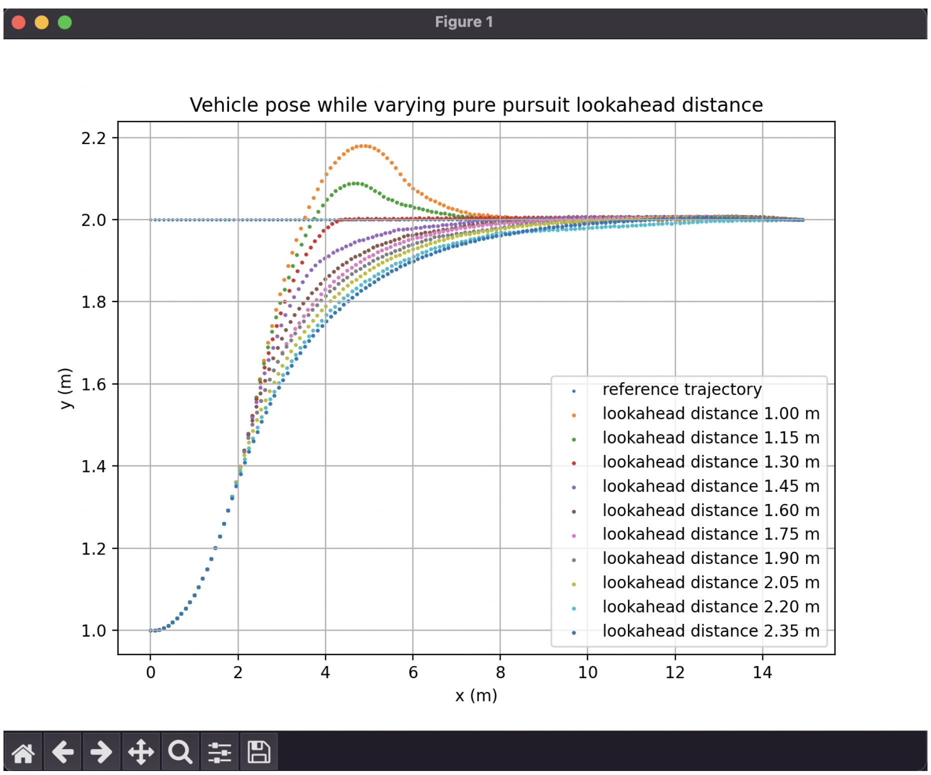
Report 9 - Nihal Afsal

PID and PPC Control in Carla

Note: Homework must be uploaded as a **single pdf file**, not a zip file. If a problem solution requires a video, add it as a hyperlink in the pdf. The hyperlink should open the video file which is stored on your Google Drive. Any problem that requires Python code must show the entire code as well as a description of how the code works. Duplicate code submissions will result in a zero.

1. Compare 10 unique lookahead distance values for the Python PPC algorithm shown in lecture. Show the results in a well formatted Python plot. *(12.5 pts)*



2. Compare 5 unique P values, 5 unique I values, and 5 unique D values values for the Python PID control algorithm shown in lecture. Show the results in a well formatted Python plot. *(12.5 pts)*

