Thomas Kesselheim
Alexander Braun

May 6, 2021
Due: May 19, 2021 at noon

Algorithms and Uncertainty

Summer Term 2021

Exercise Set 4

Due to the public holiday (Christi Himmelfahrt) on Thursday, May 13, there will be no tutorial sessions next week. As a consequence, this sheet is only due in two weeks, but also covers content from next weeks' lectures (on Optimal Stopping and the Secretary Problem). We will discuss this sheet in the tutorials on May 20.

Exercise 1: (4 Points)
Let us consider a generalization of the version of Markov decision processes covered in the lecture. For every state $s \in \mathcal{S}$, only a subset of the actions $\mathcal{A}_s \subseteq \mathcal{A}$, $\mathcal{A}_s \neq \emptyset$, is available. Devise an algorithm that computes an optimal policy for a finite time horizon T, show its correctness, and give a bound on its running time.

Exercise 2: (5 Points)
We consider the stochastic decision problem from Tutorial 4, Exercise 1: There are n boxes; box i contains a prize of 1 Euro with probability q_i and is empty otherwise. The game ends when we have found a non-empty box. That is, the final prize is either 0 Euros or 1 Euro. At each point in time, we can also decide to stop playing. We can open as many boxes as we like but opening box i costs c_i Euros. Find an optimal policy.

Hint: It can be useful to consider the cases $n = 1$ and $n = 2$ first.

Exercise 3: (2 Points)
Consider the cost-minimization variant of the optimal stopping problem in which we know the prior distributions. In step i, we can stop the sequence at cost c_i. We have to stop the sequence at some point and want to minimize the cost for doing so.
Show that there is **no** $\alpha < \infty$ such that for all distributions the optimal policy has cost at most $\alpha \mathbf{E}[\min_i c_i]$.

Hint: It sufficies to consider $n = 2$.

Exercise 4: (3 Points)
Consider the following algorithm for the Secretary Problem from Lecture 10: Do not select any of the first $\lceil \frac{n}{2} \rceil$ candidates. Afterwards, pick the first candidate whose value exceeds all previous ones. Observe that this corresponds to a threshold algorithm with $\tau = \lceil \frac{n}{2} \rceil$.
Do not use Theorem 10.1 to show that this algorithm selects the best value with probability at least $\frac{1}{4}$.