Albert-Ludwigs-Universität Freiburg
Lecture: Introduction to Mobile Robotics
Summer term 2019

Institut für Informatik

Prof. Dr. Wolfram Burgard
Dr. Daniel Büscher
Lukas Luft
Marina Kollmitz
Iman Nematollahi

mobilerobotics@informatik.uni-freiburg.de

Sheet 2

Topic: Locomotion, Differential drive kinematics

Due date: 10.05.2019

Exercise 1: Locomotion

A robot equipped with a differential drive starts at position $x = 1.0\,\text{m}$, $y = 2.0\,\text{m}$ and with heading $\theta = \frac{\pi}{2}$. It has to move to the position $x = 1.5\,\text{m}$, $y = 2.0\,\text{m}$, $\theta = \frac{\pi}{2}$ (all angles in radians). The movement of the vehicle is described by steering commands (v_l = speed of left wheel, v_r = speed of right wheel, t = driving time).

(a) What is the minimal number of steering commands (v_l, v_r, t) needed to guide the vehicle to the desired target location?

(b) What is the length of the shortest trajectory under this constraint?

(c) Which sequence of steering commands guides the robot on the shortest trajectory to the desired location if an arbitrary number of steering commands can be used? The maximum velocity of each wheel is v and the distance between both wheels is l.

(d) What is the length of this trajectory?

Note: the length of a trajectory refers to the traveled distance along the trajectory.

Exercise 2: Differential Drive Implementation

Write a function in *Python* that implements the forward kinematics for the differential drive as explained in the lecture.

(a) The function header should look like

```
def diffdrive(x, y, theta, v_l, v_r, t, l):
    return x_n, y_n, theta_n
```

where x, y, and θ is the pose of the robot, v_l and v_r are the speed of the left and right wheel, t is the driving time, and l is the distance between the wheels of the robot. The output of the function is the new pose of the robot x_n, y_n, and θ_n.

(b) After reaching position $x = 1.5\,\mathrm{m}$, $y = 2.0\,\mathrm{m}$, and $\theta = \frac{\pi}{2}$ the robot executes the following sequence of steering commands:

(a) $c_1 = (v_l = 0.3\,\mathrm{m/s}, v_r = 0.3\,\mathrm{m/s}, t = 3\mathrm{s})$

(b) $c_2 = (v_l = 0.1\,\mathrm{m/s}, v_r = -0.1\,\mathrm{m/s}, t = 1\,\mathrm{s})$

(c) $c_3 = (v_l = 0.2\,\mathrm{m/s}, v_r = 0\,\mathrm{m/s}, t = 2\,\mathrm{s})$

Use the function to compute the position of the robot after the execution of each command in the sequence (the distance l between the wheels of the robot is $0.5\,\mathrm{m}$).