- *P Values:*

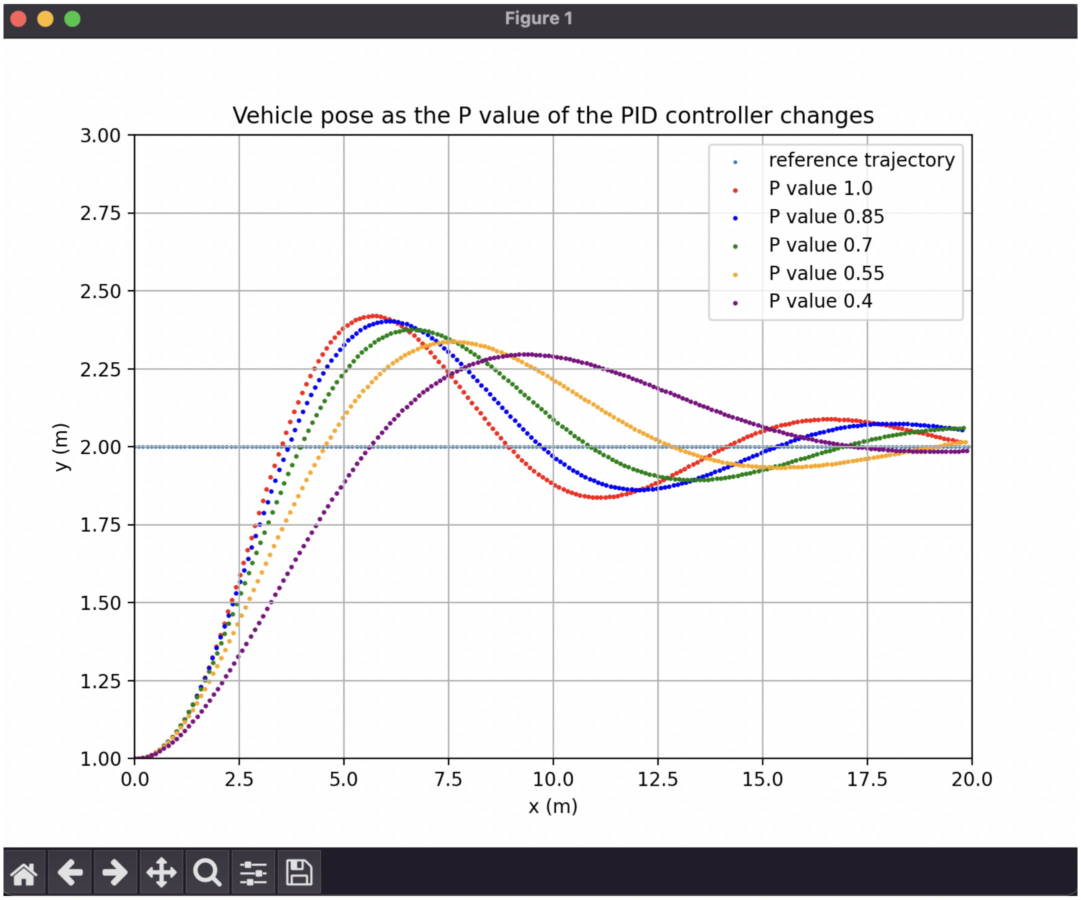


Figure 1
Vehicle pose as the P value of the PID controller changes
reference trajectory
P value 1.0
P value 0.85
P value 0.7
P value 0.55
P value 0.4
y (m)
x (m)
3.00
2.75
2.50
2.25
2.00
1.75
1.50
1.25
1.00
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

