Project description and guidelines

Robotics, BSc Course, 2nd Sem., Dr. Julien Nembrini, Manuel Mondal

Handout on Thursday Mars 21 2019 — Week 7 to 13 — Due on **Monday May 16 2019, 12h00**

Presentation on **Thursday May 23 2019**

The remaining part of the semester will be devoted to a group project which consists of a basic task and a challenge. The challenge can be freely defined by the students but must be presented to the teaching team for approval before starting work on it.

Important dates :

April 3, 23h00 Short description of challenge idea, sent per email to the teaching team

April 4 Challenge idea discussion

April 11 Final short challenge description, sent per email

May 16, 12h00 Hard deadline for submitting the final project (including presentation)

May 23, 13h15 Final presentation

The techniques exercised during the first part of the semester are sufficient in combination to solve the task proposed. All behaviors needed are variants of Explorer/Lover behaviors or PID controllers. However, it is allowed to use other techniques if wished. As during the first part of the semester, all robots should use the exact same code.

Grade The final project is evaluated on the basis of :

- the project code/algorithm (30%)
- the challenge code/algorithm (20%)
- the report (30%)
- the video (10%)
- the final presentation (10%)

Important elements include :

- dynamicity, robustness and efficiency of the algorithm
- clarity and readability of the code
- structure and clarity of the report
- orthography and grammar

Hand in :

Upload your code, report, video and presentation (pdf) on Moodle before May 16, 12h00 (noon). You may amend your presentation between the deadline and May 23, but you must submit it before 12h00 (noon) on May 23.

Zip files not following the naming convention will not be considered.

Basic task

The basic task consists of letting 3 robots (1) find the color block in their respective area of the arena, (2) detect which area they are located in based, on differences in the environment, and (3) go the their assigned area based on information communicated by others (robot in the left area → red, middle area → no movement, right area → blue).

Environment

In the basic task, the robot arena, through which the e-pucks navigate, has a height and width of 80x160 centimeters. The inside is divided in three areas by lines on the ground. Each area contains blocks, possibly with one white and the other colored (see fig 1). Apart from the fact that they are joined and that the white block hides the color from other areas, the configuration and the position of the blocks are not fixed : the algorithm should be robust to blocks positioned differently and with colors distributed differently over the areas. Note that the blocks will not be placed near the walls or the lines.

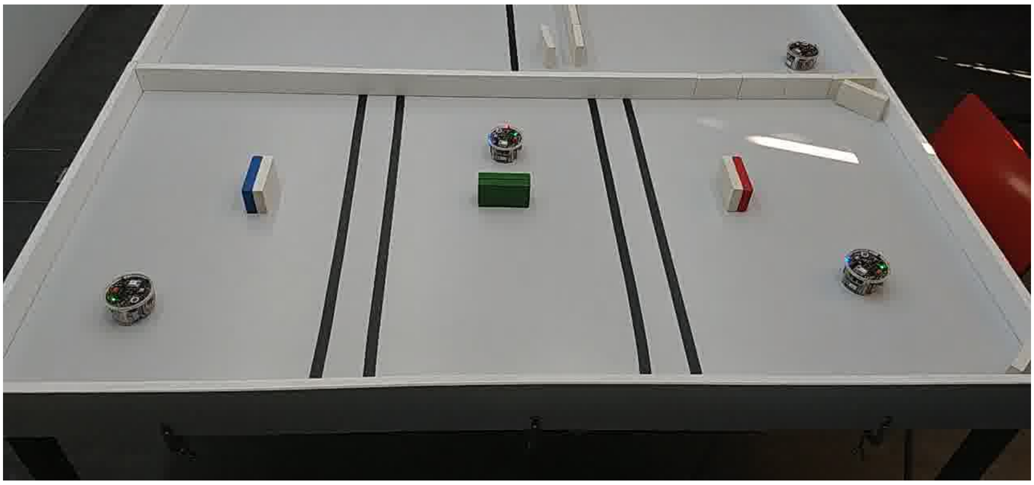

FIGURE 1 – Project arena (three areas, each one with a group of colored blocks)

In general, robots are not allowed to cross lines on the ground. Only at the end of the task are they allowed to cross them to reach their target areas. They are however allowed to follow lines.

We assume that 3 robots are in the arena and that the communication is reliable. Each robot is placed randomly in each area. It then follows the following phases (in order) :

- Phase 1 : Find the color of the block inside its area.
- Phase 2 : Detect in which area it is located.
- Phase 3 : Move to its assigned area based on color and area detected by all robots.

