Vehicle Control Project Final Report

Group 9 Changqing Lu, Eric Martin Sosa-Lesso, Gabe Alexander Perez, Yichen Zhou

Speed Control

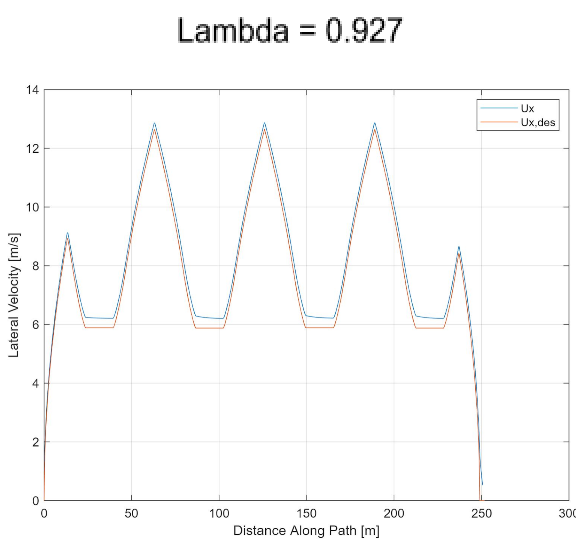
$$F_x = ma_{des} + F_{rr} + F_d + m\lambda(U_{x,des} - U_x)$$

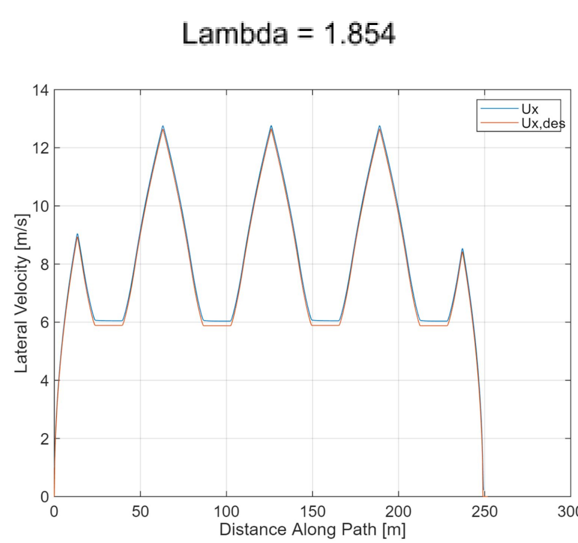
The equation above formed the basis of our speed controller. Our goal was to choose a gain, lambda, that would be successful in closely tracking our desired speed profile.

$$\lambda = \frac{F_x - ma_{x,des} - F_{rr} - F_d}{m(U_{x,des} - U_x)}$$

We started by attempting to choose a gain that was physically reasonable. From the given speed profile, ranging from 0 to 13m/s, we concluded that 6m/s would be a reasonable speed to use for our calculations. Furthermore, we inspected our desired desired longitudinal acceleration profile and noticed the car appeared to have a negative longitudinal acceleration for the majority of the duration of the simulation. Thus we chose to use a small negative desired longitudinal acceleration for our calculation. Additionally we pulled from the assumptions made in Assignment 4 to choose values for Fx (0.1m*g N) and (Ux,des - Ux) (1m/s). With these assumptions, we calculated that lambda to be 0.927.

Given that our vehicle might encounter small grade disturbances in the real world, we tested our speed controller on a simulated straight road with a constant grade of 3%. The speed controller tracked the speed profile closely. We noticed that a more realistic assumption for longitudinal speed error was 0.5 m/s, so we recalculated the gain with this new assumption to get a final lambda of 1.854.





Steering Control

Lookahead Controller

Our first step in developing our lookahead controller was thinking of a physically reasonable gain. We used the following simplified model to calculate this gain:

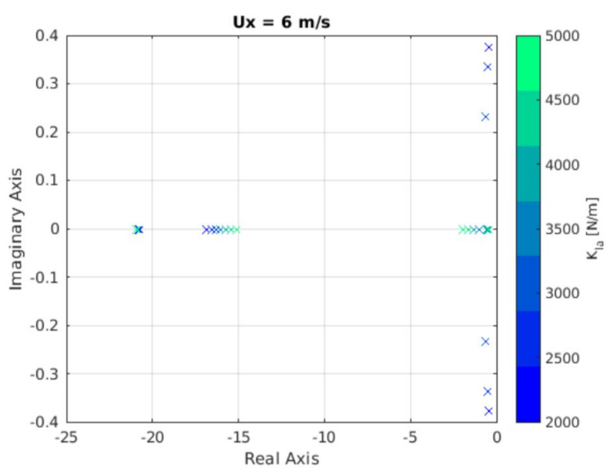
$$\delta = \frac{-K_{la}\left(e + x_{la}\Delta\Psi\right)}{C_{af}}$$

