Final Project Results

Title: Mobile Robot Control System with Application to Visualize its Trajectory.

Class: Tue 15:15
Tutor: PhD Witold Paluszyński

Wroclaw University of Technology

Author: **Mikołaj Przybylak**

February 9, 2022

1 Project Introduction

The main goal of the project was to create fully functional system to control mobile robot and visualize its trajectory.
System consists with: PC application, mobile platform and platform controller. After finished system allows to control the platform from the application on PC using simple commands like; forward and backward, as well as it allows to set certain points and orientation for platform to reach. Controller of platform should response for this signals and for the second task it should calculate reverse kinematics. Additionally the PC application should visualize trajectory of the platform.
Tasks assumptions:

- the application to send commands to robot runs on PC computer,
- the PC app is written in c++ in framework QT,
- the platform that is used is already developed,
- the robot controller runs on Raspberry Pi and it is written in Python,
- a communication between the PC app and the robot is based on Wifi,

2 Application

Application to control entire system runs on PC computer. It is written in c++ language using QT framework. Application allows to:

- connect to robot via Wifi,
- check the status of platform connection,
- send simple commands,
- send coordinates and orientation to reach by a robot,
- visualize trajectory of robot.

Main window of application is shown on Figure 1.

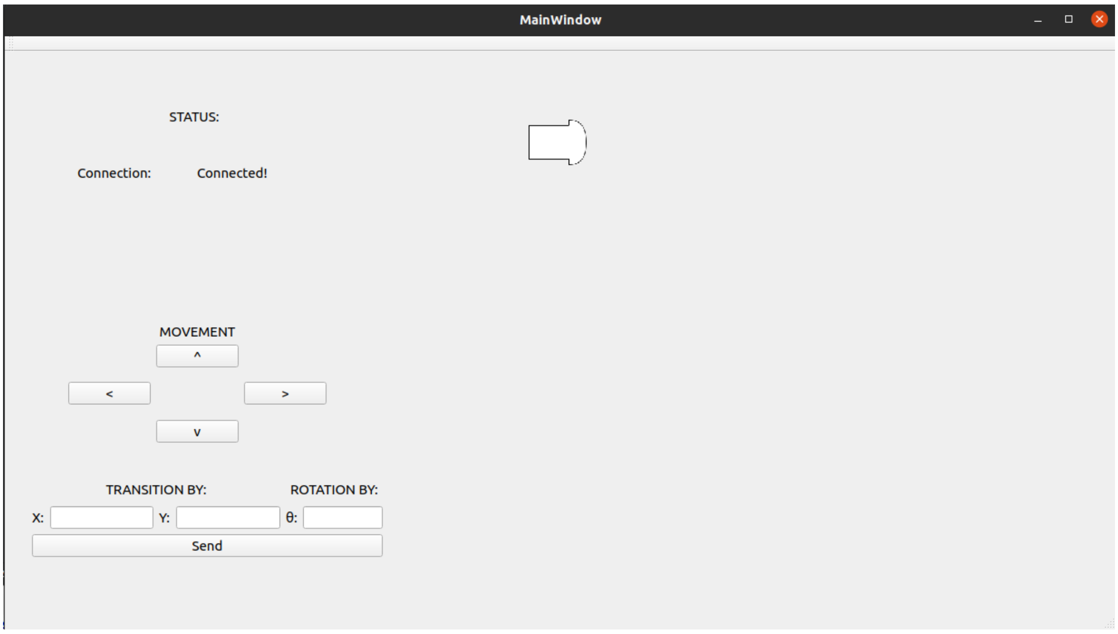

Figure 1: Main window of application.

2.1 Functionality of Each Section

2.1.1 Status

Shows status of connection to the platform.

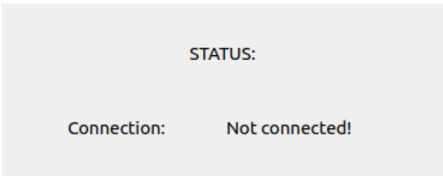

Figure 2: Status in the application.

