

Homework Assignment 6

November 25, 2003

Due Date: December 9

Core Problems

- (15 points) Consider the following on-line problem. You have to buy an item and know that you are going to be offered a sequence of prices $\langle p_1, p_2, p_3, \dots, p_m \rangle$ all in the range, $\ell \leq p_i \leq h$. When presented with the option of buying an item at price p_i , you already know p_1, \dots, p_{i-1} and have passed on these offers, but you know nothing about p_i, \dots, p_m except that they are all in the range of $\ell - -h$. So when presented with p_i you can either buy the item for that price, or pass in which case you have committed to buying that item at price among p_{i+1}, \dots, p_m . (So if you wait until p_m , then you are obligated to buy the item at that price.) Your goal is to buy the item for the cheapest price. Here is a deterministic on-line algorithm A for this problem. Accept the first p_i such that $p_i \leq \sqrt{\ell h}$. (If all $p_i > \sqrt{\ell h}$ then A accepts p_m .)

Prove that this algorithm is \sqrt{r} competitive where $r = h/\ell$ is the ratio of the highest to lowest possible price.

- (10 pts) Consider the “searching for a hole in the fence” problem that was considered in class. You are initially standing at the origin of the real line. The hole is at some unknown (positive or negative) integer coordinate h . You can move left or right at a cost equal to the distance moved, and the game continues until you reach h . The goal is to minimize the distance traveled. In class we gave a deterministic on-line algorithm with a competitive ratio of 9 (which is optimal for deterministic strategies). In this problem you are to describe and analyze a randomized algorithm whose competitive ratio is at most 7. If you prove something better than 7-competitive then you will receive more than 10 pts. So challenge yourself to prove the best competitive ratio that you can.

Advanced Problems (Required only for CS 539T students)

- (10 pts) Consider the following on-line problem. You have a specialized piece of equipment that people will pay to use. However, only one person at a time can use this equipment. In an on-line fashion you receive requests (i.e. people come and ask to use the equipment but you have no idea how many customers will come in the future). Each person wants to use the equipment for a 1 hour period and also gives a slack period $s \geq 0$ (for s an arbitrary real) which indicates how long the customer is willing to wait to use the equipment. So if the customer i comes at time t_i and announces that she has a slack of s_i , she is willing to begin using the equipment anytime between time t_i and time $t_i + s_i$ and will use the equipment for one hour after starting. You can wait up until time $t_i + s_i$ to make a final decision as to whether or not to let the customer use the equipment, but once you tell the customer that she can use the equipment, you cannot go back on that. Your goal is to decide which customers to allow to use the equipment and which to send away. Your goal is to maximize the number of requests serviced. Give a 2-competitive on-line algorithm. Be sure to clearly describe the algorithm (which should say which customers get to use the equipment and when they use it so that you can argue that you have never allowed two customers to use the equipment at the same time). You must also be sure to prove your algorithm has a competitive ratio of 2.