ME227 Project Report

Ilies Ghanzouri, Vince Giralo, Joe Lorenzetti, Yu Wang

May 21, 2020

1 Introduction

This project aims to design a vehicle controller to track a path by controlling the speed and steering. The desired path consists of two laps around a parking lot, forming an oval, where the straight and curved portions are joined by clothoids. The speed controller uses a feedback/feedforward approach, whereas two separate steering controllers were designed. The first approach uses a lookahead method, while the second uses a linear-quadratic regulator (LQR) formulation. These controllers were tested against simulations of varying fidelity.

In the following design of longitudinal and lateral controller, and the choice of controller parameters , we aim at satisfying the following (joint) constraints:

- The longitudinal acceleration, a_{xdes}, should not be larger than 3 m/s^2 or less than -4 m/s^2
- The lateral acceleration, a_y, (calculated from V_{des} and κ) should not exceed 4 m/s^2
- The combined acceleration magnitude should not exceed 4 m/s^2

2 Speed Control

The speed of the vehicle is controlled through the drive force F_x.To achieve desired level of acceleration, we solve for the net force required by accounting for drag, rolling resistance and grade.

$$F_x = m\dot{U}_{des} + F_{rr} + F_d + F_g = m\dot{U}_{des} + F_{rr} + F_d + mg\sin(\theta_r) \tag{1}$$

To design a speed controller, we assume we have a desired speed profile V_{des} and its acceleration a_{des} such that $\dot{V}_{des} = a_{des}$. Let's also suppose that we want the system to have first order dynamics with time constant λ so

$$\frac{d}{dt}(V_{des} - U_x) = -\lambda(V_{des} - U_x) \tag{2}$$

This means

$$\begin{aligned} a_{des} - \frac{1}{m}(F_x - F_{rr} - F_d - F_g) &= -\lambda(V_{des} - U_x) \\ \implies F_x &= \underbrace{ma_{des} + F_{rr} + F_d + F_g}_{\text{Feedforward}} + \underbrace{K_{long}(V_{des} - U_x)}_{\text{Feedback}} \end{aligned} \tag{3}$$

We update the basic longitudinal control that we developed by including the feedforward/feedback speed controller above. The gain selection for this controller is presented in the next section, in parallel with the lateral controller.

3 Steering Control

The lateral dynamics of the vehicle are controlled by the steering angle of the wheels. Two separate controllers are presented.

3.1 Controller A - Lookahead

A lookahead lateral controller was implemented, given by

$$\begin{aligned} \delta &= -\frac{K_{la}}{C_{\alpha f}}(e + x_{la}\Delta\psi) + \delta_{ff} \\ \delta_{ff} &= \frac{K_{la}x_{la}}{C_{\alpha f}}\Delta\psi_{ss} + \kappa(L + KU_x^2) \end{aligned} \tag{4}$$

In the following subsections, we discuss an optimization based gain tuning method by minimizing tracking error.

3.1.1 Gain tuning by directly minimizing tracking error

We explored a method for tuning the gains (K_{la}, K_{long}) and lookahead distance x_{la} by directly minimizing the tracking error. The tracking error is defined to be the mean squared-error of longitudinal speed U_x, longitudinal acceleration a_x, and lateral error e, after normalization to the range $[0, 1]$ to account for their different magnitudes. Specifically, it is desired that the longitudinal speed is close to the prescribed speed profile, longitudinal acceleration is close to the prescribed acceleration profile, and lateral error is close to zero. It is of note that the tracking error L is a function of controller gains and lookahead distance, i.e. K_{la}, K_{long} and x_{la}.

The optimization algorithm is given in algorithm 1. We choose to use coordinate descent with bisection due to some observation from trial and error. Firstly, the gradient evaluation is computationally expensive due to the driving simulation, i.e. a full simulation on the entire path is required for one update to one parameter. Secondly, there seems to be some convexity to the loss function L (e.g. empirically, fixing two parameters and there is typically a direction of modifying the third parameter to minimize tracking loss; there seems to be a unique minimizer for the noise-free simulator). Algorithm 1 is essentially a localization method, it iteratively eliminates half-spaces using gradient information, and hence shrink the size of the set containing the optimal gains. The initial set containing the optimal point is defined by the lower and upper bounds described below.

The upper and lower limits for bisection algorithm are shown in Table 1. In each iteration, the bisection on one coordinate is started from the corresponding limits, and they are modified after bisection converges on this coordinate, i.e. to eliminate a half-space. It is of note that the limits in Table 1 are set to cover wide range of gains, including impractical ones. In practice, fortunately, the algorithm converges to reasonable control parameters.

