Homework 5, CS520, Spring '09, due Tuesday, April 28

1. Give another proof that integer linear programming is NP-hard by describing a polynomial-time reduction of SUBSET-SUM to it. Let $\{i_1, i_2, \ldots, i_n\}$ be the set of integers and let t be the target sum. Describe the integer linear program; you don't need to prove that it's correct.

2. Simplex requires that you start out with a feasible solution. This raised the question of how we find an initial feasible solution.

   We showed that the problem of finding a feasible solution can itself be reduced to linear programming. The trick is illustrated by inequalities 29.105-29.108 and 29.112-29.114 on pages 811 and 813.

   Show that the converse is true, that linear programming reduces to the problem of finding a feasible solution. That is, show that if you have a program P that inputs the constraints of a linear program and gives a feasible solution if one exists, you can use it to find an optimum solution to an arbitrary linear program.

   Instead of explaining the reduction, illustrate it on the following example:

   Maximize $5x_1 + 4x_2 + 3x_3$

   Subject to:
   $2x_1 + 3x_2 + x_3 \leq 5$
   $4x_1 + x_2 + 2x_3 \leq 11$
   $3x_1 + 4x_2 + 2x_3 \leq 8$
   $x_1, x_2, x_3 \geq 0.$

   You need to show what inequalities you pass to P in order to induce it to give you an optimum solution to this linear program.

3. Problem 26-3, page 693. Let us call a solution *feasible* if, for each experiment brought along, the required instruments are also brought along. It's feasible even if some instruments are brought along that aren't used in any of the experiments that were brought along.

   (a) Think about part a. Describe a bijection it implies from the set of feasible solutions to the set of finite $s-t$ cuts in the graph the book proposes. Interpret all the edges the book proposes to be directed edges.

   (b) Give a formula that describes the profit of a solution in terms of capacity of the corresponding cut. *Hint: Let* $P = \sum_{i=1}^{m} p_i$ *be Spock's "dream profit." (He majored in logic, and this is his first introduction to the disappointing world of economics.) Consider including* P *and his degree of disappointment as elements of your formula.*

4. In a communication network, two paths from s to t are *vertex-disjoint* if the only vertices they share are s and t, and *edge-disjoint* if they share no edges. Assume that the network is an undirected graph; data can travel either direction on an edge.

(a) A bad person armed with a cable cutter wants to disconnect s from t. A cable cutter can take out an edge. Every time she cuts a cable, she has a chance $p > 0$ of getting caught; p is the same for all cables. Describe an algorithm for identifying a set of cables that maximizes the bad person's probability of getting away with it.

Your algorithm should be based on a reduction to max-flow. It suffices to describe the reduction.

(b) Now suppose you're a good person at s. Your goal is to maximize the bad person's chance of getting caught. Network administrators insist that you have to decide on a set of edge-disjoint paths to communicate with t on. Once established, you can't change your mind about which paths to use. What's worse, the bad person can find out which paths you chose. Now the bad person doesn't have to sever all paths between s and t, just the chosen ones.

The administrators' rule seems to work against you. Give an algorithm that chooses the paths in a way that prevents the bad person's knowledge of the chosen paths from helping her. That is, prove that the bad person has to cut just as many cables as if she didn't know which paths you chose, and as many cables as if you were free to change your mind about the communication paths after the sabotage got underway.

Your algorithm should be based on a reduction to max-flow. It suffices to describe the reduction.

(c) Now consider the variant where the bad person is armed not with cable cutters, but with bombs. Each bomb can blow up a node. Because they are heavily guarded, she can't blow up s or t. Each time she plants one at some other node, she has a probability $p' > 0$ of getting caught. This probability is the same for all nodes other than s or t. Describe an algorithm for identifying a set of nodes that maximizes the bad person's chance of getting away with it.

It suffices to describe a reduction to max-flow.

(d) Suppose you're a good person at s. This time, you have to decide in advance on a set of vertex-disjoint paths between s and t, and the bad person can find out about them. Give an algorithm for selecting them that keeps this knowledge from helping the bad person get away with it.

It suffices to describe a reduction to max-flow.

(e) A secondary goal of the good person is to avoid ridicule. Unless you've taken steps in part b to prevent this, you might be in for some, despite having maximized the terrorist's chance of getting caught. Why is this, and what can you do to fix the problem?

(f) Apply a similar analysis to part d.