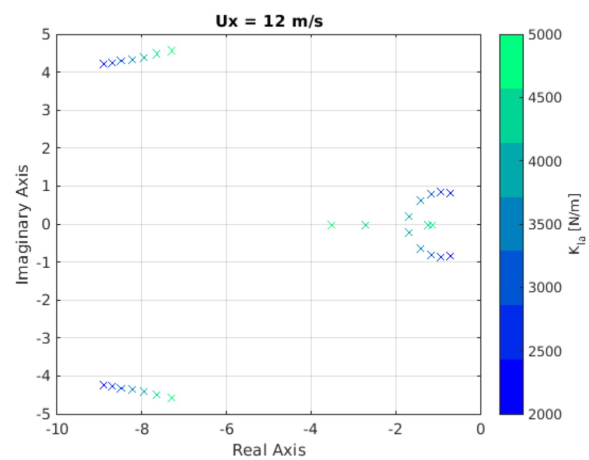
We wanted our controller to produce a 3 degree steering angle in response to a lateral error of 1 meter with the vehicle pointed straight along the path and a 15 meter lookahead distance. This gave us a gain in the range of 4000 N/m.

Next, we considered our feedforward steer controller, varying gains to analyze their effect on pole placement, and thus stability of the system.

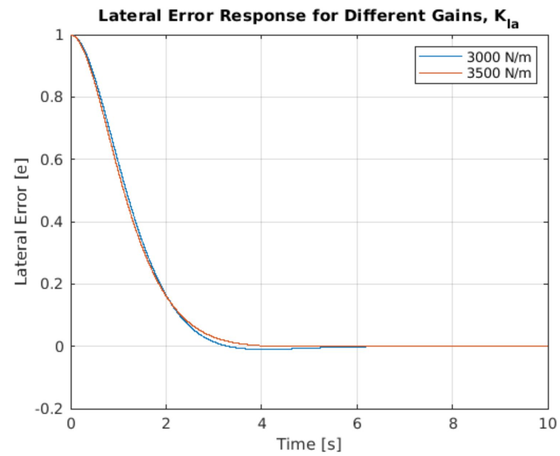
$$\begin{aligned}
\delta &= \frac{-K_{la}}{C_{af}}(e + x_{la} * \Delta\Psi) + \delta_{ff} \\
\delta_{ff} &= \frac{K_{la}x_{la}}{C_{af}}\Delta\Psi_{ss} + \kappa(L + KU_x^2) \\
\Delta\Psi_{ss} &= \kappa\frac{m * a * U_x^2}{(L * C_{ar}) - b}
\end{aligned}$$

We focused on the range of gains between 2000 and 5000, given that these seemed most physically realizable given our earlier calculation with the simplified steering model. Understanding that our desired speed was variable according to our given speed profile, we decided to check the gains for both our minimum longitudinal speed case and maximum longitudinal speed case.





Analyzing our root locus diagrams, we found that changing the longitudinal velocity had a significant effect on the damping ratio of the system. Seeking to understand the error response of the system, we decided to simulate the lateral error response for gains of 3000 N/m and 3500 N/m for Ux = 12m/s. We were able to conclude that even at the largest longitudinal speed of our desired path, the simulated overshoot was not significant. Thus, we chose a Kla of 3500 N/m because it provided the lowest physically reasonable gain with no overshoot. This would give our controller the flexibility to handle any sensor noise or system disturbances the vehicle might encounter in the real world/difficult simulator.



PID Controller

To determine the gains for our PID controller, we first analyze the linear model that is simplified from nonlinear vehicle dynamics by reasonable assumptions. We assume small angles and constant Ux, which gives us a starting point in tuning the gains. Since we add an integral controller for lateral error term, we include a new term into the state vector, which is the integral of lateral error, e. Thus, the state vector becomes

$$\boldsymbol{x} = \begin{bmatrix} \int e \\ e \\ \dot{e} \\ \Delta\Psi \\ \Delta\dot{\Psi} \end{bmatrix}$$

To develop a linearized model, we need to first visit the dynamic systems without any assumptions. The nonlinear system has dynamic relations as follows,

$$\begin{aligned}
m\dot{U}_x &= F_{xr} + F_{xf}\cos\delta - F_{yf}\sin\delta - F_{rr} - \frac{1}{2}\rho C_D A U_x^2 - mg\sin\theta_r \\
m\dot{U}_y &= F_{yf}\cos\delta + F_{yr} + F_{xf}\sin\delta - mrU_x \\
I_z\dot{r} &= aF_{yf}\cos\delta + aF_{xf}\sin\delta - bF_{yr} \\
\dot{s} &= \frac{1}{1-e\kappa}\left(U_x\cos\Delta\Psi - U_y\sin\Delta\Psi\right) \\
\dot{e} &= U_y\cos\Delta\Psi + U_x\sin\Delta\Psi \\
\dot{\Delta\Psi} &= r - \kappa\dot{s}
\end{aligned}$$

