PID-Controller Project Writeup

Effect of P, I and D components

The P component is the main component to reduce the error, i.e. heading the car into the right direction. If K_P is set too low, the car is not able to stay on track (especially in curves). If K_P is set too high, it overshoots and circles around the desired track and potentially even leaves the track if it overshoots too much.

The D component solves the "circle around" issue of the P component, choosing K_D is thus strongly dependent on the choice of K_P.

The I component had a rather limited influence in my implementation, with the final parameters it is even possible to set the K_I parameter(s) to 0 (only using a PD controller) while we are still able to drive around the track successfully.

Tuning of parameters

Manual tuning has been done to get the final parameters. In the end, I used a PID controller for the steering and the throttle to control the speed (using $-(v_{max} - CTE_{steering} - v_{current})$ as CTE – i.e. slowing down if the difference to the desired track gets higher).

I started with a P-Controller (setting $K_I = K_D = 0$) trying to find a value where we get reasonable values for the steering and are able to drive safely as far as possible. K_D has then been chosen to reduce the overshooting, but still keeping it low enough to not remove the influence of the P component (i.e. reduce errors fast enough). I've also tried some different values for K_I but there was no big difference visible (as there is no bias in the simulation).

The parameters for the Speed PID controller have then been chosen similar to those for the Steering PID controller, except that K_D has been chosen lower as "circling around" in speed is not that bad but we rather want be able to react fast (slow down) if the error gets too high – which will make us able to drive faster in average.

The pid_recording.mp4 video shows the car driving two laps with the final implementation.