Project

For the OS course we had to build a Lego NXT/MindStorm robot which shall be the winner of a "shoot-and-defend" soccer-like game. The robot participates in a competition which rounds last 2 minutes. Before the game starts, the robots are placed in a black line delimited zone of the arena that cut into two equal parts with a black line and which is 1.2 meters large and 2 meters long, and protected on all sides with a fence high of 7 cm.

At the end of the competition, each robot should have played the role of the attacker and the defender without changing the robots’ shapes “sensors, architecture…” ( roles are exchanged after a round finishes). However, the robots can be freely re-programmed between competition rounds.

When playing the role of the attacker, the robot has to scores points and it can go wherever it wants in the arena. To do so, it should manage to send a ball to the opposite fence (two points for a direct shot/ one point for an indirect one/ one point bonus for the red ball). Furthermore, the attacker can contain up to two blue balls at startup, two others are placed in its side at 20cm of the delimitation black line, and 20cm from the fence. The arena contains two other red balls which are placed similarly to the blue ones in the defender's camp.

When playing the role of the defender, the robot shouldn’t cross the black delimitation line. It loses four points the first time it does.

The winning robot of a competition, is the one with the highest score after two rounds and the winner of the tournaments is the one with the highest number of victories and accumulated points.

Like in a Mario Kart game, robots are allowed to throw and drop things in the arena which shouldn’t be harmful for them. It may be still annoying.

Architecture

For our robot, we have used three different sensors:

* A color sensor which is used in the two roles played by the robot:

While playing the role of attacker, the sensor is used to detect the delimitation black lineand when combined with the results from the compass, we can tell if the robot is in its own camp or in the enemy’s. It’s also used for the scanning process to grab the ball. Indeed, when the detected object has either a blue or red color the robot can catch it.

While playing the role of the defender, the sensor is used to detect the delimitation black line and thus to prevent the robot from crossing it, what makes us lose points if it had to happen.

* A compass sensor which is used for the scanning process (rotation to the left and right looking for balls) and also to drive the robot toward the opponent goal so as to score goals.
* An ultrasonic sensor which used to detect object near the robot which can be either the wall, the opponent or the enemy. Once the objects are detected, the robot need to analyze their nature by calling the color sensor. Indeed, if the color of the detected object is red or blue, it means that it’s a ball if not it means the robot should avoid the obstacle by going the other way (the detected object may be the wall, the opponent or the black line (while defending))

To grab and throw a ball, we have designed an arm that looks like a jaw which is opened when we get near to catch it and closed afterwards.

The robot is also equipped with three motors, two of them are used to move the robot, and the third one is used to open and close the jaws so as to grab a ball.

The main behavior of the robot is to always be in movement scanning the arena, as long as it didn’t grab a ball. When a ball is grabbed, the robot goes to the opponent camp and throw it to score a goal. This task is repeated until the game finishes.

Media

Team

The first meeting we had was about discussing the project and trying to draw the objectives so as to determine its specifications before starting the conception and the development cycles.

Once we have defined the objectives we started working on the architecture. How should the robot look like if it has to find balls, grab them, move…? The answers to these questions helped us determine the hardware specifications: what sensors, motors are needed and what’s the design of the element that should be used to grab balls and throw them but also the software ones: what functions should be implemented.

“An” was responsible of assembling the robot according to the architecture we came up with after defining the hardware specifications.

In the meantime, “Kawtar” and “Nhu” started by designing the algorithms for the attacking role according to the software specifications.

“Kawtar” was responsible for the development of the “grabbing/throwing” functions whereas “Nhu” was responsible of the development of the “scanning process“function. Kawtar started by coding some basic functions which have been used as frameworks for the main algorithms (ex: reading colors, moving, avoiding obstacles, avoiding the black line, opening and closing the jaws). “Nhu” was helping “An” assembling the robot and checking the coherence of the chosen architecture. Once the robot was assembled and the frameworks implemented and tested. The real coding started.

“An” started working on the “defending” algorithm whereas “Nhu” and “Kawtar” were each coding the mentioned attacking algorithms above. Since we were using a spiral model for the development, each time we built a prototype for the functions we were responsible for we had to discuss together and improve the implementation for the next prototype.

After finishing working on the “throwing and grabbing” functions “Kawtar” started on implementing a way to know where the robot is in the arena using the compass, the color sensor to detect the delimitation black line and a timer so that we could score a goal once the robot gets to the opponent camp then go back to his own and look for balls again and if it doesn’t find a ball we use the timer to go back to the robot’s camp and restart the process.

“Nhu” started working on the integration of the whole functions while “An” was working on avoiding the black line when the robot is playing the defender role.

At this point, “Kawtar” started designing the website and working on the avoiding black line function to improve “An” algorithm.

To manage our work, we created a dropbox folder, in which we have shared our work and put some backups. We have also tried to meet each Thursday to test the code and discuss about progress of each one of us and also about the upcoming tasks.

Code

* Algorithms :
  + Attacker :
    - Scan the arena looking for the balls
    - Navigate to the opponent camp
    - Throw the ball
    - Avoid obstacles
  + Defender :
    - Navigate without crossing the delimitation black line.
    - Grabbing the red ball
    - Defending the goal
* Source code

The source code of our projects can be found on this [link](../Dropbox/Wall-E/robots/project_robot4)

* Instructions

To use our code, do as follows:

* Download our code
* Install the Bricx Command Center, a powerful IDE to compile and upload the code to your Robot ([download](http://bricx-command-center.software.informer.com/3.3/) ).
* Plug your Lego NXT into your computer and start the program downloaded above, be sure to chose “NXT” and “usb”
* Compile and download the program on your NXT, and you're done!