

PID Controller Project

In the following I am describing how the PID controller works.

In fact there are two implementations of said controller – one for keeping the car on the track and one for controlling the speed of the car.

The general implementation of the PID controller follows the standard pattern.

There is a proportional part which gets weighted by K_p and proportionally processes the Cross Track Error CTE. This causes the car to oscillate around the middle line of the lane. To counter this the derivative part K_d is incorporated. It weights the change of the CTE between two time steps. This causes the car to start adjusting the steering angle while coming closer to the target quicker, hence reducing the oscillation. Finally the integral part K_i makes for the controller to eliminate the bias if there is one caused by the implementation of the car by adding up the CTE and incorporating the cumulative error over time.

Implementing this in the main.cpp causes the car to stay on track very smoothly, but only if there right hyperparameters were chosen.

I tried out multiple parameters manually, be only choosing 1 or 2 parameters at once and leaving the other part at "0". If for example the car was always off a bit, i slightly increased the K_i integral part. Or if it was overshooting to harshly, I increased the K_d part. With this I found good hyperparameters in $K_p = 0.3$, $K_i = 0.00015$, $K_d = 4.0$ for the controller controlling the lane following.

To control the speed I implemented another object of this controller with different hyperparameters which were more fitting for the speed (see code for values). The vector angles is grabbing the steering angles used and then builds the average of the last 15 angles that were input into the cars control. To generate the input for the error function of the PID controller it takes 50 as the max speed in comparision to the current speed. The max speed (50) gets reduced according to the average of the last steering angles and a proportional factor of 3. Hence in curves the car automatically slows down.