- *I Values:*

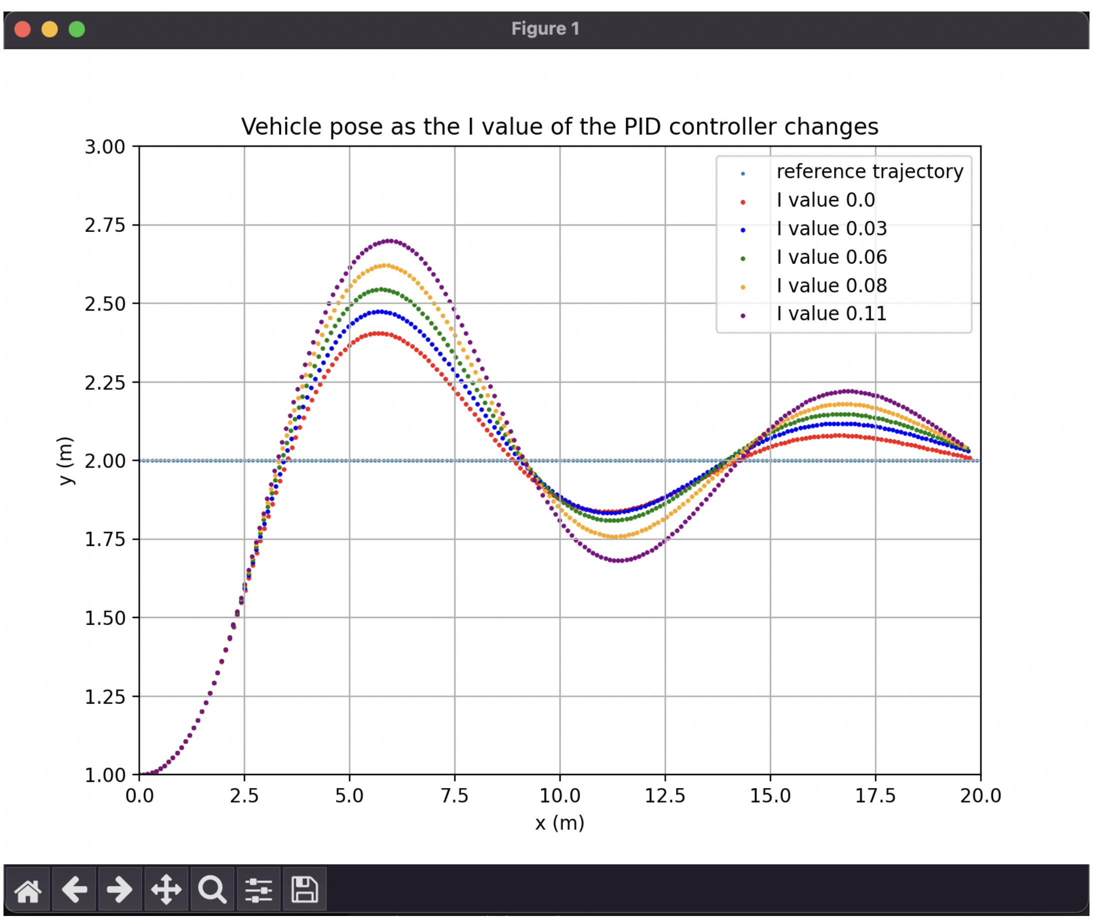
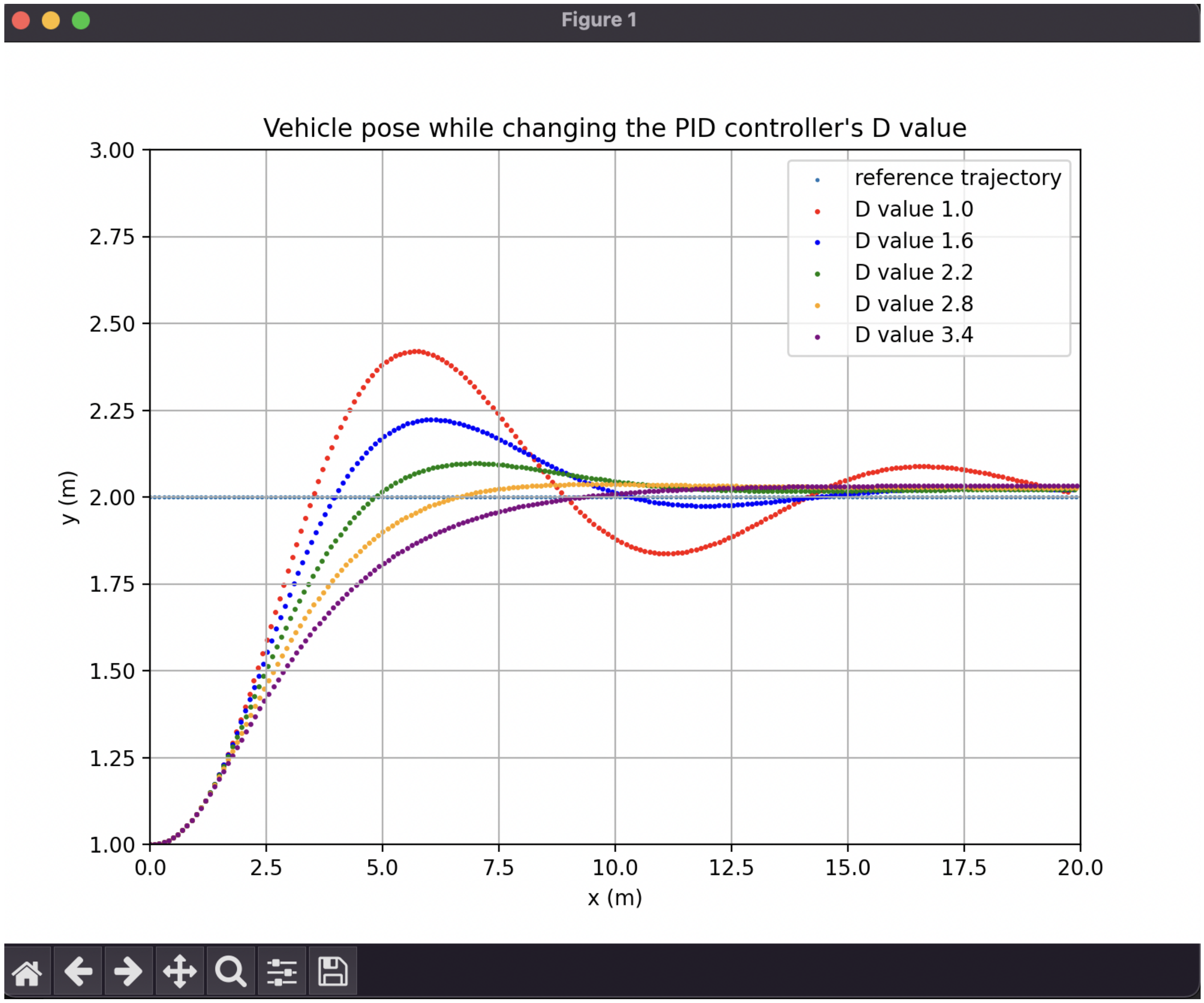


