PID Controller

Compilation

The Code compiles without errors.

Implementation

The goal of the project is to implement a PID controller in C++ to maneuver the vehicle around the track! I implemented to PID controllers; steering angle PID control and speed PID control. The steering angle controller helps to keep the CTE as low as possible and the speed controller makes the car speed stays in a certain speed level (here I defined the desired speed 30.)

The PID implemented in PID.cpp. In PID::UpdateError, I defined the proportional, integral and differential errors and in PID::TotalError , I calculated the total amount of error using the tuned coefficents.

Reflection

Describe the effect each of the P, I, D components had in your implementation.

P: Propotional

The proportional part tries to vehicle stays in the center of the lines. Increasing the proportional coefficent (Kp) makes the car keeps on the track, but also makes more error in steering angles.

I: Integral

The integral part makes to reduce the steady state error system. It can be calculated by the summation of cross track errors.

D: Derivative

The differential part makes a better damping and help to avoid overshooting.

Describe how the final hyperparameters were chosen.

The final hyperparameters (P, I, D coefficients) was chosen through manual tuning, trial and error. First I initilized the coefficient values to zero. Then I increased the Kp until I got the a steady response. Then add the Kd to help to avoid overshooting and I did this steps again and again until reached a suitable values. The Ki parameters stayed close to zero. So the final parameters for steering angle and speed controllers to get the best fit are:

- Steering angle PID controller: Kp= 0.15, Kd= 1.5, Ki= 0.0001
- Speed PID controller: Kp= 0.1, Kd= 0.0001, Ki= 0.0002