Abstract:

In this project we will get hands on with a simulation of an autonomous vehicle that requires longitudinal and lateral vehicle control design to track a predefined path along a racetrack with a given speed profile.

Learning Objectives:

- Integrate vehicle modeling and controller design into a complete vehicle control system.
- Develop a working simulation with a python-based vehicle autonomy agent.
- Tune a control system for tracking performance on a complex path.

The repository for project:
https://github.com/Padmanabha123/Self_Driving_Using_carla/tree/main/Part%201%20-%20Introducton%20to%20self%20driving%20car%20with%20report

Requirements:

1) In this project, we will write and implement a controller for the CARLA simulator. Our goal is to control the vehicle to follow a race track by navigating through preset waypoints. The vehicle needs to reach these waypoints at certain desired speeds, so both longitudinal and lateral control will be required.

To do this project, the CARLA simulator along with the assignment code needs to be installed.

2) Download the above repository and unpack into the subfolder folder "PythonClient" inside the "Carla Simulator" (root) folder. This will create a subfolder "Course1FinalProject" under "PythonClient" which contains the project files.

It is very important to have the contents of "Course1FinalProject.zip" be under the folder "PythonClient\Course1FinalProject". Installing it into another directory might cause runtime issues.

After successfully downloading CARLA and the assessment script, you can now begin the Project.

3) The "controller2d.py" file contains a controller object. We have implemented the controller in the update_controls method labeled by the comment blocks.

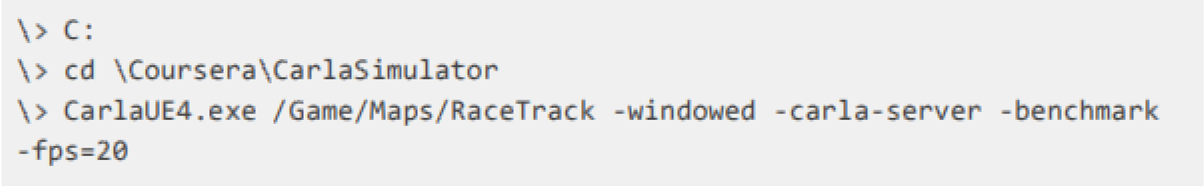
The controller provides you with the following relevant information required for its implementation. All units are in **SI** (meters, seconds, radians), and CARLA works in the **left-handed coordinate system** (due to the Unreal Engine adopting the left-handed coordinate system**).**

4) The waypoints variable is a Python list of waypoints to track where each row denotes a waypoint of format [x, y, v], which are the x and y positions as well as the desired speed at that position, respectively. More details on the structure of the waypoint variable is written in the comments of the controller2d.py file.

Along with the other variables, the waypoints will update on each simulation step - so please do not assume the waypoint variable never changes. Here the waypoints are a **linearly interpolated (for location and speed) subset of the entire set of waypoints** (from racetrack_waypoints.txt). In other words, the waypoints variable is an enhanced (finer resolution) portion of the entire set of waypoints that is near the vehicle. This is done to reduce the computation time and the performance of the controller.

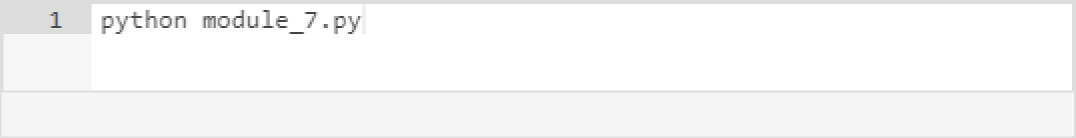
Procedure:

1) In one terminal, start the CARLA simulator at a 20hz fixed time-step:



```
\> C:
\> cd \Coursera\CarlaSimulator
\> CarlaUE4.exe /Game/Maps/RaceTrack -windowed -carla-server -benchmark
-fps=20
```

2) In another terminal, change the directory to go into the "Course1FinalProject" folder, under the "PythonClient" folder.

