UDACITY

PROJECT

Path Planning

A part of the Self-Driving Car Engineer Program

PROJECT REVIEW

CODE REVIEW 5

NOTES

SHARE YOUR ACCOMPLISHMENT!

Meets Specifications

Dear Udacious Learner, This work is very impressive and am glad to have reviewed it. Also the codes are clearly written and accompanied with an elaborate readme. This application is indeed intelligently developed and I must say this car can serve as a model for most cars out here . I must endorse the hard work and determination I perceived in this implementation and want this spirit to be maintained through out learning with us here are Udacity.

More In-depth Knowledge

- http://ais.informatik.uni-freiburg.de/teaching/ss10/robotics/slides/16-pathplanning.pdf
- http://correll.cs.colorado.edu/?p=965
- http://www.coppeliarobotics.com/helpFiles/en/pathPlanningModule.htm
- https://www.cs.cmu.edu/afs/cs/project/jair/pub/volume9/mazer98a-html/node2.html
- http://www.roborealm.com/help/Path_Planning.php

Compilation

Code must compile without errors with `cmake` and make.

Given that we've made `CMakeLists.txt` as general as possible, it's recommend that you do not change it unless you can guarantee that your changes will still compile on any platform.

Well done with this submission. The project compiles error free with just these two commands

Valid Trajectories

The top right screen of the simulator shows the current/best miles driven without incident. Incidents include exceeding acceleration/jerk/speed, collision, and driving outside of the lanes. Each incident case is also listed below in more detail.

The car doesn't drive faster than the speed limit. Also the car isn't driving much slower than speed limit unless obstructed by traffic.

Awesome! The ego car does not exceed the speed limits during driving and that was well done. It is smart to know when to slow down and when to speed up.

The car does not exceed a total acceleration of 10 m/s^2 and a jerk of 10 m/s^3.

Nice work. The ego car is cautious enough not to exceed the acceleration and jerk as required in this section.

The car must not come into contact with any of the other cars on the road.

This is intelligent. The car reduces its speed when behind a slow vehicle and also changes its lane if adjacent lanes are free. It never collided with other cars in the course of this simulation. Nothing more could be asked here.

The car doesn't spend more than a 3 second length out side the lane lanes during changing lanes, and every other time the car stays inside one of the 3 lanes on the right hand side of the road.

The car was always on the lane during driving and never spent more than 3 seconds out of a lane. Nicely done!

The car is able to smoothly change lanes when it makes sense to do so, such as when behind a slower moving car and an adjacent lane is clear of other traffic.

Awesome! the ego car was smart enough to know when to switch lanes especially when behind a slower car. It always ensured that the new lane it is headed is free of other cars before making the move. This was intelligently done.

Reflection

The code model for generating paths is described in detail. This can be part of the README or a separate doc labeled "Model Documentation".

This `readme` submitted in adequately describe the reasoning behind this implementation. The ideas are really intriguing and highly commendable.

DOWNLOAD PROJECT

5 CODE REVIEW COMMENTS ›

RETURN TO PATH

Student FAQ