Composite behaviors

– Intelligent Robotic Systems –

Andrea Roli
andrea.roli@unibo.it
Dept. of Computer Science and Engineering (DISI)
Alma Mater Studiorum Università di Bologna

In this lab activity you are asked to combine simple behaviors into a composite one. In case, you can reuse code developed in the previous lab activity.

Task

The robot is asked to find a light source and go towards it, while avoiding collisions with other objects, such as walls, boxes and other robots. The robot should reach the target as fast as possible and, once reached it, it should stay close to it (either standing or moving). For physical constraints the wheel velocity cannot exceed the value 15 (i.e., 15^{-2} m/s). The robot (a *footbot*) is equipped with both light and proximity sensors.

Some suggestions:

- Compare the behaviour of the robot in a condition without noise and another with noise (set noise level in both actuator and sensors; a reasonable noise level does not exceed 0.05).
- Test your robot in different environments; for example, an arena with boxes placed randomly (vary number and size of boxes), another arena with one or more walls between the robot and the light—the walls may also have thin doors. Try also to set the light at different intensities and heights—this latter parameters impact the visibility of the target.
- Try the controller also with more than one robot in the arena (just place n robots randomly by using `distribute`). Is the control software still achieving the goal?
- As you know the position of the light bulb, you can compute the distance between the robot and the target by means of the positioning sensor. Therefore, you can evaluate the performance of the robot by estimating the time needed to reach the light, e.g. in terms of the number of steps. To this aim, you may want to set a minimum radius around the target. Alternatively, you can either count how many times the robot

can reach the light in a given time (just set the simulation duration in the ARGoS configuration file) or take the distance between the robot and the light at the end of the simulation.

Food for thought

- Is it better to focus first on the two basic behaviors separately and then combine them, or to program a composite behavior from scratch?
- Does the controller need memory to let the robot achieve the desired task? If yes, why? If not, would it help?
- Think about possible alternative ways to combine two behaviours.