TTIC 31010 / CMSC 37000 Algorithms, Winter Quarter 2019

Homework # 5 **Presentations: March 11 – 13, 2019**

This is an oral-presentation assignment. You should work in groups of three. At some point before midnight on Sunday March 10, your group should sign up for a 1-hour time slot on the signup sheet by following the instructions on the course home page.

The Professor/TA/Grader reserves the right to select which group member presents which problem. You are allowed to bring in notes to help you (e.g., a writeup of your solutions).

You are not required to hand anything in at your presentation, but you may if you choose.

Problems:

1. **Randomized rounding.** In the Set-Cover problem we are given n points $\{1, 2, \ldots, n\}$ and m subsets of these points $S_1, S_2, \ldots, S_m$. Our goal is to find the smallest number of subsets needed to cover all the points. In the linear-programming relaxation of the problem, we assign variables x_i to each set, and ask to minimize $\sum_i x_i$ subject to the constraints:

   - for all i we have $0 \le x_i \le 1$, (each set chosen at most once)
   - for all j we have $\sum_{\{i:j \in S_i\}} x_i \ge 1$. (each point is "covered")

   Note that if we added the requirement that $x_i \in \{0, 1\}$ rather than just requiring $x_i \in [0, 1]$, then this would be identical to the set-cover problem.

   Now, suppose we solve the linear-programming relaxation, yielding a solution with $\sum_i x_i = \text{OPT}_{LP}$. Recall that $\text{OPT}_{LP} \le \text{OPT}_{SC}$ where OPT_{SC} is the number of sets needed to solve the original (integral) problem.

   (a) Recall that in the *vertex cover* problem we rounded the fractional solution to an integral solution by choosing all vertices i such that $x_i \ge 1/2$, yielding a 2-approximation. Show that this type of approach (even replacing the "1/2" with any other number) cannot provide a $o(n)$-approximation for set-cover. Specifically, give an example with n points and n sets such that the optimal LP solution has $x_i = \frac{1}{n-1}$ for all i, and yet $\text{OPT}_{SC} = 2$. This means that if we "round" the fractional solution by taking all sets S_i for $x_i \ge v$ then either we don't get a legal solution (if we use $v \ge \frac{1}{n-1}$) or we get a solution that uses n sets (if we use $v \le \frac{1}{n-1}$), which is a factor of $n/2$ larger than OPT_{SC}.

   (b) On the other hand, we can get an $O(\log n)$ approximation by using a method called "randomized rounding". Specifically, let $p_i = x_i/\text{OPT}_{LP}$ so that $\sum_i p_i = 1$. Now, pick $(2 \ln n)\text{OPT}_{LP} \le (2 \ln n)\text{OPT}_{SC}$ sets independently at random from the probability distribution $(p_1, p_2, \ldots, p_m)$.

   Prove that this has probability at least $1 - 1/n$ of covering all the points.

   A helpful fact to recall is that $(1 - 1/k)^k \le 1/e$ for all $k \ge 1$.

2. **TSP approximation.** Given a weighted undirected graph G, a *traveling salesman tour* for G is the shortest tour that starts at some node, visits all the vertices of G, and then returns to the start. We will allow the tour to visit vertices multiple times (so, our goal is the shortest cycle, not the shortest simple cycle). This version of the TSP that allows vertices to be visited multiple times is sometimes called the *metric* TSP problem, because we can think of there being an implicit complete graph H defined over the nodes of G, where the length of edge (u, v) in H is the length of the shortest path between u and v in G. (By construction, edge lengths in H satisfy the triangle inequality, so H is a metric. We're assuming that all edge weights in G are positive.)

   (a) Briefly: show why we can get a factor of 2 approximation to the TSP by finding a minimum spanning tree T for H and then performing a depth-first traversal of T.

   (b) The minimum spanning tree T must have an even number of nodes of odd degree (only considering the edges in T). In fact, *any* (undirected) graph must have an even number of nodes of odd degree. Why?

   (c) Let M be a minimum-cost perfect matching (in H) between the nodes of odd degree in T. I.e., if there are $2k$ nodes of odd degree in T, then M will consist of k edges in H, no two of which share an endpoint. Prove that the total length of edges in M is at most one-half the length of the optimal TSP tour.

   (d) It turns out that every connected graph in which all nodes have even degree must have an Euler tour: a tour that traverses each edge exactly once (you don't need to prove this). Moreover there is an efficient algorithm to find such tours when they exist (you don't need to prove this either). Finally, there are efficient algorithms to solve for minimum-cost perfect matchings (you don't need to prove this either). Argue how you can combine all this together, along with your analysis above, to get a 1.5 approximation to the TSP.

3. **Skiing games.** Consider the rent-or-buy problem in the simple case that the cost to buy skis is twice the rental cost. The optimal deterministic algorithm (as discussed in class) is: "rent the first time, buy the second time" for a competitive ratio of $3/2$.

   For a randomized algorithm, the definition of competitive ratio is the worst case, over all possible scenarios, of the ratio of our *expected* cost under that scenario to the optimal cost for that scenario. (Scenario = how many times we go skiing.) Equivalently, think of a matrix game with a row for each deterministic strategy, a column for each possible scenario, and with entry M_{ij} equal to cost of strategy i in scenario j, divided by the optimal cost for scenario j. The competitive ratio of a randomized algorithm A is then the minimax value of A for this game, and the *optimal* randomized algorithm is the minimax *optimal* strategy for this game.

   (a) This game, annoyingly, has an infinite number of columns and an infinite number of rows. Specifically, column j (for $j = 1, 2, 3, \ldots$) is the scenario in which we ski j times, and row i (for $i = 1, 2, 3, \ldots$) is the strategy "rent the first $i - 1$ times you ski and buy skis the ith time" (so $i = 0$ means "buy right away"). In fact, let's add one more column, called "column ∞" for the scenario we ski forever.

Argue that without loss of generality, we can assume the adversary chooses only column $j = 1$ or $j = \infty$. Formally, argue that for any $j > 1$ we have $M_{ij} \leq M_{i\infty}$ for all i. This means that an algorithm achieving competitive ratio v in a world with only those two scenarios possible (ski once or ski infinitely often) will achieve competitive ratio v over the whole range of scenarios.

(b) Now that we have reduced the game to having just two columns, argue that the minimax-optimal strategy can put probability 0 on all rows except for $i = 1$ and $i = 2$. Formally, argue that for any $i > 2$ we have $M_{ij} \geq M_{2j}$ for all j.

(c) Finally, write down and solve the 2-by-2 game that results. What randomized strategy has the best competitive ratio and what is its ratio?