Prof. Dr. Susanne Albers
Dr. Suzanne van der Ster
Dario Frascaria
Lehrstuhl für Theoretische Informatik
Fakultät für Informatik
Technische Universität München

Problem set 4
May 6, 2016
Summer Semester 2016

Online and Approximation Algorithms

Due May 13, 2016 before 10:00

Exercise 1 (Dynamic List Update problem - 10 points)

We modify the list update problem by adding the operations INSERT(a) and DELETE(a). Let n be the current length of the list. Whenever INSERT(a) is called, the list is searched for element a. If a is already in the list, the costs for the operation are equal to its position. If not, it is inserted to position $n+1$ of the list. In this case, the cost is $n+1$. In addition, we are allowed to move the inserted item to any position in the list.
If DELETE(a) is called, the list is searched for element a. If a is found, the cost for the operation is equal to its position, otherwise the cost is n.
The algorithm Move-To-Front (MTF) moves the newly inserted element to the front of the list. Show that MTF is still 2-competitive.

Exercise 2 (List Update, Paid exchanges - 10 points)

(a) Consider the three-element list with the following initial configuration: (A, B, C). Prove that the optimal offline cost to serve the request sequence C, B, C, B is 8. Now consider an optimal offline algorithm without paid exchanges and prove that it pays 9 to serve the same sequence.

(b) Show that in the static model, any list accessing algorithm can be transformed into an algorithm that only uses paid transactions without increasing the cost.

Exercise 3 (Modified BIT - 10 points)

Recall the BIT algorithm for the list update problem which assigns a random bit to every item before any request is served. When an item is requested, then its bit is flipped. If the bit becomes 1, then the item is moved to the front of the list. Otherwise, its position does not change.
We modify the algorithm as follows. If the requested item is already in front of the list, then we do not flip its bit. Show that the modified algorithm is no longer $\frac{7}{4}$-competitive.

Exercise 4 (RMTF$_p$ - 10 points)

In the lecture we saw the randomized online list update algorithm Random-Move-To-Front (RMTF), that moves the requested element to the front of the list with probability

$\frac{1}{2}$. We consider a generalized version RMTF_p that moves a requested element to the front of the list with probability $p \in (0,1)$. Show that the competitive ratio of RMTF_p is lower bounded by $\frac{1}{p} - \epsilon$ for any constant $\epsilon > 0$.