To develop the simplified model, we assume constant velocity U_x and assume that $U_x \approx \dot{s}$. And since Niki is a front-drive vehicle, we assume that $F_{xf} = F_x,\ F_{xr} = 0.$ And for the longitudinal force, since the velocity is constant, we have,

$$F_x = F_{xr} + F_{xf} = F_{yf}\delta + F_{rr} + \frac{1}{2}\rho C_D A U_x^2 + mg\sin\theta_r$$

Assume linear tire model and assume that $r \approx \Delta\dot{\Psi} + \kappa U_x,\ U_y = \dot{e} + U_x\Delta\Psi$, we can derive the state vector derivatives.

$$\begin{aligned}
m\ddot{e} &= m\left(\dot{U}_y + U_x\dot{\Delta\Psi}\right)\\
&= -\frac{C_{\alpha f}+C_{\alpha r}}{U_x}\dot{e} + (C_{\alpha f}+C_{\alpha r})\Delta\Psi - \frac{aC_{\alpha f}-bC_{\alpha r}}{U_x}\dot{\Delta\Psi} - (aC_{\alpha f}-bC_{\alpha r})\kappa - mU_x^2\kappa + C_{\alpha f}\delta\\
I_z\ddot{\Delta\Psi} &= I_z\dot{r} - I_z\dot{\kappa}U_x\\
&= \frac{bC_{\alpha r}-aC_{\alpha f}}{U_x}\dot{e} + (aC_{\alpha f}-bC_{\alpha r})\Delta\Psi - \frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{U_x}\dot{\Delta\Psi} + \dots\\
&\quad \dots + (aC_{\alpha f}+aF_{xf})\delta - (a^2C_{\alpha f}+b^2C_{\alpha r})\kappa - I_z\dot{\kappa}U_x
\end{aligned}$$

For our PID lateral controller, we want to track both the lateral error as well as the heading. So the general control scheme is,

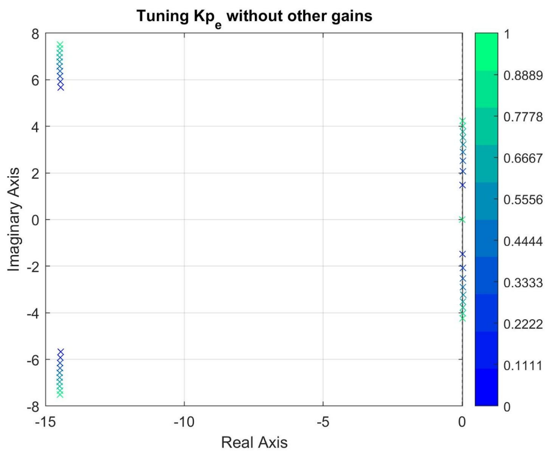
$$\delta = -K_p e - K_d\dot{e} - K_i\int e\mathrm{d}t - K_{ph}\Delta\Psi - K_{dh}\dot{\Delta\Psi}$$

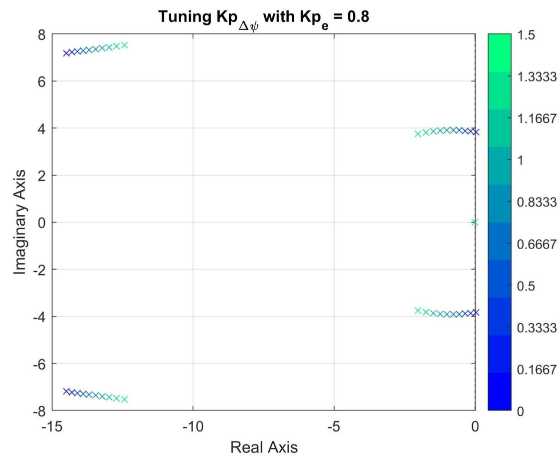
With this controller, we can develop the autonomous system carrying the gains yet to determine.

