UDACITY

PROJECT

Model Predictive Control (MPC)

A part of the Self-Driving Car Program

PROJECT REVIEW

CODE REVIEW 2

NOTES

SHARE YOUR ACCOMPLISHMENT!

Requires Changes

1 SPECIFICATION REQUIRES CHANGES

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.

Implementation

Student describes their model in detail. This includes the state, actuators and update equations.

Student discusses the reasoning behind the chosen *N* (timestep length) and *dt* (elapsed duration between timesteps) values. Additionally the student details the previous values tried.

Good observation. In general, smaller dt gives better accuracy but that will require higher N for given horizon (N*dt). However, increase N will result in longer computational time which effectively increase the latency. The most common choice of values is N=10 and dt=0.1.

A polynomial is fitted to waypoints.

If the student preprocesses waypoints, the vehicle state, and/or actuators prior to the MPC procedure it is described.

You fitted 3rd order polynomial which is correct. However, you forgot to update the equation of psides0. The term inside atan should be derivative of 3rd order polynomial but you're still using derivative to linear equation from the exercise.

```
AD<double> f0 = coeffs[0] + (coeffs[1] * x0) + (coeffs[2] * x0 * x0) + (coeffs[3] * x0 * x0 * x0);
AD<double> psides0 = CppAD::atan(coeffs[1]);
```

The student implements Model Predictive Control that handles a 100 millisecond latency. Student provides details on how they deal with latency.

You have implemented update equations to predict the states 100ms into future before sending them to MPC. Therefore the solutions returned by MPC will be meant when the actuators receive them after the latency has elapsed. Well done!

Simulation

No tire may leave the drivable portion of the track surface. The car may not pop up onto ledges or roll over any surfaces that would otherwise be considered unsafe (if humans were in the vehicle).

The car can't go over the curb, but, driving on the lines before the curb is ok.

Your car was very fast with speed of over 90mph! However, the MPC waypoints appeared to be like a straight line where it is supposed to curve smoothly and that's because you're still using linear equation inside atan(). The MPC waypoints also swung violently and in one occasion the tires touched the curb slightly.

RESUBMIT

DOWNLOAD PROJECT

2 CODE REVIEW COMMENTS ›

Best practices for your project resubmission

Ben shares 5 helpful tips to get you through revising and resubmitting your project.

Watch Video (3:01)

RETURN TO PATH

Student FAQ