Online Motion Planning, WT 13/14
Exercise sheet 4
University of Bonn, Inst. for Computer Science, Dpt. I

- *You can hand in your written solutions until Tuesday,* 19.11., 14:15*, in room E.06.*

**Exercise 10: Competitive analysis, minimum distance (4 points)**

We consider the problem of finding a door in a wall. Starting from point $s$ on a line $\ell$, a robot moves along $\ell$ until it has found a "door" – in other words, a destination point $t$ on $\ell$.

It is a common assumption that the target $t$ cannot lie arbitrarily close to $s$. Recall that $ALG$ is $C$-competitive, if there exists a constant $\alpha \geq 0$, where

$$\text{ALG}(t) \leq C \cdot \text{OPT}(t) + \alpha$$

holds for all possible placements of $t$ on $\ell$.

Show that the following holds for any two constants $K > k > 0$ and any algorithm $ALG$ for locating $t$:

$ALG$ is a $C$-competitive algorithm for finding $t$, assuming that the distance from $t$ to $s$ is at least $k$, if and only if $ALG$ is $C$-competitive assuming that the distance from $t$ to $s$ is at least $K$.

*Please turn the page!*

### Exercise 11: Competitive complexity (4 points)

Find an upper bound on the competitive complexity of the following strategy $ALG$ for locating a door in a wall.

Let $a > 1$ be a constant, then the $i$-th move ($i = 1, 2, \dots$) of the robot is defined as follows. If $i$ is odd, the robot moves to the point at distance $a^{i-1}$ to the left of its starting point $s$, otherwise it moves to the point at distance $a^{i-1}$ to the right of $s$.

*Hint: Use the same analysis as used in the lecture for the case* $a = 2$.

### Exercise 12: Bug leaving from closest vertex (4 points)

We consider a modification to the $BUG$ algorithm. The bug starts at its starting point $s$. In order to reach a destination point $t$, the bug moves in direction of $t$, until an obstacle $O$ hinders its movements. As usual, the bug walks along the boundary of $O$ and keeps track of the distance to $t$.

The modification is as follows. Instead of leaving $O$ at a point closest to $t$, the bug leaves $O$ at a *vertex* $v$ of $O$'s boundary which is closest to $t$. Then, the bug continues in direction of $t$, until it encounters another obstacle.

Prove or disprove that the modified $BUG$ algorithm will eventually reach the target point $t$, although possibly not as quickly as the unmodified algorithm.