Parameters	lower	upper
K_{la} [N/m]	0	33500
x_{la} [m]	0	25
K_{long} [N/m/s]	0	10000

Table 1: Parameter limits for bisection

```
Algorithm 1: Coordinate descent using bisection
  Result: Gains $K_{la}, K_{long}$ and lookahead distance $x_{la}$
  initialization $K_{la} = 3500, K_{long} = 1900, x_{la} = 15$ (from HW4) ;
  for iteration $i = 1, \cdots, 9$ do
      for coordinate $j = 1, 2, 3$ (3 dimensions in this case) do
          bisection (until the interval length is shrunk to 1% of the initial length/gap between limits)
            on the loss function to find minimum for coordinate $j$;
          modify limits based on the optimal value from previous step (eliminating half-space);
      end
  end
```

Using simulation mode 0 as simulator, the trajectory of K_{la}, K_{long}, x_{la} and loss over all iterations in the above optimization are shown in Figure 5. The converged values are

$$K_{la} = 12560 \text{ N/m}, x_{la} = 5.86 \text{ m}, K_{long} = 808 \approx 0.04mg \text{ N/m/s}.$$

3.1.2 Performance

Figure 1, Figure 2, Figure 3, Figure 4 show the state and actuator time histories when running this controller with the above gains using the four different simulation modes. As can be seen from these plots, the peak steering angle is about 18 degrees; the peak rate of change in steering angle is about 13 degrees per second, which roughly corresponds to 200 degrees per second on the driving wheel. The lateral error e is generally no more than 15 to 20 cm. We also checked the lateral force constraints and they are well within the physical limits. The acceleration constraints are also satisfied in general, with a few small violations.

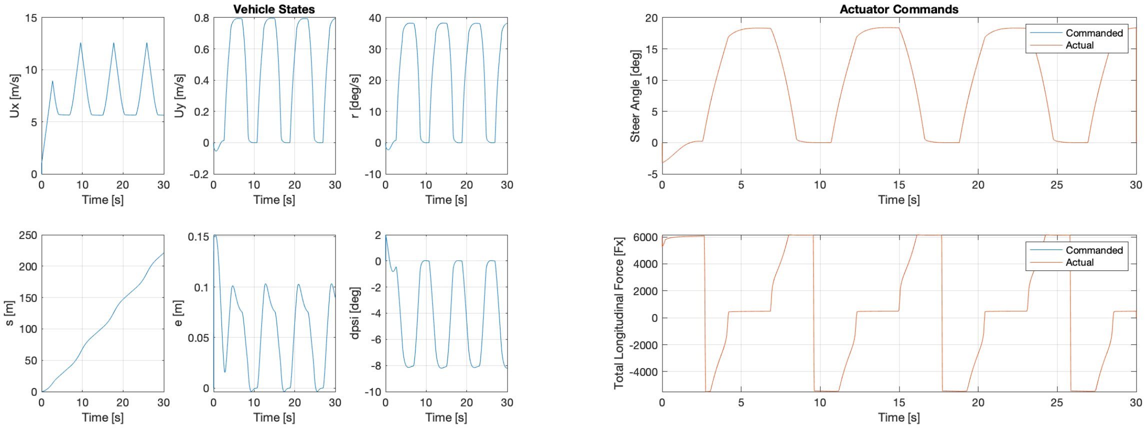

Figure 1: Lookahead controller `sim_mode=0`

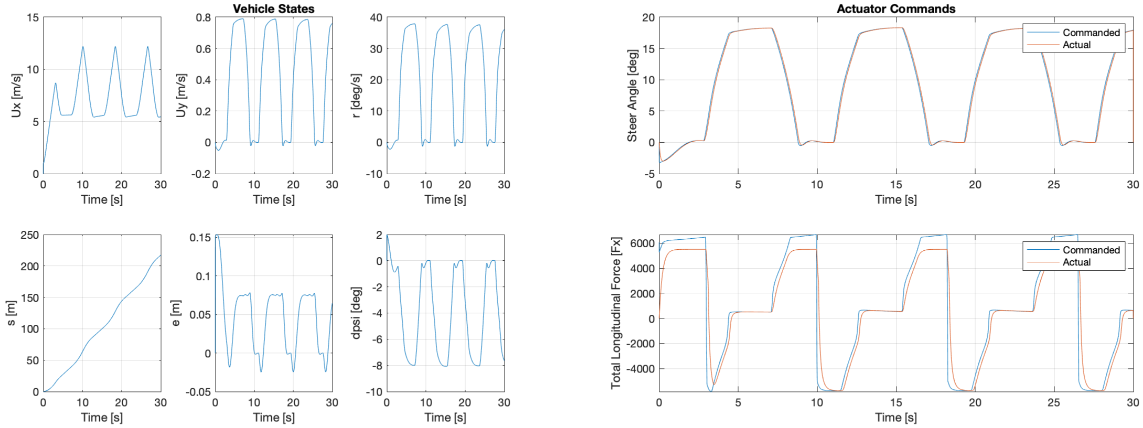

Figure 2: Lookahead controller `sim_mode=1` (harder simulation with actuator dynamics and weight transfer)

