Task Sheet 2

Braitenberg Vehicles

Deadline 09:00am November 10, 2023

Review on November 10, 2023

Lecture: *Introduction to Autonomous Robotics*, Winter Term 2022/23

Lecturer: Prof. Dr.-Ing. Heiko Hamann

Tutor: Jonas Kuckling & Simay Atasoy

Invented by neuroscientist Valentino Braitenberg, the so-called Braitenberg Vehicles exhibit quite complex behavior while only following simple rules. In this exercise sheet, we implement two Braitenberg Vehicles: *Attractive and Repulsive* and *Indecisive.* We choose between the two behaviors by pressing the forward button for *Attractive and Repulsive* and by pressing the backward button for *Indecisive.* By pressing the center button, we stop the robot.

Specification of Behaviors

Attractive and Repulsive: When an object approaches the robot from behind, the robot runs away until it is out of range.

Indecisive: When the robot does not detect an object with its front sensors, it moves forward. When it detects an object, it moves backward as long as it detects the object.

Task 2.1 Selection mechanism

a) Download and extract the ZIP-file *IAR2.zip* from ILIAS. Open the controller template *IAR2.py.*

b) Set the robot's *state* according to the Braitenberg Vehicle to be executed. Pressing the right button starts *Attractive and Repulsive* (state *"AR"*), pressing the left button starts *Indecisive* (state *"INDECISIVE"*), and pressing the center button stops the robot (state *"STOP"*). In the template, the state variable is already initialized to the state *"STOP"*.

```
state = "STOP"
```

Additionally, we have already specified that pressing the forward button starts *Attractive and Repulsive* by setting the state to *"AR"*.

```
if robot["button.forward"] == 1:
 state = "AR"
```

Extend the existing code to include the other two cases.

Task 2.2 Attractive and Repulsive

a) Implement the behavior of the **Attractive and Repulsive** Braitenberg Vehicle in the section

```
if state == "AR":
 continue
```

of the template. Replace the line `continue` with your solution. The values of the back proximity sensors of the Thymio II can be accessed via

```
robot["prox.horizontal"][5]
robot["prox.horizontal"][6]
```

b) Launch the Webots simulator:

macOS/Windows: Double click on the application icon

Ubuntu: Open a new terminal and run the command

```
webots
```

c) In Webots, click *File > Open World*. Navigate to the folder *IAR2/.worlds* and open *IAR2.wbt*.

d) Run the simulation (Play button).

e) Open a terminal and navigate to the folder *IAR2*. Test your implementation by running your code using the command

```
python3 IAR2.py -s
```

You can test the behavior by manually moving the robot close to the arena wall (backwards!).

Stop your program by pressing *CTRL + C*.

Task 2.3 Indecisive

1. Implement the behavior of the **Indecisive** Braitenberg Vehicle in the section

```
elif state == "INDECISIVE":
  continue
```

of the template. Replace the line `continue` with your solution.

2. Test your solution in the Webots simulator.