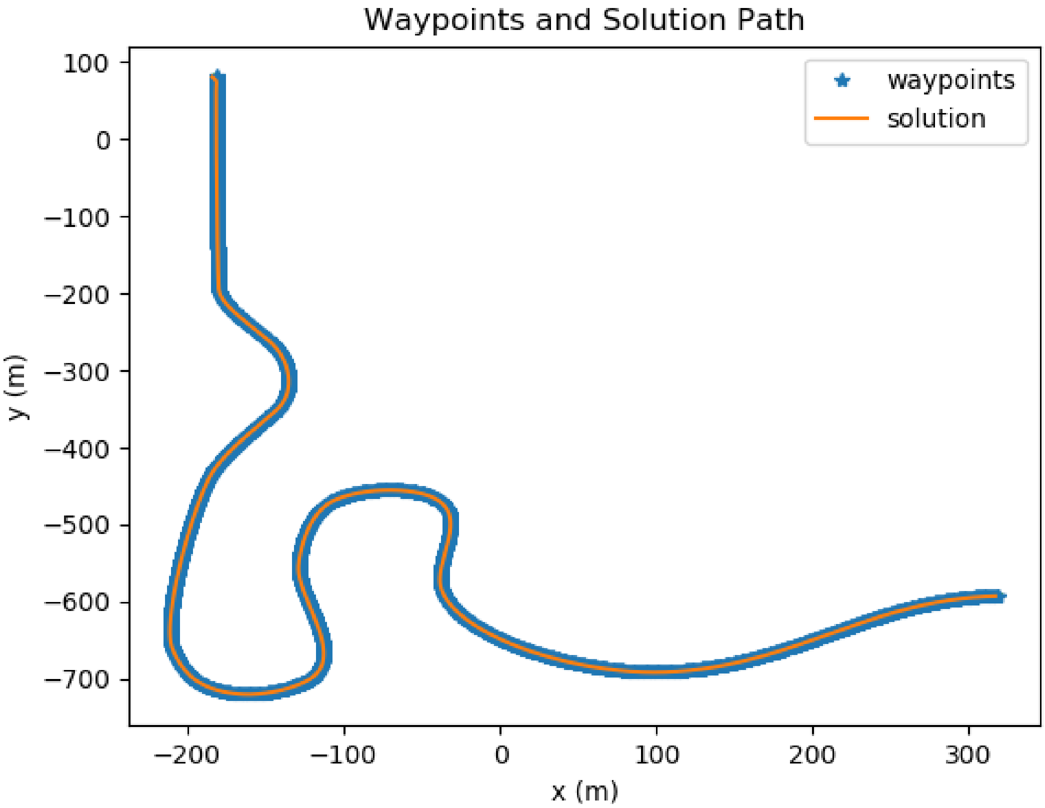


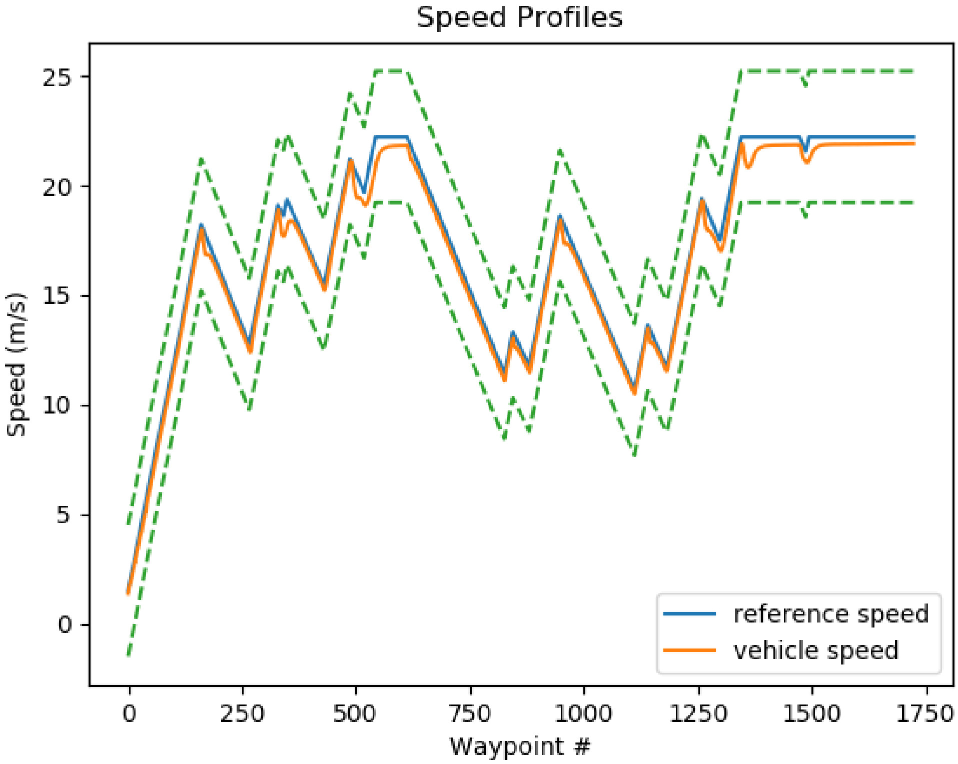
```
python module_7.py
```

- The simulator will begin to run if the module_7 client connects to the server properly.
- The trajectory feedback will contain the car, start and end positions, entire path/path traveled and a small shaded region, which denotes the subset of interpolated points to be sent into the controller for control updates. Linear interpolation is used between waypoints to provide a finer resolution path/speed requests for the controller. The X and Y-axes are in meters.
- The controls feedback shows the throttle, steering and brake outputs, as well as the speed response for the simulation (desired speed and current speed in the single plot). This is a general feedback for viewing what the client is sending to the CARLA server in terms of control commands. The desired speed is set to the closest interpolated speed point to the current position of the car. The speeds are in meters per second and the throttle (0 to 1), brake (0 to 1) and steering (-1 to 1, or left to right turn) are unit less. Note that the steering command output inside controller2d.py is automatically converted from radians (-1.22 to 1.22 rad) to a percentage (-1 to 1) before the command is sent to the CARLA server. The X axis for all four plots in the controls feedback is the in-game time, in seconds.

<u>YouTube video link</u>: https://youtu.be/MsI7zZgifTY

Results:


Waypoints and Solution Path
waypoints
solution
y (m)
x (m)
100
0
−100
−200
−300
−400
−500
−600
−700
−200
−100
0
100
200
300


Speed Profiles
reference speed
vehicle speed
Speed (m/s)
Waypoint #
25
20
15
10
5
0
0
250
500
750
1000
1250
1500
1750