Lab 07

Motion Planning and Control using Pure Pursuit

Objectives

- Implement the Pure Pursuit algorithm in a Python script
- Run a path following simulation with the implemented script
- Understand the limitations of the Pure Pursuit algorithm

Download the material

Download the file `lab07_material.zip` from Portale della Didattica.

Exercise

Task 1

Open `pure_pursuit_students.ipynb` and follow the instructions in the notebook that guide you through the implementation of a Pure Pursuit control algorithm.

You will find the concrete exercise of this lab,starting from the definition of the Differential Drive Motion Model and the Velocity Update rule.

For a working program, you should complete the notebook where indicated, in particular, you have to:

- Complete the Differential Drive Motion Model.
- Write the velocity update funcion f(v, a).
- Complete the `velocity_update` and `update_state` methods of the Robot class where needed.
- Complete the `proportional_control` function when needed.
- Complete the `lookahead_point`, `target_velocity` and `angular_velocity` methods of the `PurePursuitController` class when needed.
- Complete the `run_simulation` function.

Once you complete the notebook, you should obtain an animation of the robot moving, controlled by the Pure Pursuit algorithm.

Save a gif of the output changing the `save` parameter to True in the last cell.

To visualize the result of this experiment you can plot **linear velocity** (hint: compare it with the target velocity), **angular velocity**, **acceleration** and **trajectory**, comparing it with the target path.

Task 2

Using the provided script, run the simulation using different combinations of proportional coefficient Kp and lookahead distance Lt.

Experiment: Try at least the following values and comment on the results (numerically and graphically).

- **Kp**: 0.1, 0.3, 1
- **Lt**: 0.5, 1.0, 5.0 [m]

Compare the results with the plots suggested in the previous task.

Task 3

Using the provided script, run the simulation using different combinations of target velocity.

Experiment: Try the following values and comment on the results (numerically and graphically).

- **vt**: 0.05, 0.1, 0.5 [m/s]

Compare the results with the plots suggested in task 1.

Task 4

Using the provided script, run the simulation change the frequency at which the control input is computed.

Experiment: Try at least the following values and comment on the results, (numerically and graphically).

- **controller_dt**: 0.2, 0.5, 1.0 s

Compare the results with the plots suggested in task 1.

Task 4

Using the provided script, run the simulation changing the trajectory (Hint: change the list of waypoints to obtain a different path and use a function to sample points from a geometric trajectory).

Experiment: Try at least a circular trajectory and a star like one.

Compare the results with the plots suggested in task 1.