.

Recall that we looked at a linear programming formulation for this optimization problem in which, for every s_i-t_i path P, there was a variable $x_P \geq 0$ indicating the amount of s_i-t_i flow we choose to place on path P. We will call this linear program $MCF(G, \{(s_i, t_i)\})$. In the dual of this linear program, there is a length $y_e \geq 0$ for each edge e so that, if $d(s_i, t_i)$ denotes the shortest path distance from s_i to t_i with respect to the edge lengths $\{y_e\}$, then we have $\sum_{i=1}^{k} d(s_i, t_i) \geq 1$. We will call this dual linear program $SP(G, \{(s_i, t_i)\})$.

Consider optimal solutions $\{x_P^*\}$ and $\{y_e^*\}$ to this pair of linear programs. Suppose that in an optimal solution to $MCF(G, \{(s_i, t_i)\})$, we place a positive amount of s_i-t_i flow on a path P; that is, we set $x_P^* > 0$. Is it necessarily the case that P is a shortest s_i-t_i path with respect to the edge lengths $\{y_e^*\}$ defined by the optimal solution to $SP(G, \{(s_i, t_i)\})$? Give a proof, or a counter-example with explanation.

(2) The dual linear program $SP(G, \{(s_i, t_i)\})$ discussed in Question (1) has a polynomial number of variables, but can have an exponential number of inequalities.

Give a linear program $SP2(G, \{(s_i, t_i)\})$ that has a number of variables and inequalities that is polynomial in the input size, and which has the following property: From an optimal solution to $SP2(G, \{(s_i, t_i)\})$, one can compute in polynomial time a set of edge lengths $\{y_e^*\}$ corresponding to an optimal solution to $SP(G, \{(s_i, t_i)\})$.

(3) Many NP-complete problems come with an auxiliary parameter: consider the number of colors in a graph coloring problem; the length of a clause in a satisfiability problem; or the size of an independent set one is seeking in the decision version of the independent set problem. Sometimes (as in the first two of these examples), NP-completeness sets in once the value of this parameter reaches a particular constant (e.g. 3); other times (as in the third example), the problem is polynomial-time solvable for any fixed constant value of this auxiliary parameter.

Let's determine which is the case for the "narrowest cut" problem from class. Recall that we are given an undirected graph $G = (V, E)$ on n nodes, with k pairs of terminals $(s_1, t_1), \ldots, (s_k, t_k)$. We define

$$\eta(G, \{(s_i, t_i)\}) = \min_{U \subseteq V} \frac{e(U, V - U)}{\text{number of } (s_i, t_i) \text{ pairs split by } (U, V - U)},$$

We have claimed in class that computing $\eta(G, \{(s_i, t_i)\})$ is NP-hard in general. Determine its tractability with respect to the parameter k (i.e. the number of terminals), by doing one of the following two things:

- Describing an algorithm to compute $\eta(G, \{(s_i, t_i)\})$ with a running time that, for any fixed value of k, is polynomial in n; or

- Proving that for some fixed value of k, the decision problem $\eta(G, \{(s_i, t_i)\}) \leq \gamma$ is NP-complete.

(4) Consider the following version of the expansion problem, specialized to bipartite graphs. We are given a bipartite graph $G = (V, E)$, with V partitioned into X and Y in such a way that each edge has one end in X and the other in Y. We are also given an integer k and a rational number p. We ask: does there exist a set $S \subseteq X$ such that $|S| \leq k$ and strictly fewer than $p|S|$ nodes in Y have a neighbor in S?

We will call this the *Bipartite Expansion* problem. Note that it is really asking whether there is a small subset of X that does *not* "expand" by a sufficient amount.

Prove that *Bipartite Expansion* is NP-complete.

(5) Suppose we are given two systems of linear inequalities: $Ax \leq b$, where A is an $m \times n$ matrix and $x \in \mathbf{R}^n$, and $A'x' \leq b'$, where A' is an $\ell \times k$ matrix (with $k < n$) and $x' \in \mathbf{R}^k$.

Consider the following two-player game. The first player sets the first k coordinates in the vector x in such a way that if x' denotes the projection of x onto its first k coordinates, the system of inequalities $A'x' \leq b'$ is satisfied. The second player then tries to set the remaining $n - k$ coordinates in x in such a way that the system of inequalities $Ax \leq b$ is satisfied. The second player wins if she is able to succeed in doing this; the first player wins if she is not able to succeed.

The *Linear Programming Completion* problem asks the following: Given A, b, A', and b', can the first player choose x' so as to force a win?

Determine the complexity of the *Linear Programming Completion* problem by doing one of the following two things:

- Giving a polynomial-time algorithm for *Linear Programming Completion*; or

- Proving that one could use a polynomial-time algorithm for *Linear Programming Completion* to solve a known NP-complete problem in polynomial time.

(6) Make up an interesting open question based on the material we've covered so far. Specifically, describe a question that seems interesting to you; that you don't know how to answer; and for which — as far as you can tell — the answer is not known. Additionally, think about it for a little while and write down any ideas you have.

As with the other questions, you may work in groups on this but should write down the names of the people you collaborate with.