HS 2014 Prof. R. Wattenhofer / K.-T. Foerster, T. Langner, J. Seidel

Discrete Event Systems

Exercise Sheet 12

1 PhD-Scheduling [Exam]

In this assignment we wish to study the PhD-Scheduling-Problem, in which Professor Arno Nym wants to assign some of his more cumbersome tasks to his m PhD students. When Prof. Nym enters his office in the morning there is a pile of tasks on his desk, and he is only able to see the topmost task at a time. He wants do decide *online* to which of his m PhD students he assigns the topmost task before taking a look at the next task. Prof. Nym does not know in advance when the last task is reached, because there might be more tasks hidden beneath the topmost one. Every task needs to be assigned to exactly one PhD student, and every PhD student is able to solve any task. Of course, Prof. Nym would like all tasks to be finished as fast as possible.

A task is defined by its *effort* $e_i \in \mathbb{R}^+$, independent of the PhD student who processes the task, and different tasks may require different effort. The *load* L_j of a PhD student j is defined by the sum of the efforts of all tasks assigned to j. He thus needs time L_j to finish all assigned tasks. Prof. Nym's goal is that all tasks are finished in as little time as possible, which means that the maximum load of any PhD student is to be minimized. For a sequence of tasks $\sigma = (e_1, e_2, \ldots)$, the cost of an assignment $\mathcal{A}$ of tasks to students is thus defined by

$$\text{cost}(\mathcal{A}(\sigma)) := \max_{j=1,\ldots,m} L_j(\mathcal{A}(\sigma)).$$

In the following we analyze the algorithm SmallLoad that assigns the topmost task to a student whose load is minimal with respect to the previous assignments.

a) Assume that Prof. Nym has only two PhD students, i.e., $m = 2$.

(i) Describe the execution of SmallLoad for the following sequence of tasks σ.

$$\sigma = (2,\ 5,\ 4,\ 3,\ 7)$$

What would be an optimal solution for this sequence? How large is the competitive ratio of SmallLoad with respect to σ?

(ii) The cost of a solution found by SmallLoad can be worse than that of an optimal solution. Construct a sequence of tasks σ' for which the costs of SmallLoad are as high as possible compared to the costs of an optimal solution.

(iii) Is the algorithm SmallLoad c-competitive for some constant c? If so, give the smallest possible such c. Prove your claim!

(iv) Is there an efficient way to compute an optimal *offline* solution? Explain your answer.

b) Now, analyze the case in which the number of PhD students m is arbitrary. What is the smallest competitive ratio of SmallLoad now? Prove your claim!

2 The Winter Train Problem

We consider two trains T1 and T2 transporting skiers from Sarnen and Lucerne to Engelberg. Because there is only one ground rail track from Stans to Engelberg, at most one train might be between these two villages at any time. There is a switch in Stans, which either connects the track between Sarnen and Engelberg xor the track between Lucerne and Engelberg. After the train conductor has pressed a button m in (Sarnen or Lucerne), its train moves to Engelberg, but might have to wait in Stans until the other train has left the critical section. Once arrived in Engelberg, the train waits for 100s and then returns.

The sensors a1, a2, b1, b2 and c indicate the presence of a train with the value 1, otherwise, the value is 0. The switch in Stans is accessed through a variable G, as indicated in the picture. Finally, the motion of the trains is regulated by assigning 'R', 'L' or 'S' to the train, to move right, left, or stop, respectively.

The situation is shown below. Draw the corresponding State Chart using the notation introduced in the Figure!