Autonomous Mobile Robots
Homework 2

Ahmed Mohsen ALi

October 2022

Task Description

In this task, I try to execute the reference path following using control points. The points are $[[3,0],[6,4],[3,4],[3,1],[0,3]]$ which are used as T_i and T_{i+1} for each path segment with duration = 50, $k_p = 0.4$, and $k_\phi = 3$.

Problem solution

Initially the robot is assumed to start at pose: $[0,0,0]$ and moves to first control point. In order to know which path segment to follow, this formula is calculated:

$$u = \frac{V^T r}{V^T V}$$

Also, the orthogonal distance between the robot and current path segment is calculated by projection r into V_n so that $d = \frac{V_n^T r}{V_n^T V_n}$. To determine the control laws, the orientation and rotation angles are:

$$\phi_{lin} = arctan2(\Delta V_y, \Delta V_x)$$

$$\phi_{rot} = atan(k_r d)$$

$$\phi_{ref} = \phi_{lin} + \phi_{rot}$$

$$e_\phi = \phi_{ref} + \phi$$

So the appropriate controller in this case will be:

$$v = k_p \, cos(e_\phi)$$

$$\omega = k_\phi \, e_\phi$$

The terminal condition will be either:

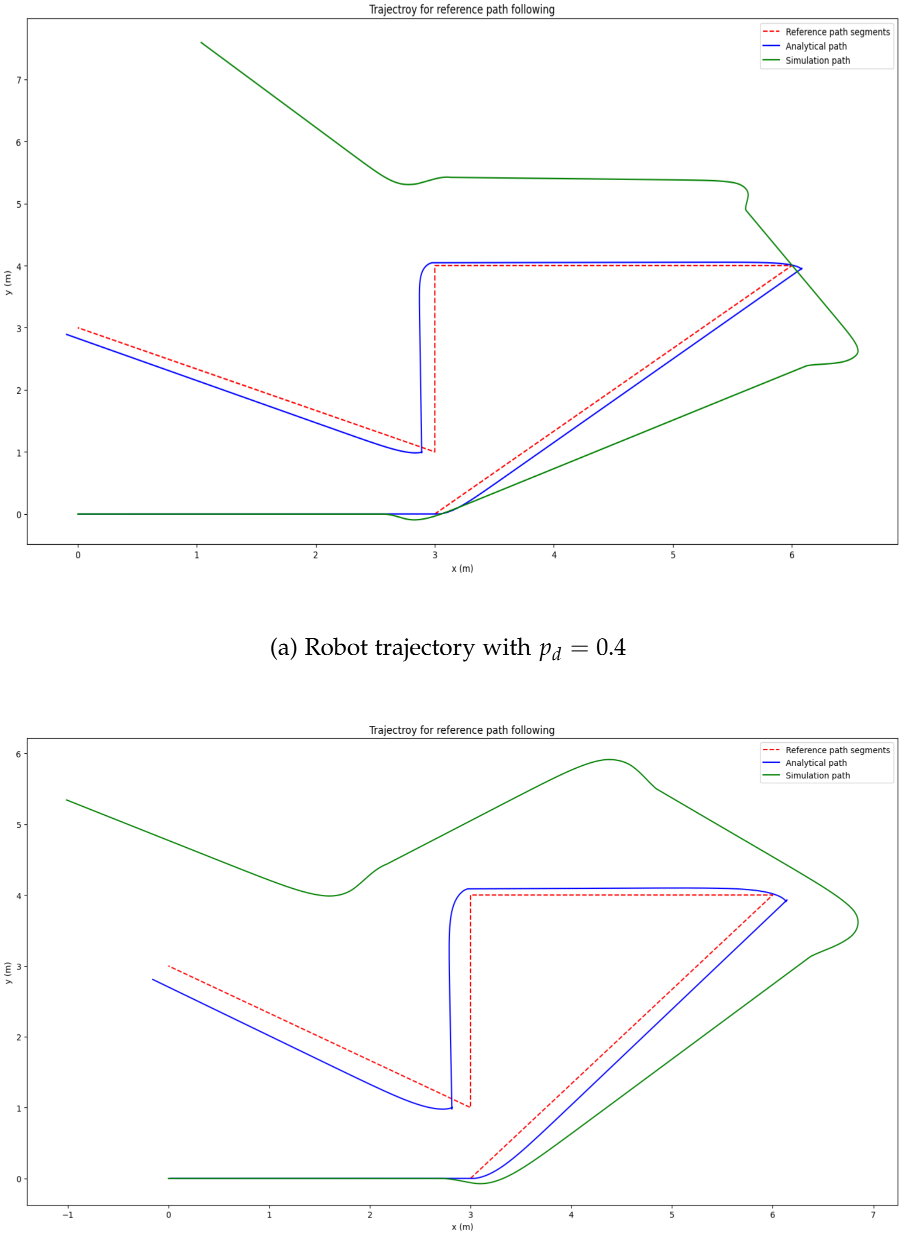
- robot reaches final control point.
- simulation time exceeds input duration.

The software used are :

- ROS2 humble on Ubuntu 22 (Docker container).
- Python3 scripts.
- Gazebo and Rviz for simulation.

Results and Discussion

The following graphs show the robot trajectory with the same k_ϕ but with different k_d. The simulation data are obtained from Gazebo while analytical data are generated from python script.



(a) Robot trajectory with $p_d = 0.4$

(b) Robot trajectory with $p_d = 0.7$

Figure 1: Trajectory data for different k_p gains

As seen in the figure 1, both analytical and simulation data are different from each other and from the reference path itself. This dilemma occurs for two distinct reasons:

1. **Analytical data** are different from reference path because the controller takes some time to change the robot orientation and since the reference path has some abrupt changes in orientations, the controller can not match it simultaneously. At the end, the controller manages to reach the correct orientation but with a little offset from the reference segment.

2. **Simulation data** are different from the expected shape since there is no feedback controller and we do not know whether the control signals are sent exactly to the robot or not. This issue happens due to noisy data, friction, or/and slipping.

Remarks

- Due to the abrupt changes in orientation that the robot must follow, it is recommended to use high values for k_{ϕ}. Low values for this gain results in a worse performance.

- feedback controller will enhance the accuracy of overall model and reduce the difference between the simulation data and analytical data