Robotics Challenge 2015

Fabienne Gürtler
IN.2022 Robotics, BSc Course, 2nd Sem.
University of Fribourg
fabienne.guertler@unifr.ch

May 28, 2015

Abstract

Brief description of the content (5-10 lines). Helps people decide whether the report is relevant for them or not. Usually written at the end.

Keywords. add, keywords, for, indexing

Introduction

Objectives of this project and brief description of the problem at hand, the achieved solution and results.

1 Problem statement

There are 3 e-pucks in an arena. In this arena there are 2 blue doors and 4 red switches. To open a door a switch must be pressed, meaning to pass through a door two e-pucks have to work together. One finds either the door or the switch and waits in front of it while the other searches the missing component. Once both the door and the switch have been found the door opens and the e-puck can pass. Another requirement for the challenge is that all e-pucks must have the same code, except for calibration values. Before I begin presenting one of the many possible solution strategies I'll present the e-puck and the arena in detail.

E-puck The Ecole Polytechnique Fédérale de Lausanne started the e-puck project with the main goal to develop a miniature mobile robot for educational purposes.

Figure 1: E-puck

The design of the e-puck is based on desktop size and flexibility. By default it comes with sound sensors, a 3D accelerometer, 8 proximity sensors and a camera. It is possible to extend the e-puck with more hardware like a Color LED Communication Turret, ground sensors, magnetic wheels and more. For this project we used e-pucks with ground sensors. The e-pucks communicate via Bluetooth. A e-puck is a event-based robot, it can both receive and emit events.

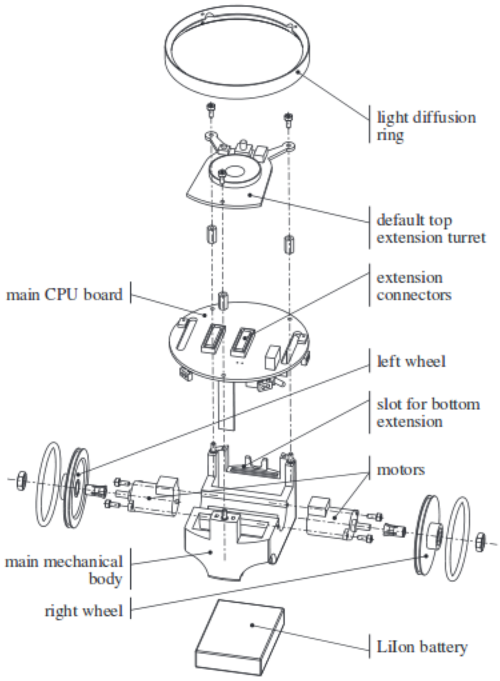

Figure 2: Mechanical structure

The structure is based on one single part to keep the desing simple and elegant. This basic part has a diameter of 7 cm and supports the motor, the circuit and the battery. The e-puck uses a miniature stepper motor with gear reduction and the wheels have a diameter of 4,1 cm.

Arena The arena is a 80cm x 160cm zone divided into two subspaces of roughly the same area. Both the floor and the walls are white. In the area there are the following randomly placed objects:

- 2 blue doors, equipped with black stripes on the floor
- 4 red switches, two per subspace
- three epucks

Software For the completion of the project we used the help of two programs, Aseba Studio and Aseba Playground.
Aseba Playground simulates the arena, enabling us to test the robots behaviour virtually before trying to program the real robots.
Aseba Studio ...TODO

2 Solution Strategy

The principle of our solution is to manage the e-puck's behaviour with the three variables *redFound, blueFound* and *color*.

redFound	blueFound	color	do	send
0	0		searchRB	-
0	1		searchR	-
0	1		wait	foundBlue
1	0		searchB	-
1	0		wait	foundRed
1	1		wait	-
1	1		wait	-
1	1		passDoor	passed

Figure 3: Coding for e-puck behaviour

3 Implementation

Description of how the solution strategy in Section 2 was implemented. Only short excerpts of code or pseudo code should be used here. Longer excerpts can be included in Appendix B.

4 Validation

Description of how the solution turned out and what problems were encountered. Since this report is accompanied by a short video, it can be referenced to illustrate the result.

