IT University of Copenhagen November 11, 2024

Advanced Robotics

Exam project: robot tag (draft)

Project Description

The robots have decided to play tag. Your task is to develop a seeker robot and an avoider robot (it could be the same robot with different behaviors, or two separate robots). In tag one robot is assigned to be the seeker and the others are avoiders. The seeker starts in the center and the avoiders in the corners. In the center of the arena a zone is marked with gray floor in which an avoider is safe. However, there is only space for one avoider at a time! The winner of a round is the avoider that stays alive the longest or all the avoiders still alive after 3 minutes.

Technical requirements

You receive a Thymio II robot and you are welcome to use LEGO too. We will make a rectangular arena where the floor is white, the safe zone is gray and the perimeter is a black line. There will be no walls.

1. You are free to use any approach, sensor and robot (taught/used in the course)
2. It is expected that you employ some form of robot learning or evolutionary robotics, but maybe this is not where you want to start.
3. A seeker robot has to turn its LEDs red and if in the safe zone orange.
4. An avoider robot has to turn its LEDs blue, but if it is safe green and if it is tagged purple (due to embarrassment).
5. A seeker transmits "1" and if an avoider receives this it is tagged and has to stand still.
6. An avoider transmits "2" and if an avoider in the safe zone receives this it has to leave immediately and can first after 5 seconds begin to transmit "2".
7. It is essential that you focus on how to make a robot that performs well also in practice.

Requirements determined in class

Every robotic system is optimally designed through a synergy of morphology, task, control, and environment. Thus, early in the design phase, we must determine the most suitable arena. Furthermore, is the suggested color scheme above the best choice?

Implementation specifics

To enable IR communication between the Thymio robots, please refer to the `IR_Communication.py` script. This script provides an illustration of IR communication with Thymio.
How the communication works:

- The script compiles and forwards an Aseba program to the robot.
- Upon establishing communication, the variable `prox.comm.rx` updates when information is received. To access this variable, use `node.v.prox.comm.rx`.

For effective integration:

1. Incorporate methods and functionalities from the `IR_Communication.py` script into your project.
2. Actively monitor incoming communications. While the provided example demonstrates one approach, you're encouraged to explore alternative methods. For instance, you might consider implementing an emit function to send data to a Raspberry Pi when communication is received.

Handin

Work on the project will start in full on the 19th of November. There will be no mandatory teaching for the rest of the course, but some outings and talks may be planned. We will have a tag-game on the 10th of December at 10:00 with prizes for best performing robots. Handin of a group report documenting your project the 18th of December at 14:00 (exam handin in on LearnIT - this is a strict deadline set by SAP so don't be late). The report should contain documentation of how you met the technical requirements supported by experimental validation. Remember proper approach and experimental methods!