Figure 1
Vehicle pose as the I value of the PID controller changes
reference trajectory
I value 0.0
I value 0.03
I value 0.06
I value 0.08
I value 0.11
y (m)
x (m)
3.00
2.75
2.50
2.25
2.00
1.75
1.50
1.25
1.00
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

- *D Values:*

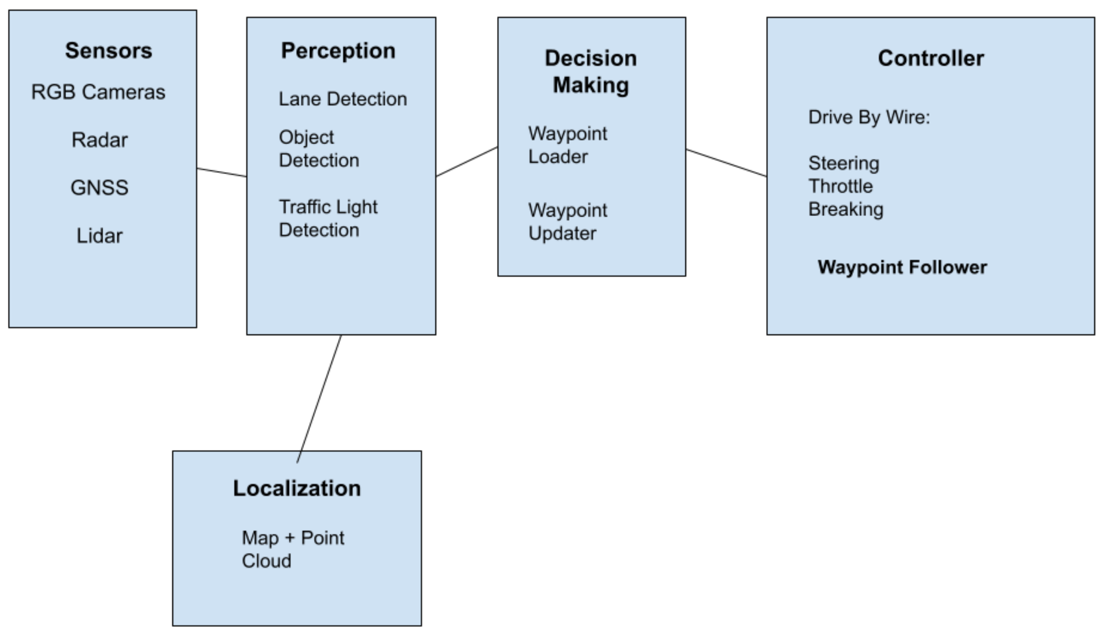


3. Use the AD demo files that came with the Carla ROS bridge and show a video of at least 2 sensor outputs from the simulator in real time in rviz. *(15 pts)*

- *Lidar and Radar!*

- https://drive.google.com/file/d/1ipNQZ5QOpTggsX4RUetugA-AQvnYeC4t/view?usp=sharing

4. Describe and sketch out how you will combine a perception algorithm with a control algorithm to drive autonomously in Carla for the final project. Use a simplified version of a flow diagram like this one using a few blocks for the sensors and controls. Try to include specific ROS information as best you can. *(15 pts)*



- *In order to supply information to perception algorithms that will determine traffic signals, the locations of objects, and lane lines, our vehicle will use an RGB Camera, Radar, GNSS, and LIDAR. The waypoints for these objects are transmitted to the system's decision-making area, where the waypoint updater creates new waypoints. The control section receives the new waypoints and passes them along to the waypoint follower, who then passes the adjustment instructions to the drive-by-wire section, which applies them to the steering, throttle, and breaks as required to guide the vehicle.*

5. Using the files provided on Elearning, create a Python plot comparing the recorded (x,y) vehicle locations to the achieved (x,y) vehicle locations using a PPC. Compare at least 2 different PPC algorithms that use different lookahead distances and/or speeds. *(30 pts)*