2.1.2 Movement

Allows to send simple commands to platform. Commands:

- Go forward.
- Go backward.
- Turn right.

- Turn left.

Buttons are monostable but works in two states, when pressed, robot is moving in chosen direction, when released the platform stops.

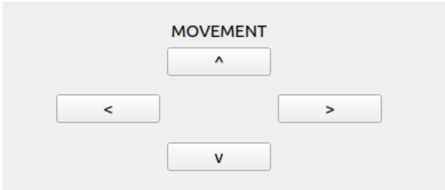

Figure 3: Movement in the application.

2.1.3 Position and Rotation Commands

This section allows to send coordinates and rotation to platform. After receiving, platform will reach the position and finish movement with rotation set. It is accomplished by calculating revers kinematics on the platform side.

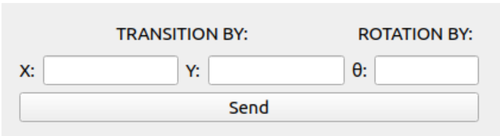

Figure 4: Position and orientation in the application.

2.1.4 Trajectory

Trajectory section display path that platform has traveled. Data that application receives from platform are impulses from encoders, using this data it calculates forward kinematics and visualize trajectory of the platform. Sample trajectory generated by movement buttons is show on Figure 5.

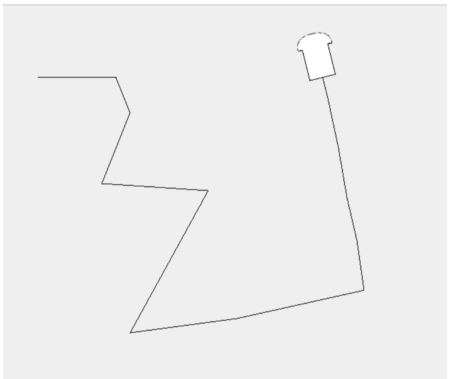

Figure 5: Sample trajectory.

3 Mobile Platform (Robot)

Platform that was used in this project is shown on Figure 6. It was created as other project subject a few years before. On the Figure 6 we can see initial state of platform. This platform is differential drive mobile robot, which has two additional wheels at the rear section to keep balance. Initially platform consisted with:

- frame,
- dc motors with wheels and encoders on the shaft,

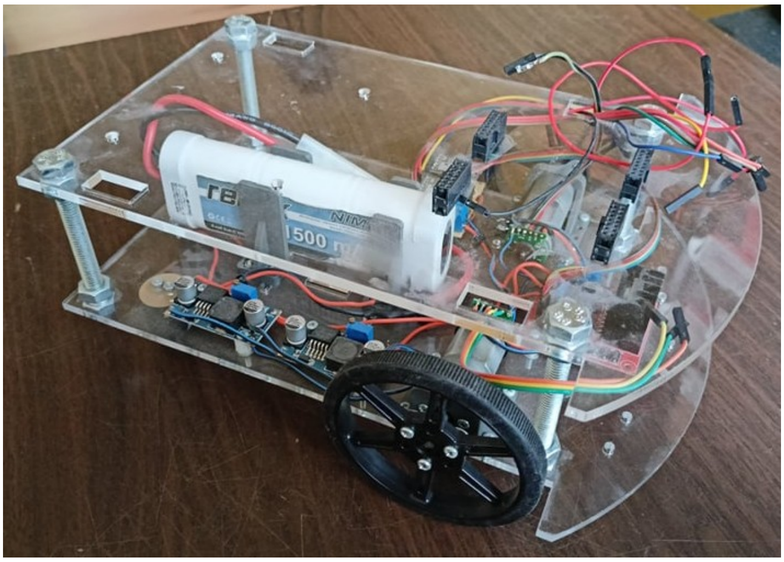

Figure 6: Platform.

- accumulator with power management section,
- dc motor controller.

During this project Raspberry Pi computer was added at the top of the platform in order to control it. Figure 7 shows platform with control unit mounted and connected to it.

