Igor Trotskii: 050388617

Trung Van: H293914

Exercise 5 - Report

# Status:

Task 5.1 Finished

Task 5.2 Partially implemented

Task 5.3 Finished

# Exercise 5.1 Circular obstacle avoidance by using switched controller.

ugtg, uavo are obtained by using usual proportional controllers with variables gain (which is dependent either on the distance to the goal or on the distance to the obstacle). dsafe is calculated as a sum obstacle radius, small safety margin (0.01) and robot radius. The figure displays obstacle as red, safety margin as yellow and robot’s dimensions are only considered within calculations.

Chart

Description automatically generated

Figure 1. Controller output vs time.

Chart

Description automatically generated

Figure 2. Errors vs time.

Chart, line chart

Description automatically generated

Figure 3. Distance to the obstacle over time.

Chart, line chart

Description automatically generated

Figure 4. State trajectory and target over time.

Chart, bubble chart

Description automatically generated

Figure 5. XY trajectory of the robot.

The controller does a relatively good job at avoiding the obstacle. But by doing so, it causes a lot of fluctuations in in changing it direction during the actual avoidance as it can be seen from Figure 1 and 2. These results into “jigging” movement of the robot and can be harmful.

# Exercise 5.3 QP-based controller

Chart, line chart

Description automatically generated

Figure 6. Control inputs for y(h) = 0.2

Chart, line chart

Description automatically generated

Figure 7. h(x) for y(h)=0.2

Chart, scatter chart, bubble chart

Description automatically generated

Figure 8. Robot’s trajectory for y(h) = 0.2

Chart, line chart

Description automatically generated

Figure 9. Control input for y(h) = 10

Chart, line chart

Description automatically generated

Figure 10. h(x) for y(h) = 10

Chart, scatter chart, bubble chart

Description automatically generated

Figure 11. Robot trajectory for y(h) = 10

As it can be seen from the figures above, lower y makes the controller a lot more responsive and so makes the robot to act early to avoid the obstacle and so deviate more from the optimal route. Higher y values might be preferable in tight spaces with static obstacles.

# Exercise 5.2 Wall-following behavior

For this task, we didn’t finish all the requirements. There was a problem with applying all the conditions to the controllers. It seems not to switch to a suitable control. We have implemented the obstacle detections with sensors and individual parts of the controller.

# Implemented code

To be attached with the submission file or can be seen from this github link:

https://github.com/VANHUUTHANHTRUNG/MoRo5