Conclusion

Synthesis of the paper and outlook for further work.

Personal Comments

Feedback to the course and project (what you liked, what you didn't like, what you learned, ...).

References

[1] Justin Zobel. *Writing for Computer Science*, 2nd edition. Springer-Verlag, London, 2004, 275 pages.

[2] Valentino Braitenberg. *Vehicles: Experiments in Synthetic Psychology*. MIT Press, 1986.

[3] *Aseba User Manual*. https://aseba.wikidot.com/en:asebausermanual. Last visited: 29.04.15.

Appendix A Experimental Results

Place to list the gathered data.

Appendix B Source Code

Place to list source code.

B.1 Advanced Love Behavior

The code below shows an e-puck implementing the advanced love behavior.

```
<!--node e-puck 2-->
<node nodeId="2" name="e-puck 2">
#-----------------------------------------------------------
# Advanced love behaviour
#-----------------------------------------------------------
var proxRight
var proxLeft
var ds   # delta speed

onevent ir_sensors
    # proximity
    proxRight = (prox[0] + 3*prox[1] + prox[2])/5
    proxLeft = (prox[7] + 3*prox[6] + prox[5])/5

    # check which side is closer to obstacle
    if proxRight > proxLeft then # turn left
        # delta speed ds
        call math.muldiv(ds, S_INIT, proxRight, P_THRESH)
        # right and left speed
        speed.right = ds
        speed.left = S_INIT - ds
    else # turn right
        # delta speed ds
        call math.muldiv(ds, S_INIT, proxLeft, P_THRESH)
        # right and left speed
        speed.right = S_INIT - ds
        speed.left = ds
    end
</node>
```

B.2 Explore Behavior

The code below shows an e-puck implementing the explore behavior.

```
<!--node e-puck 3-->
<node nodeId="3" name="e-puck 3">
#-------------------------------------------------------------
# Explore behaviour
#-------------------------------------------------------------
var proxRight
var proxLeft
var ds   # delta speed

onevent ir_sensors
    # proximity
    proxRight = (4*prox[0] + 2*prox[1] + prox[2])/7
    proxLeft = (4*prox[7] + 2*prox[6] + prox[5])/7

    # check which side is closer to obstacle
    if proxRight > proxLeft then # turn left
        # delta speed ds
        call math.muldiv(ds, S_INIT, proxRight, P_THRESH)
        # right and left speed
        speed.right = S_INIT + ds
        speed.left = S_INIT - ds
    else # turn right
        # delta speed ds
        call math.muldiv(ds, S_INIT, proxLeft, P_THRESH)
        # right and left speed
        speed.right = S_INIT - ds
        speed.left = S_INIT + ds
    end
</node>
```

Appendix C LATEX Examples

This section shows some common uses of LATEX features.

C.1 Images

Example of how to include an image can be seen in Figure 4. All figures must be referenced somewhere in the report.

C.2 Tables

Example of how to include a table can be seen in Figure 5. All figures must be referenced somewhere in the report.

Figure 4: Including an image.

Title 1	**Title 2**
item 11	item 12
item 21	item 22

Figure 5: Table with caption.

C.3 Listings

Example of how to include listing can be seen in Figure 6 and Figure 7. All figures must be referenced somewhere in the report.

C.4 Font Style and Text Size

The font style may be modified: **bold**, *italic*, *Emphasis*, CAPITALS, `verbatim`, etc.
The text size can be changed: tiny, small, large, huge, etc.

C.5 Enumerations and Other Lists

Enumerations are easy, there is the `enumerate` environment:

1. First item
2. Second item

```
<!--list of constants-->
<constant value="1500" name="P_THRESH"/>
<constant value="600" name="S_INIT"/>
```

Figure 6: Listing included from file.

```
var v [3]
onevent ir_sensors
    ground.get_values(v)
```

Figure 7: Listing within LaTeX.

3. Third item

For lists, there is the `itemize` environment:

- First item
- Second item
- Third item

For definitions lists, there is the `description` environment:

First term – Description of the first term

Second term – Description of the second term

C.6 Quotations and References

Books and other documentation can be referenced as [2] and websites as [3].