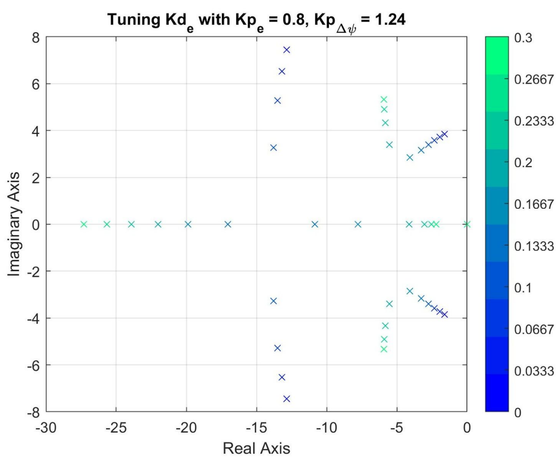
$$\begin{aligned}
\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0\\ 0 & 0 & -\frac{C_{\alpha f}+C_{\alpha r}}{mU_x} & \frac{C_{\alpha f}+C_{\alpha r}}{m} & \frac{aC_{\alpha f}-bC_{\alpha r}}{mU_x}\\ 0 & 0 & 0 & 0 & 1\\ 0 & 0 & \frac{bC_{\alpha r}-aC_{\alpha f}}{I_zU_x} & \frac{aC_{\alpha f}-bC_{\alpha r}}{I_z} & -\frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{I_zU_x}\end{bmatrix}\begin{bmatrix}\int e\\ e\\ \dot{e}\\ \Delta\Psi\\ \dot{\Delta\Psi}\end{bmatrix} + \dots\\
&\dots + \begin{bmatrix}0\\0\\ \frac{C_{\alpha f}}{m}\\ 0\\ \frac{aC_{\alpha f}+aF_{xf}}{I_z}\end{bmatrix}\begin{bmatrix}-K_i & -K_p & -K_d & -K_{ph} & -K_{pd}\end{bmatrix}\begin{bmatrix}\int e\\ e\\ \dot{e}\\ \Delta\Psi\\ \dot{\Delta\Psi}\end{bmatrix} + \begin{bmatrix}0\\0\\ -\frac{aC_{\alpha f}-bC_{\alpha r}}{m}\kappa - U_x^2\kappa\\ 0\\ -\frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{I_z}\kappa - \dot{\kappa}U_x\end{bmatrix}\\
&= \begin{bmatrix} 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0\\ -K_i\frac{C_{\alpha f}}{m} & -K_p\frac{C_{\alpha f}}{m} & -K_d\frac{C_{\alpha f}}{m} - \frac{C_{\alpha f}+C_{\alpha r}}{mU_x} & -K_{ph}\frac{C_{\alpha f}}{m} + \frac{C_{\alpha f}+C_{\alpha r}}{m} & -K_{dh}\frac{C_{\alpha f}}{m} + \frac{aC_{\alpha f}-bC_{\alpha r}}{mU_x}\\ 0 & 0 & 0 & 0 & 1\\ -K_i(\star) & -K_p(\star) & -K_d(\star) + \frac{bC_{\alpha r}-aC_{\alpha f}}{I_zU_x} & -K_{ph}(\star)\frac{aC_{\alpha f}-bC_{\alpha r}}{I_z} & -K_{dh}(\star) - \frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{I_zU_x}\end{bmatrix}\begin{bmatrix}\int e\\ e\\ \dot{e}\\ \Delta\Psi\\ \dot{\Delta\Psi}\end{bmatrix} + [\sim]
\end{aligned}$$

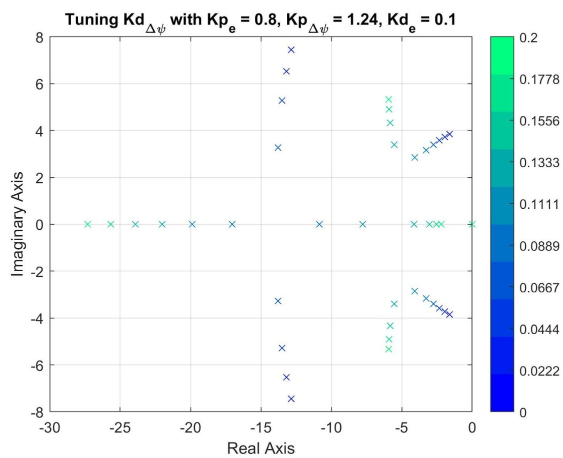
where $(\star) = (aC_{\alpha f} + aF_{xf})/I_z$ and [~] is the term due to the curvature of the road.

Using this model, we can see how poles of this system change with our choice of gains. We start with tuning proportional gains, then turn to derivative gains and finally the integral gain. The values such as Ux and Fxf are obtained from the average value of desired Ux in the provided path information. Some plots we used during tuning is shown below:

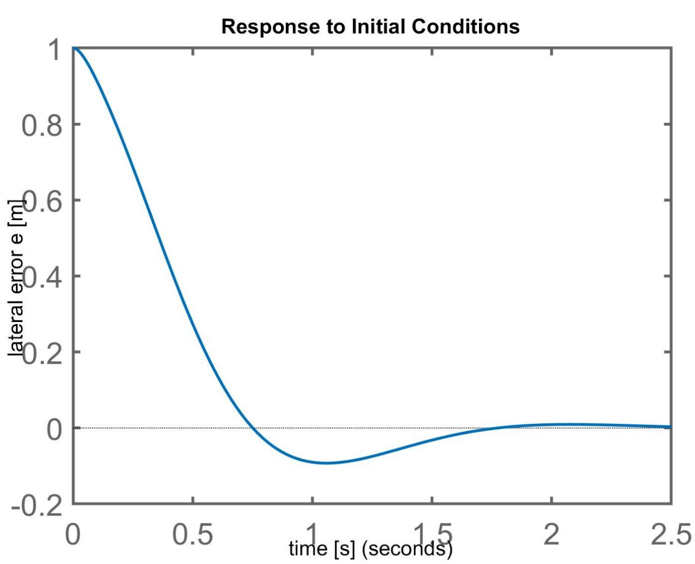








The test of our gain choice is applied with an initial condition e = 1 m. Our preliminary choice for gains are $K_{pe} = 0.8,\ K_{de} = 0.1,\ K_{ie} = 0.001,\ K_{p\Delta\varphi} = 1.24,\ K_{d\Delta\varphi} = 0.08$, the response is shown here:



Thus, our preliminary PID tuning works for the estimated linear system of the vehicle.

