Development of a Code Interpreter and Simulator in Flutter/Unity for the Mobile Robot Yellow

Degree programme: BSc in Micro- and Medical Technology | Specialisation: Robotics
Thesis advisor: Prof. Dr. Gabriel Gruener
Expert: Jaqueline Staub (ETHZ)

Robot prices are dropping. Small and medium enterprises (SMEs) can afford buying a fixed or mobile robot. The programming interfaces remain often targeted at trained users. In this project, a simple and intuitive programming interface for a mobile robot was created, which can also be used to teach basic programming concepts. A simulation enables flexible training without requiring the physical robot.

Motivation

The BFH roboticsLab has its own mobile robot Yellow (figure 1). A simple programming interface has been already developed for android handheld devices, like tablets and smartphones. To use it on a broader array of operating systems the app was redeveloped in Flutter. Flutter is Google‘s UI toolkit for building applications for mobile, web, desktop, and embedded devices from a single codebase. The simple programming is based on Scratch, a visual programming language (VPC) developed by MIT.

Objective

The objective of this bachelor‘s thesis was to take an existing application and expand its features to add more complex programming capabilities as well as to integrate a robot simulation.

Programming complex robot behaviors in real time requires executing the instructions on the robot‘s on-board computer. The program written by the user on the handheld device needs to be encoded, transferred and interpreted on Yellow.

Results

The first results of the simulation show how Yellow interacts with objects in a virtual environment. The programming capabilities have been updated and new functionality was implemented. The BFH‘s own VPC ‚prograblocks‘ (figure 2) is used.

The program is parsed on the handheld device and sent to the robot encoded using protocol buffers (Google‘s serializing mechanism). The program is then decoded and executed on the robot.

Test will be done with all age groups. This experience will help improve the interface for more intuitive interaction as well as the simulation.

As a next step during the bachelor‘s thesis, the interpretation of the code will be integrated with the simulation. This will enable testing the program before it is executed on the actual robot.

Outlook

Custom functions can be added to enrich the versatility of the code, such as being able to modify properties of the control system and other parameters.

In future developments, the code can be used with industrial robots.

Eric Furest
eric@furest.ch

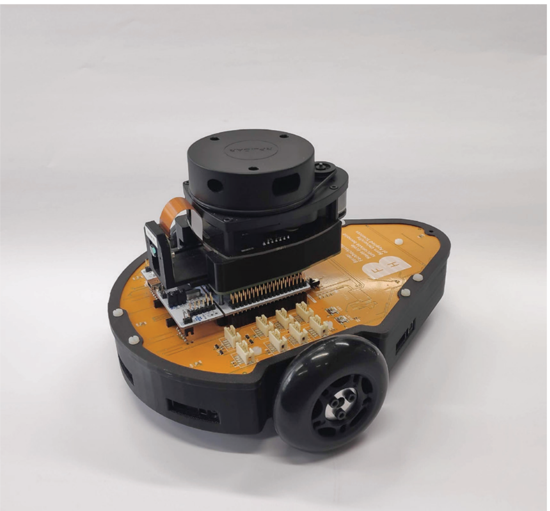
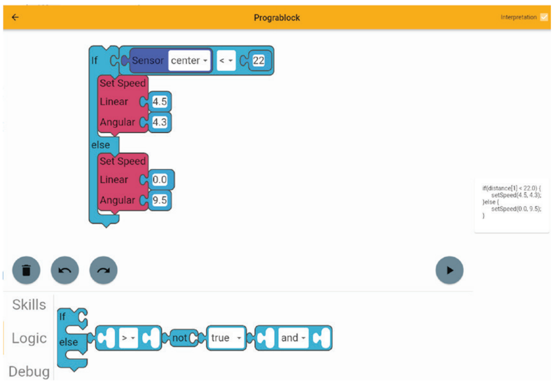


Figure 1: BFH mobile robot Yellow



```
if(distance[1] < 22.0) {
      setSpeed(4.5, 4.3);
}else {
      setSpeed(0.0, 9.5);
}
```

Figure 2: Prograblock program