RoboCupJunior 2021

Team Description Paper 2021 - Soccer league

Members: Patrik Kralj, Borna Svržnjak, Ivan Matošević, Goran Vukmanović
Mentor: Dalija Milić Kralj

[1] HDR

[2] Croatia

dalija.milic-kralj@skole.hr

Abstract. This paper will introduce our team members in a few words, tell something about their past experiences with robotics competitions as well as their roles in this team. It will describe our robots and the simulated environment, in which programs they were made in, how we programmed them and the strategies which we chose. It will also shortly go over what we learned in the process and what we plan to improve in our programming in the future.

Keywords: RoboCupJunior, Webots, Python

1 Introduction

1.1 Team members

Our team consists of four members: Patrik Kralj, Borna Svržnjak, Ivan Matošević, Goran Vukmanović and our mentor, Dalija Milić Kralj (Fig. 1.). Patrik Kralj is 17 and lives in Dubrovnik. He previously participated in a few RoboCup competitions in both the Rescue Line and Rescue Maze categories. He is the team leader of ZTK Dubrovnik CRO team and programs the goalkeeper. Borna Svržnjak is 16 and lives in Varaždin. He won first place at the National RoboCup Croatia competition in the OnStage Primary category which qualified him for the World RoboCupJunior competition in Nagoya in 2017. He also won first place in the Soccer Beginner category at the RoboCupJunior event in Hanover in 2019. He programs the goalkeeper. Ivan Matošević is 15 and lives in Zagreb. He participated in multiple RoboCup competitions in Croatia, Canada and Germany, mainly in the OnStage and Rescue Maze categories. He programs the attackers. Goran Vukmanović is 16 and lives in Zagreb. He also participated in multiple RoboCup competitions in Croatia, Canada and Germany, mainly in the OnStage and Rescue Maze categories. He programs the attackers.

Fig. 1. ZTK Dubrovnik CRO team

2 Robots and Results

2.1 Software

Since this competition is held in a simulated environment, there is no hardware that needs to be mentioned, except of course the laptops and computers which support the environment. As for the software, two programs were used to simulate and code the football field and the robots. Webots is a free program which is used for simulations of various sorts. The model of the field and the robots as well as the physics engine were both made in webots (Fig. 2.). Of course, that code was downloaded from the [1]official RoboCupJunior Soccer Simulation website so every team has the same setup. [2]Python is the second program, which we used to make the code which would enable the robots to move as we wanted (Fig. 3.).

Fig. 2. Simulated soccer field

```python
import math


from rcj_soccer_robot import RCJSoccerRobot, TIME_STEP
import utils

class MyRobot1(RCJSoccerRobot):
    def run(self):
        while self.robot.step(TIME_STEP) != -1:
            if self.is_new_data():
                data = self.get_new_data()
                robot_pos = data[self.name]
                ball_pos = data['ball']

                ball_angle, robot_angle = self.get_angles(ball_pos, robot_pos)
                kut = math.degrees(robot_angle)

                x1 = ball_pos['x']
                y1 = ball_pos['y']
                y = ((0.53*(-y1)+y1*(10-x1)-x1*(-y1))/(10-x1))

                direction = utils.get_direction(ball_angle)
                if robot_pos['x'] > 0:
                    if robot_pos['x'] < 0.5:
                        if kut >= 260 and kut <= 280:
                            right_speed = 10
                            left_speed = 10
                        elif kut < 260 and kut > 90:
                            right_speed = -7
                            left_speed = 7
                        else:
                            right_speed = 7
```

Fig. 3. Python code

2.2 Results

Since this was our first time dealing with a simulation and we aren't very familiar with python, we wanted to keep the programming simple, but also implement some basic logic into it. We decided on having two attackers and a goalkeeper. The attackers follow the ball and rotate towards it at different speeds, depending on the angle between the ball and the robot. For a bigger angle, the robot will rotate faster. The goalkeeper is programmed in a way that, he always stays between the goal and the ball. This is achieved by taking the y coordinate of the ball and the y coordinate of the robot and moving the robot so that they are the same. Of course, during that time the robot's x coordinate always stays the same so he basically moves back and forth on a

straight line in front of the penalty area. At first all three of our robots were attackers, but since we simply didn't have the programming knowledge for a very successful attack, we decided that the best option was to have a goalkeeper to add some degree of protection against our opponent's attacks and the faults of our own attackers.

3 Conclusion

3.1 What we learned

While working on this simulation, we gained a fair bit of knowledge and experience working in python and webots. Of course it isn't a lot but since this is our first time doing something like this, there is a lot of room for improvement for next year. This was also our first time writing this kind of a document, something which will be very valuable to know in the world of engineering.

3.2 Future work

Of course, we know our programs aren't even nearly perfect. That means we intend to work and prepare even better for the next competition. We already have some improvements in mind. First, we would like to make our attackers target the opponent's goal while leading the ball, rather than just hope it happens on its own. We would also like for our goalkeeper to move on a curved line around the penalty area, not just in front of it. That way the robot could get from the beginning to the end of its path faster as it would be a bit shorter. He would also cover a smaller area, thus making more room for the attackers to do their own thing.

4 References

[1]https://robocupjuniortc.github.io/rcj-soccer-sim/how_to_robot.html

[2]https://realpython.com/