After developing the preliminary PID gains for the controller, we discovered that we can tune down the gains a little if we categorize the controller scheme (straight road vs. curved road). Since our objective is to make the steering command less aggressive, it might be helpful to determine whether certain terms of the controller are important.
For the curved road scheme:

(1) Since the vehicle might stay on the right of the road, the lateral error integral terms might be consistently contributing to an increase in the steering angle but essentially it does not help with minimizing the error. Therefore, the integral gain is tuned to zero for the curved region.
(2) When the vehicle enters the curved region, both the rate of the lateral error and the rate of the heading will experience a large increase, which will cause an aggressive change to the steering command, which is not what we want. Therefore, in the curved region, the derivative gains are reduced by a factor.

Therefore, the final PID controller is designed in the following fashion. The gains are selected as, for the straight region,

$$K_{pe} = 0.8,\ K_{de} = 0.1,\ K_{ie} = 0.001,\ K_{p\Delta\varphi} = 1.24,\ K_{d\Delta\varphi} = 0.08$$

while for the curved region,

$$K_{pe,curved} = 0.8,\ K_{de,curved} = 0.05,\ K_{ie,curved} = 0,\ K_{p\Delta\varphi,curved} = 1.24,\ K_{d\Delta\varphi,curved} = 0.04$$

A quick check on the command to steer angle can be applied to see if the steer angle we required is reasonable or not. To track a curved road with zero lateral error, from lectures we know that the steer angle required is

$$\delta_{des} = \kappa(L + KU_x^2)$$

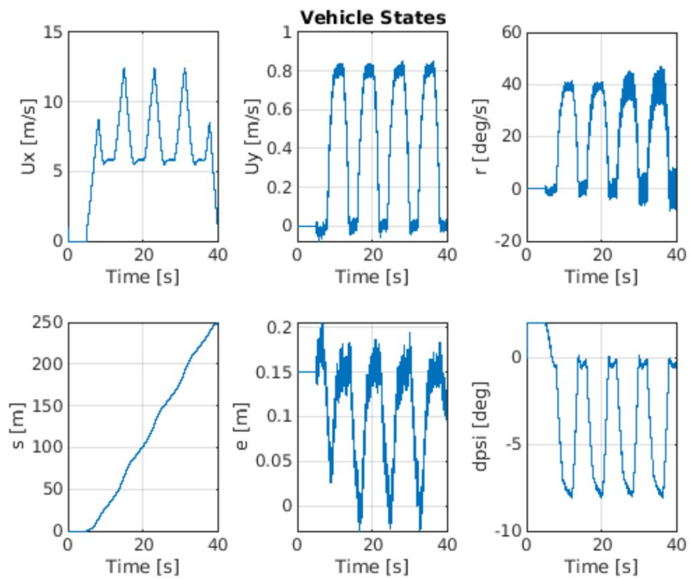
Where K is understeer gradient. Therefore, the changing rate can be estimated by

$$\frac{d\delta_{des}}{dt} = \dot{\kappa}(L + KU_x^2) + 2\kappa K U_x a_x = \frac{d\kappa}{ds}U_x(L + KU_x^2) + 2\kappa K U_x a_x$$

Plugging in desired values in the path information, we can estimate the rate of command should not exceed about 0.2816 rad/s. In our simulation, if dt is set to 0.001 sec, for each step, steer angle change should not exceed 0.016 deg.

Simulation Results

Lookahead Controller (sim_mode = 3)

