Universidade de Aveiro

DETI - Departamento de Electrónica Telecomunicações e Informática

Robótica Móvel e Inteligente

2017/18

Project 2

Pathfinder solver using the CiberRato simulation environment

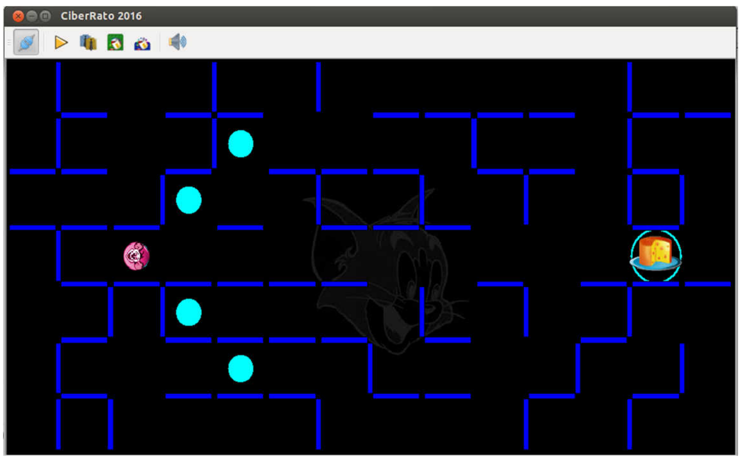
Objectives

- The main objective is the development of a robotic agent to command a mobile robot, making it move to an unknown target position in a semi-structured environment and then return back to the start position through the shortest path. At the end, the agent should also show the acquired map and the path used to return.
- In order to fulfil the main objective:
  - o Information from the obstacle sensors has to be used in order to avoid and follow the walls.
  - o The target position is detected using the ground sensor, as the value changes when the robot center is inside this region; the start position is not marked in any way.
  - o There are no encoder sensors but the robot pose may be estimated using the motor model and the velocity commands sent to the simulated robot.
  - o To deal with the noise in sensors, some kind of filtering should be used.
- The agent should save a file called "map" (with an optional extension) that represents the estimated labirynth map. This representation should clearly show all cells that have been visited by the robot and also all walls separating cells that have been detected. The path planned for the return should also be represented.
- The CiberRato simulation environment will be used to test and assess the developed agent.

Environment

The environment where the robot navigates is a delimited rectangular arena that can be seen as a bi-dimensional array of fixed size cells. Each cell is a square with side length equal to twice the diameter of the robot. The maximum size of the arena is 7-cells tall and 14-cells wide.

A maze is defined by putting thin walls between adjacent cells. The target cell is marked with a non IR reflective colour. The next figure shows a possible scenario.



The agent may be developed in C/C++, Java or Python. The CiberRato Environment includes libraries that provide adequate APIs for communication between the agent and the simulator in these programming languages.

Work plan

1. To facilitate the development of the agent, a set of appropriate basic behaviours (coping with obstacle avoidance, wall following, moving towards a point, rotation, etc.) should be implemented.

2. In order to accomplish the returning phase, a map representing the environment should be constructed. Then, a search algorithm (such as A*) should be applied to the map in order to find the shortest path to the start position.

3. The map of the environment discovered by the robot and to be saved and shown at the end, can be saved/displayed either in graphical or textual manner.

4. Write a report describing the approach used in the resolution of the previous points. The report should describe the overall architecture of the agent, the implemented behaviours and how they can be combined to solve the problem, and the results obtained by the agent. It should also contain references to documents (articles, books, reports, etc.) that supported the development of the work.

   This report should be written using the LNCS template available at the following link: http://www.springer.com/computer/lncs?SGWID=0-164-6-793341-0).

About the simulation environment

- When reaching the target area, the robot must turn on its 'visiting led'.
- Before starting the returning phase, the robot must turn on its 'returning led' while it is still inside the target cell.
- The ground sensor can be used to sense the target cell.
- **Only the following sensors may be used in the final demonstration**: ObstacleSensors, GroundSensor, Compass, Bumper and Time. The noise configurations should be the same as those provided in the source code.
- Other sensors (ex: GPS, Beacon, Score, etc.) may be used for development but will not be available in the final demonstration.
- The initial position of the robot will always be centered in one cell and horizontally aligned.
- The agent should be able to use the command line option `-host` *`host_name_or_ip`* to connect to a simulator that is executing in a different computer. This is the setup of the demonstration to be provided during the presentation of the work.

Movement Model

Consider that the robot pose is given by (x, y, θ), where x and y define the robot position and θ specifies the robot orientation. When the command sent to the simulator at step t is `DriveMotors(`in_{left_t}`,` in_{right_t}`)`, then the following equations determine the new robot pose.

An IIR filter is applied to each of the powers provided by the agent (in_{left_t} and in_{right_t}) that models the inertial characteristics of the motors and generates the effective powers that will be applied to the motors:

$$\mathrm{out_t} = (\mathrm{in_t}*0.5 + \mathrm{out_{t-1}}*0.5)*\mathrm{noise}$$

Then, the movement is separated in a translation of the robot position, considering its current orientation, and the change of the orientation of the robot.

For translation, we have:

$$\mathrm{lin} = \frac{\mathrm{out_{right}} + \mathrm{out_{left}}}{2}$$
$$\mathrm{x_t} = \mathrm{x_{t-1}} + \mathrm{lin}*\cos(\theta_{t-1})\mathrm{y_t} = \mathrm{y_{t-1}} + \mathrm{lin}*\sin(\theta_{t-1})$$

For rotation, we have:

$$\mathrm{rot} = \frac{\mathrm{out_{right}} - \mathrm{out_{left}}}{\mathrm{robotDiam}}$$
$$\theta_t = \theta_{t-1} + \mathrm{rot}$$

This provides the new robot pose (x_t ,y_t, θ_t) at the next step.

Presentation

Each group will make a presentation of its work which consists of three parts: an oral presentation, based on powerpoint or similar (10 minutes, maximum); a demonstration of the performance of the robot agent in some unknown scenarios and finally a short discussion of the work.

Deliverables

- Source code of the developed agent
- Report (in PDF format; according to Springer LNCS paper template)
- Presentation (in PDF format).

All deliverables must be available at the group repository at the code.ua platform after the relevant “Important date” (see below).

Evaluation

- Robotic agent (structure, contents and performance): 45%
- Report: 25%
- Presentation: 30%

Important dates

- Submission of the deliverables: November, 11th, 2017

- Presentation: November, 13th, 2017

Bibliography

- “Principles of Robot Motion: Theory, Algorithms, and Implementations”, Howie Choset et al., MIT Press, Boston, 2005
- “Introduction to Autonomous Mobile Robots”, Second Edition, Roland Siegwart et al., MIT Press, 2011
- “Artificial Intelligence: A Modern Approach”, 3rd edition, Russel and Norvig, Pearson, 2009