Phase 1, Finding color blocks

Task to achieve : the 3 robots need to find the color block in their area and wait for other to do the same.

During this phase, the robots are not allowed to cross lines on the ground.

1. Each robots efficiently roam/scan its area in search of a colored block. In order to speed up this behaviour, the robot has to use its camera to steer towards a recognized color.
2. Once the first color is found, he approaches it and stops in front of it in order to be sure that it is indeed in its area.
3. Once the color is confirmed, it signals other robtos that he completed Phase 1.
4. When all robots have completed Phase 1, Phase 2 starts.

Phase 2, Detecting area

Task to achieve : the 3 robots need to detect in which area they are located by detecting differences on the wall geometry.

1. Each robot efficiently explores its area to detect differences in the environment. It is allowed to follow lines and walls.
2. Once it has recognised the area, it notifies other robots that it has completed Phase 2 by sending the area it is in (left, right, or middle) together with the color of the block it detected to the others (red, green or blue).
3. Once each robot has detected its area, phase 3 starts.

Phase 3, Moving to the assigned area

Task to achieve : Using the information communicated by other robots, the robots have to move towards the area assigned to them after the following rule :

- left area → red area,
- middle area → no movement,
- right area → blue area.

The information communicated allows each robot to cross lines the exact number of times it has to (e.g. 4x from left to right), meaning that a robust line crossing behaviour has to be developed. Once it has reached the corresponding area, each robot must signal task completion with the appropriate LED pattern.

Challenge

The challenge consists of defining, designing and solving an additional challenging task.

The challenge can be freely defined by the groups, but must be described in a short email (5-6 sentences at most) that will be sent before **April 3 23h00**. This idea will be discussed with the teaching team on **April 4** to get approval before starting work on it. A final short description will be sent per email before **April 11, 23h00**. Possible challenges are :

- autonomous "cars"
- collectively solve a maze
- recognize objects/forms
- neural network to improve robot behaviour
- make multi-robot formations
- communicate with other media (sound, light, etc)
- **your proposal**

Each group will have to choose a different challenge. If two groups choose something similar, variants that are sufficiently different will have to be proposed. If you are hesitating between multiple ideas, feel free to include all of them in your initial mail.

Guidelines

You are free to choose which strategy you implement for solving the basic task and the challenge, but keep the following guidelines in mind :

- You are expected to discuss the problem, including coding and testing, in group of three persons. Please inform if you wish to change group before starting the project.

- **Report :** each student has to write his own report in Latex, maximum 12 pages + appendices (including this document as Appendix A). The report should present the project (basic task + challenge) as a whole. An empty template is provided on moodle
- **Code :** one code per group. We ask you to hand in a modular, readable and commented code. Your code will be thoroughly read.
- **Video :** one video per group, max 4 minutes. An uninterrupted and accelerated shot of the whole basic task process (phases 1 to 3) must be included. Any acceleration must be annotated in the video and further annotations for clarity are encouraged.
- **Presentation :** each student has to do his own presentation (max. 4 minutes) focusing on a particularly interesting aspect of the project (task or challenge), different for each student.

- The e-pucks need to collaborate and communicate. 3 e-pucks have to be used.
- Navigating the environment can be done using the lover/explorer behaviors and PID controller (and possibly others). Be careful to the robustness of other controllers you may choose to use.
- Don't use any hard coded behavior. The e-pucks should be able to complete the basic task even if their starting positions change or a different arena is used (all the robots basically know is that the arena is divided into 3 areas divided by lines, that they start each one in a part with one colored block).
- Make your solution as dynamic (independent of initial configuration), robust (catch error situations) and efficient (achieving the task in a short time) as possible.
- Document your code (events, methods, algorithms, etc.).
- Use the LED patterns described in the appendix in order to indicate to an observer what the robots are doing. If you add custom patterns, document it using the same tabular presentation as in the appendix.

Appendix : LED patterns

The following is a list of the LED patterns that must be used. For each pattern, the indexes of the LEDs are given. The LEDs have to be turned on long enough to be visible in the video.

Behaviours		LED indexes	Comment
Recognized colors :	Red	3	
	Green	2	if used in other context
	Blue	1	
General behaviours :	Explorer	0 + 1/2/3	the recognized color
	Lover	1/2/3	the recognized color
	Line following	0	
	Wall following	0,1,2/0,2,3	side dependent
Phase 1 completed	In front of red block	2,3	
	In front of green block	1,2	
	In front of blue block	2,3	
Phase 2 completed		0,1,3	
Task completed		ad-lib	choose your favorite !
Others			to be specified by the students