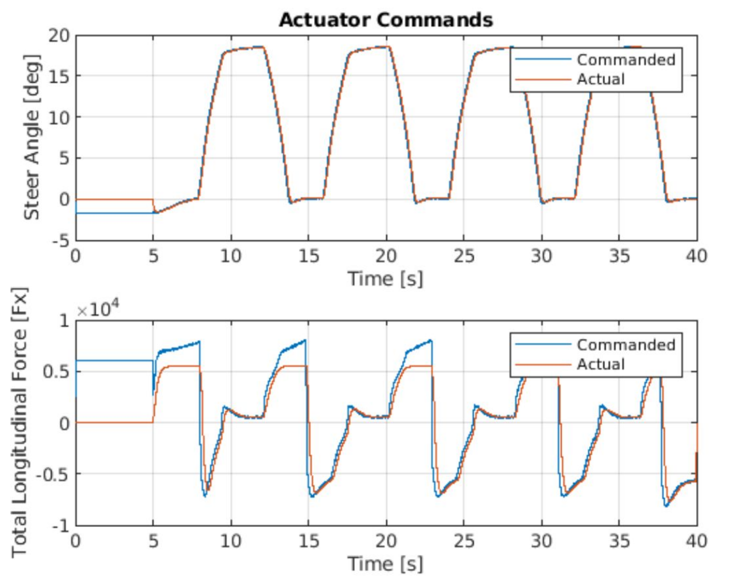


An initial observation of our simulation is that the lateral error seems to only stay to the left of our path. As the vehicle approaches incoming curves, it accounts for the curve by decreasing its longitudinal speed and steering a small amount to the left of the path. By doing so, it achieves our desired goal of staying on the path with a small lateral error (lateral error does not exceed 25cm). With our commanded speed changes, there is some noise propagated into the lateral speed. This lateral speed noise is caused by the sudden changes in longitudinal speed:

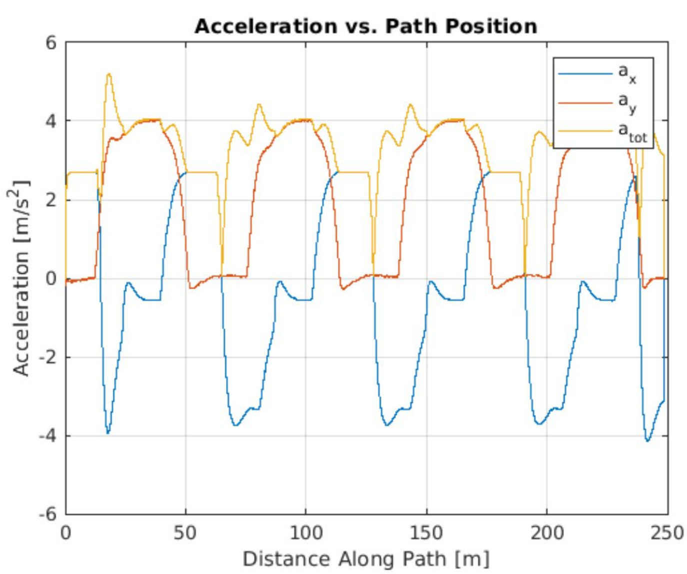
$$\dot{U}y = (F_{y,f}cos(\delta) + F_{y,r} + F_{x,f}sin(\delta))/veh.m - (r \cdot Ux)$$

This noise caused by the longitudinal speed changes further creates even more noise in the lateral error signal since it's based on both longitudinal and lateral speed.

$$\dot{e} = U_x sin(\Delta\Psi) + U_y cos(\Delta\Psi)$$

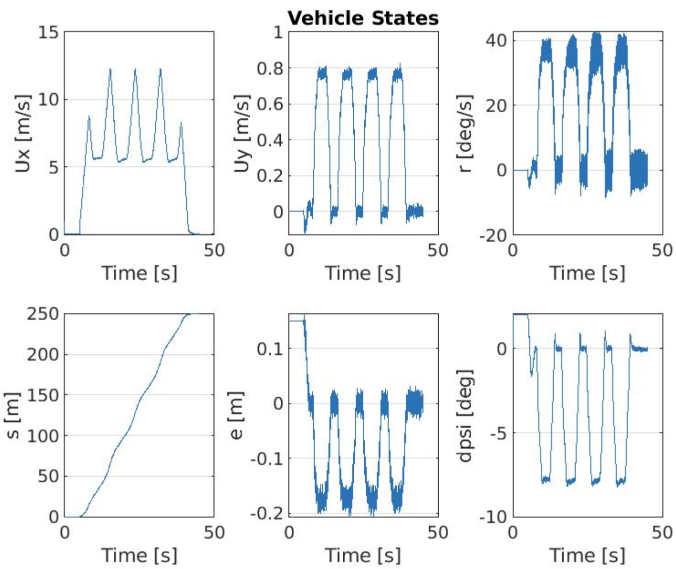


The steering angle and longitudinal force plots of our desired vehicle commands vs. actual results show that the vehicle does a good job of staying with the command signal. However, our actuator model does not account for every part of our vehicle's dynamics which would account for the small amounts of error between the two plots.

