PID Controller Project

1. Introduction

The objective of this project is to implement a PID controller that will keep a car on track. The PID controller takes into account the so called cross track error (CTE) and steers the car accordingly in order to bring said error to zero (however, this is not an optimization problem). In the next section, I will describe some of the implementation details.

2. Implementation

I have followed the Udacity lecture in order to implement the PID controller. The procedure of estimating the steering angle has three components that depends on the CTE, namely:

- Proportional component : implies that the steering angle (towards the desired trajectory) is directly proportional to the current CTE.
- Integral component: implies that the steering angle is directly proportional to the overall CTE. This component is mainly used to combat systematic bias of a car steering mechanism.
- Differential component : implies that the steering angle is directly proportional to the current derivative of the CTE. The main goal of this component is to prevent overshooting (zig-zagging) and provide smooth motions.

The steering angle is then computed according to the following equation:

$$\alpha = -\tau_p c_p - \tau_i c_i - \tau_d c_d$$

where α is the steering angle, c_p, c_i, and c_d are the proportional, integral and differential components, respectively. The contributions of each of the three components is controlled by the parameters τ.

3. Results

The code meets the specifications as indicated by the simulator and the car stays on track during the whole lap (any number of laps, actually). The most "fuzzy" aspect of this project is how to set the appropriate values for the τ parameters. Since we do not have access to the ground truth data (or at least the car motion model), I cannot run the twiddle algorithm directly (which, if I am not mistaken, belongs to the family of stochastic gradient approaches). Therefore, I have run a few laps and observed the influence of each of the parameters on the car's behaviour.

In this project, I have assume that there is no systematic bias (or that it at least can be neglected) so I have set τ_i to zero. The proportional and differential coefficient are set to 0.1 and 0.75, respectively. The resulting performance is shown in video "output_pid.mp4". The differential component is crucial to prevent zig-zagging (consistent overshooting). In fact, the behaviour of the car for τ_d equal to zero is shown in video "output_p_only.mp4". On the other hand, note that using only the differential component (with the other two switched off) would not have any effect on the steering angle since the temporal derivative of the steering angle would be always zero (there is no active agent forcing the angle to change).

4. Discussion

The PID controller is a simple yet powerful tool for controlling car's steering. The most challenging part is to find an online parameter optimizer so the τ parameters can be optimized without having to perform the entire twiddle procedure.