

CarND-Controls-PID

Writeup

In this project, one PID controller was used to control steering, and the other PID controller was used to control throttle. The car is able to drive successfully around the track at the speed of 40 mph.

The PID controllers were implemented in PID.cpp. In the PID class, the method Init() initializes a PID controller. The UpdateError() method keeps track of and update the proportional, integral, and differential errors. The TotalError() method returns the combination of the proportional, integral, and differential corrections.

In the main.cpp, I created 2 instances of the PID class—pid and pid_throttle. The pid was used to control steering. The pid_throttle was used to control throttle, or the speed. The 2 classes were initialized with the set parameters. Both PID controllers were fed when new information came from the simulator. The cross track error was passed to the pid controller, in order to update the error of pid and get the steering value. The speed difference between the current speed and 40 mph was passed to the pid_throttle controller, in order to get the throttle value.

I tuned the parameters manually multiple times to see whether the car can drive successfully. The parameters are used for the pid controller are 0.2 for Kp, 1.5 for Kd, and 0.0001 for Ki. The parameters are used for the pid_throttle controller are 0.2 for Kp, 0.2 for Kd, and 0.0002 for Ki.