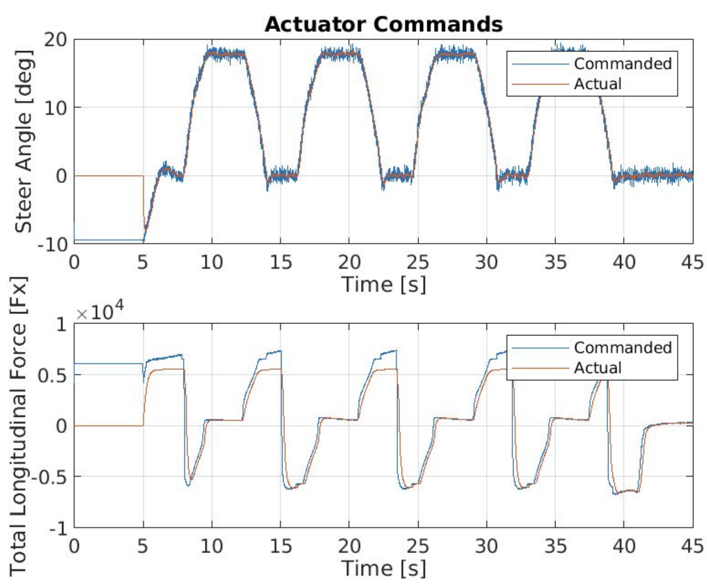


Due to this difference in actuator commands and actual measurements, our acceleration vs. path position plot looks different than the desired results. Our acceleration values seem to peak and show sudden changes in comparison to the more straight and smooth desired plot. These peaks are caused by the sudden changes in speed our vehicle produced to stay on the desired path.

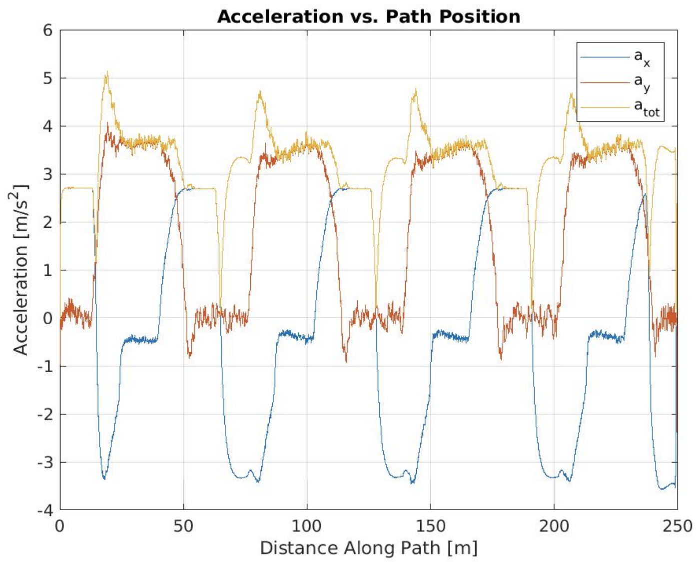
PID Controller (sim_mode = 3)



Noticeable from the lateral error vs. time plot that when turning into the curved road, the vehicle itself tends to stay at the right of the desired path. The lateral error satisfies the requirement.



The steer angle command follows a continuous and smooth curve so that we can conclude that our scheme-based gains choices do not interrupt the desired continuous steering command. The maximum steer angle is 18 degrees, and the maximum rate of change is around 14 degree/sec, which is lower than the maximum steer angle rate of change calculated in the previous section (16 degrees/sec).

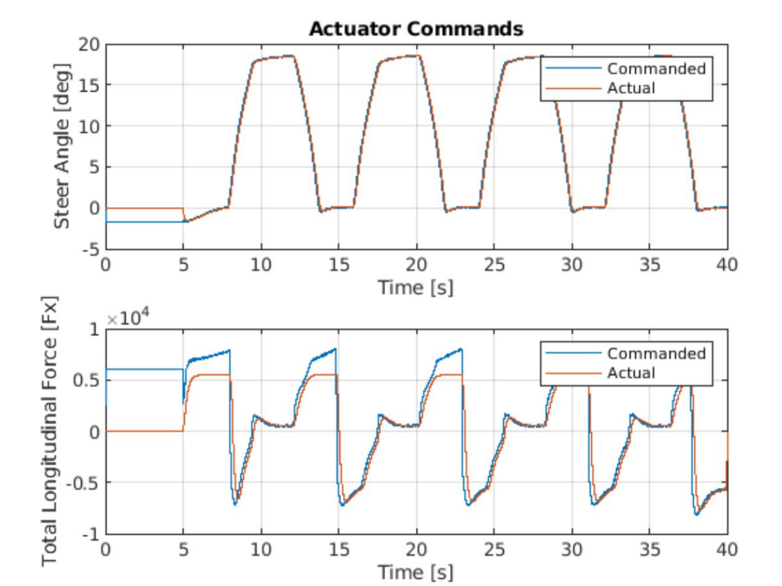


From the acceleration vs time plot, the lateral and longitudinal acceleration both satisfy the design requirement.

