Reflections

Impact of PID components of controller

a) P – P stands for Proportional. Here the control value is directly proportional the error term. The error in our case is the deviation of car from middle of the road and control variable is the steering angle. So only with this term, we expect the car to steer back to the centre of the road.

Pout = Kp . e(t)

The control variable value is at time t is directly proportional to the error term at time t. And the constant term Kp controls the magnitude of response. If we make Kp very high, then the correction will keep overshooting the mean desired state (i.e. e(t) being close to zero). If Kp is very low, then we get an unresponsive filter.

In my experiment, I manually tweaked the values, I first started with Kp tuning keeping Ki and Kd as zero. I kept the throttle constant as 0.3.

Kp values of 0.01 to 0.03 gave a smooth tracking on straight sections of the road but the car did not respond well to sharp turns. .To have a more responsive control, I increased the Kp value to about 0.9 and 1. This made the car responsive to curve but came with wild fluctuations.

I then decided to freeze Kp to middle of the above range i.e. around 0.07 and then move to tuning Kd

Video in file **Kp_only.mp4** shows the behavior of Kp = 0.07 with Ki and Kd = 0. You can see that while it is responsive and car drives nicely on the initial straight road, there is still a significant overshoot and car veers off the road even on slight bends. This need to be dampened.

b) D – D stands for derivative. Control value is proportional to the rate of change of error.

Pout = Kd . de(t)/dt

Here the control value at time t is proportional to the derivative of error term. And the constant tem Kd controls the impact of derivative term. Keeping everything else constant, increasing Kd leads to a dampening of oscillations.

After fixing Kp, in (a) above, I started tuning Kd to keep the car responsive yet control the overshoots. I started off with high values of Kd i.e. about 4. It immediately ensured that car stayed on the track. In other words a high value of Kd brought down the magnitude of overshoots, however, it created high frequency oscillations with car wavering around the middle of the road. To smoothen these, I started reducing Kd from 4 towards 0 to find the sweet spot balance between control of overshoot impact of Kd versus high frequency oscillations due Kd magnitude.

With experimentation and manual observation, I found Kd=1.1 to go well with Kp of 0.07. Sample video is shown in **Kp_kd_only.mp4**

c) I – I stands for Integral. Here correction term is proportional to the cumulative error from beginning. It provides a nice smoothening of long term bias. Since the error term here is cumulative, the value of Ki needs to be kept small.

With experimentation I used a small value of 0.0008 for Ki.

A final video of tuned PID controlled Car is contained in file **pid_video.mp4**.

PID parameter tuning made me realise the fine balance required to tune PID parameters. As extension I plan to use twiddle or some similar algorithm to tune the parameters and compare those with manually tuned values.