Reflection:

PID Controller Project

Self-Driving Car Engineer Nanodegree

Rafael Barreto

INTRODUCTION

In this project, two controllers were developed, a PID controller for controlling the steering wheel angles and a PD controller for controlling the acceleration of the vehicle. The steering wheel angles control is aimed at keeping the vehicle in the middle of the track, the error corresponding to the difference between the target (middle of the track) and the measurement value given by the sensor. This error is called CTE (cross track error. The car's throttle control aims to maintain an average speed of 15 mph along the route.

DEVELOPMENT

The first attempts to tune the parameters Kp, Kd and Ki were choosing low values of Kp and Kd for the observation of how the car behaves with the change of each parameter. The Ki term was kept zero. After this, an algorithm for the automatic tuning of the parameters was developed. The algorithm chosen was Twiddle as the lessons of the project suggested. The algorithm was implemented and after several attempts, no satisfactory result was achieved. In one attempt the values of Kp and Ki were very elevated. The car managed to complete the track but oscillated quite mainly in the last corner. After several attempts, I tested dt as a constant, in the case equal to 1, and the car was able to complete the circuit much better.This approach of considering the time difference between measuments constant and set the value of dt equal 1 is valid because the derivative and the integral erros can be adjusted only in fuction of the Kd and Ki. For instance, if the value chosen od dt is 10 the new Kd Kd_new should 10 times bigger than the old Kd. And the Ki_new = 0.1xKi_old. So the tune is easier with dt constant equal 1.

After several attempts and the perception that I did not succeed in the calculation of approximate dt using the library <ctime> I considered it better to manually tune the parameters Kp, Kd, Ki.

PID TUNING

Kp:

It has been verified that the car can make smooth and correct curves in the circuit with “low” values of Kp, for instance, between 0.5 and 2.0. Larger values such as Kp > 5 cause the degrade the stability because the response of the steering is very fast. So the Kp was set to 1.

Ki:

It was verified that typical values for this constant are quite small[1]. During the twiddle tuning process big values of Ki decrease too much the stability of the cat greater than 0.001 for example can made the car swing quite a bit and even the output of the track. Therefore Ki values such as 1e-5 or 1e-6 showed good results. Ki was then set to 1e-5. The Ki error i_error is zeroed if the CTE is approximately zero or when the CTE is a big value as 1 as suggested by the tutorial [1].

Kd:

With the values of Kp and Ki fixed Kd was increased from 0.01 to 30.0. Low Kd values caused the car to recover with oscillation around the center after passing the middle of the road (Case Underdamped [2]). With higher values of Kd the oscillations after pass de middle of the road decreased. In this project, values greater than Kd> 5 were successful in the circuit. The value of 20 was chosen due to a good performance of this value.PD Tuning

The control of the throttle is based on simple PD Controller. The Kp and Kd values were set to 0.5 and 0.01.

OTHER OBSERVATIONS:

- The maximum and minimum value of the throttle is 0.2 and -0.2.

- If the module steering angle value is bigger than 10 degrees (0.4) the throttle is set to zero to avoid the car out road.

- The ideal speed value is 15 mph.

REFERENCES

[1] George Gillard, Tutorial *“An introduction and tutorial for PID controllers”*.

[2] Channel “AerospaceControlsLab”, YouTube video Lesson “Controlling Self Driving Cars”.