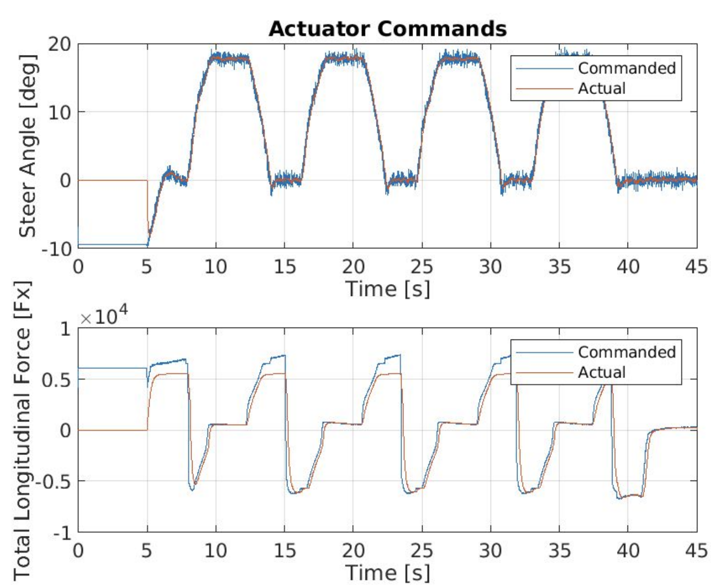
Comparison

Actuator Plots:

Lookahead Controller



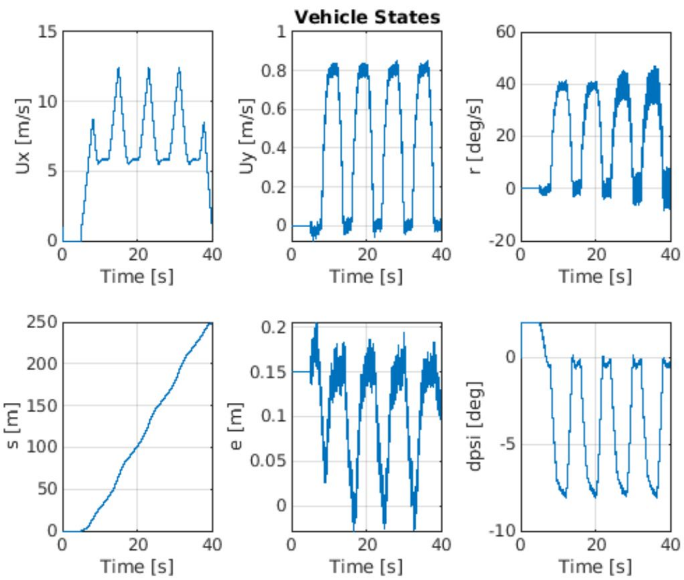
PID Controller



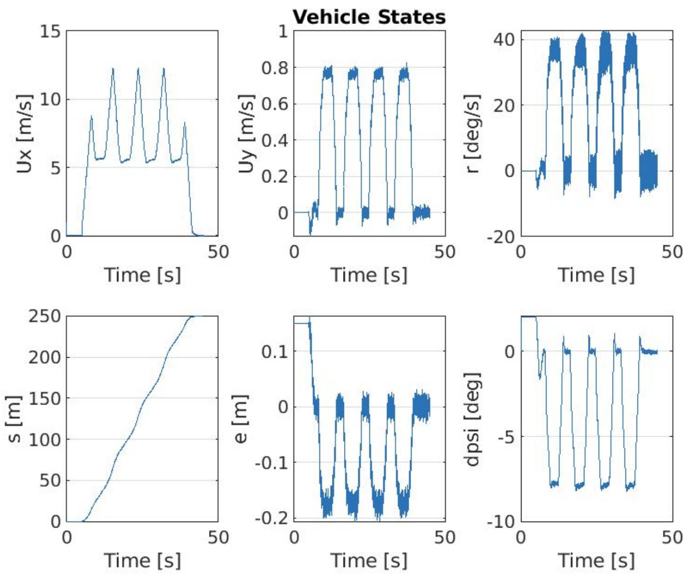
In terms of actuator commands, the vehicle is able to closely track the input from the steer angle controller. The PID controller, however, is producing a high frequency input for the vehicle’s steerangle. Thus, it seems that the noise has more impacts on the PID controller. This can be caused by the integral term used in the controller that amplifies the noise. Thus it is important to take noise into account when designing the controller for real systems.

<u>Vehicle States:</u>

Lookahead Controller



PID Controller



The PID controller shows a more controlled lateral error that stays on the right side of our path. While the Lookahead controller has more variance and noise while it stays to the left of our desired path. The PID controller’s lateral error stays to the right of the path because it acts as the car is on the path while the lookahead controller corrects the car’s steering and speed before reaching the path.

<u>Lesson Learned in our Control Design</u>

To design a controller from scratch, especially for a complex nonlinear model, we can start from a simplified linear model and apply linear control law to that model to come up with a starting point of our design. By dividing and conquering the task, we can build the required controller more easily.

Moreover, we should take real-world constraints into account when designing the controller. For instance, the noise from sensors, reasonable steer angles and rate of changing the steer angle. Those constraints can be estimated by applying dynamic equations of the model with reasonable assumptions. After considering these factors, the controller we designed then becomes useful in the real world.

Readiness to Go on the Car

As we discussed above, our controllers can satisfy the lateral error constraints in all four simulations. Moreover, we estimate the limit on the rate of changing of steer angle by using the provided path information. Our command, though very oscillating in the case considering sensor noise, satisfies the limit we estimated (16 deg/s). Therefore, we think our controller is ready to implement on the real car.