I have used the race track from behavioral cloning project and implemented the PID controller in C++ to steer the vehicle around the race track.

I have calculated the required cross track error to compute the appropriate steering angle.

PID Overview

PID controllers are simple reactive controllers that are widely used. The difference between the measured and the desired value (setpoint) of a process variable of a system is fed into the PID controller as an error signal. Depending on the PID parameters a control output is generated to steer the system closer to the setpoint. In the project, a car simulator produces the error signal as the distance between the actual car position on the road and a reference trajectory, known as cross-track error (CTE). The PID controller is designed to minimize the distance to this reference trajectory. The primary control output of the PID controller here is the steering angle.

P - Proportional Gain

The proportional term computes an output proportional to the cross-track error. A pure P - controller is unstable and at best oscillates about the setpoint. The proportional gain contributes to a control output to the steering angle.

D - Differential Gain

The oscillations caused by purely D control can be mitigated by a term proportional to the derivative of the cross-track error.

I - Integral Gain

A third contribution is given by the integral gain which simply sums up the cross-track error over time. Thereby, biases can be mitigated, for instance if a zero steering angle does not correspond to a straight trajectory. At high speed, this term can also be useful to accumulate a large error signal quickly, for instance, when the car is carried out sideways from the reference trajectory. This allows to reduce proportional gain, which causes oscillations at high speeds. It is also beneficial to limit the memory of this term to avoid overshooting.

Tunning

I have manually tuned the hyperparameters and found the following parameters helped to keep the car in track kp = 1.0, ki = 0.0, kd = 20.0.