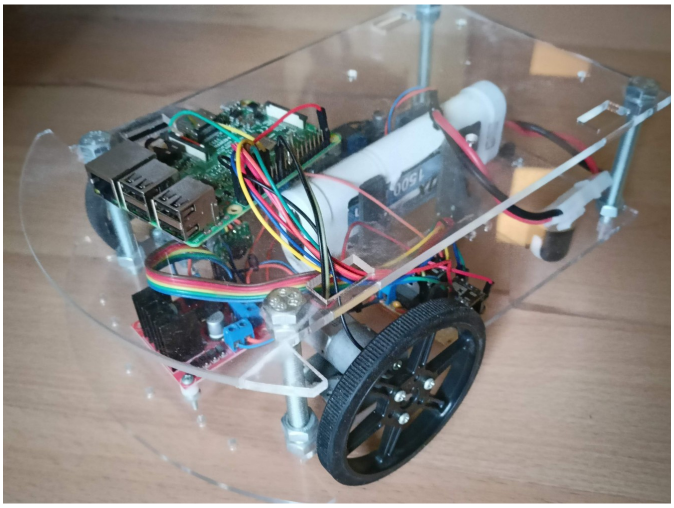

Figure 7: Platform with computer on it.

4 Robot controller

Raspberry Pi computer which it the controller of the platform runs Linux operating system. Software which is controlling whole platform is written in Python programming language. Hardware that is controlled by this computer is: two encoders and two channels dc motors controller. Controller realize two tasks: communication with PC application and platform steering. Platform steering is about calculating PID control for motors, and calculating reverse kinematics.

The robot controller uses state machine manner of controlling its functionality. On the beginning is set up TCP server to accepts connection with application and then enters state machine function. States that are possible:

- Stop

- Move (forward, backward, left, right)
- Reaching Point (calculate reverse kinematics and movement)

5 Communication

Communication between PC application and platform is held by Wifi interface. It is based on TCP sockets. The computer on the platform works as a server and application as a client. Client send control signals to the server. There is two types of commands:

- Simple commands like: "Forward", "Stop" etc.
- Complex commands, used for position and orientation.

6 Results

Everything from assumptions of the project was accomplished except for reverse kinematics. Application works fine, it is able to communicate with platform, shows trajectory of path travelled by platform (forward kinematics), and it can send commands. Platform is able to perform movement received form PC application, PID controller can control engine so platform can move forward following the straight line. Calculating reverse kinematics, which had to be implemented on platform computer, was not accomplished.

Test that was carried out on the platform is measure of absolute difference in impulses form encoders in cycles of PID controller. One cycle of the controller is calculated every 10ms. On the Figure 8 differences in impulses are shown per every 10 cycles.

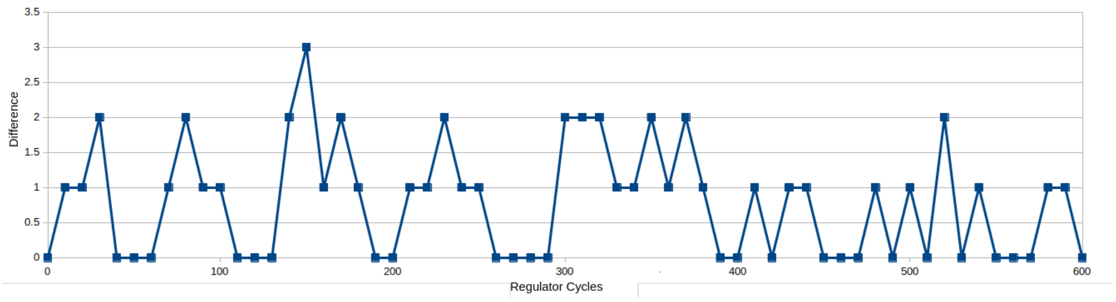

Figure 8: Absolute Difference in Impulses from Encoders

7 References

- Lecture on mobile robots kinematics, Roland Siegwart, Margarita Chli, Martin Rufli, 2027
- de Jesús Rubio, J., Aquino, V. Figueroa, M. Inverse kinematics of a mobile robot. Neural Comput Applic 23, 187–194 (2013). https://doi.org/10.1007/s00521-012-0854-0