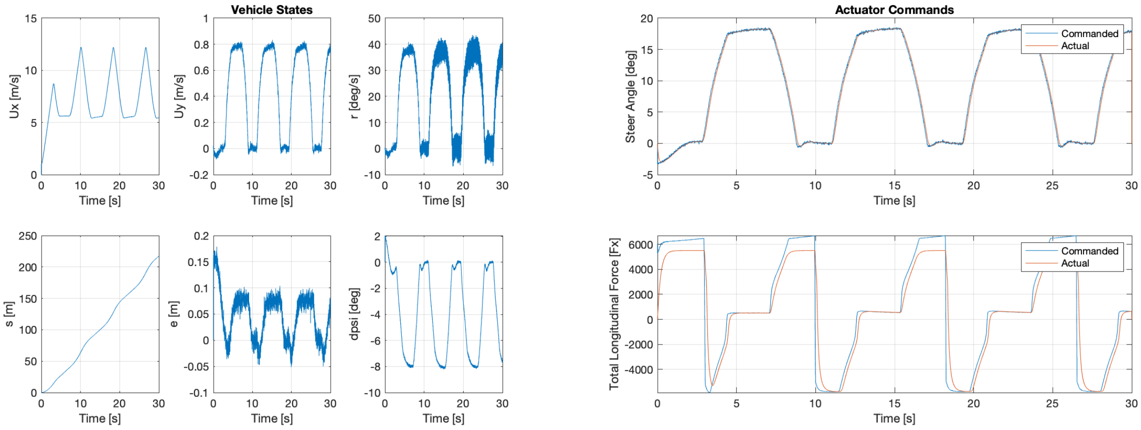

Figure 3: Lookahead controller `sim_mode=2` (add in sensor noise)

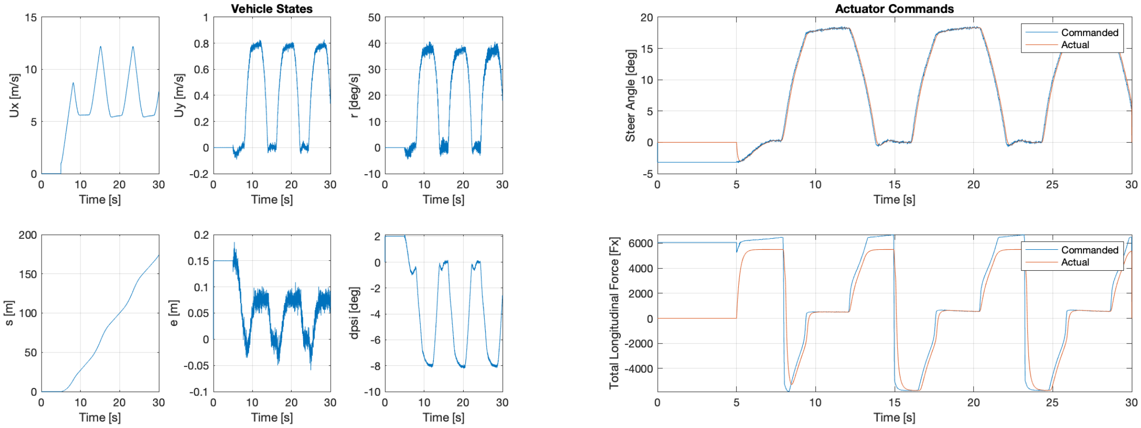

Figure 4: Lookahead controller `sim_mode=3` (hold at the starting line for 5 seconds before going)

3.1.3 Comparison of gain tuning using different simulators

simulation mode	0 (the chosen gains)	1	2
K_{la} [N/m]	12560	15710	29748
x_{la} [m]	5.86	16.6	12.53
K_{long} [N/m/s]	808	4715	253
Runtime [s]	300	310	334

Table 2: Optimal gains obtained from running coordinate descent algorithm 1

In theory, algorithm 1 can be used with any simulator. A natural question is how do the optimal gains from different simulators compare. Table 2 shows the three set of optimal gains from running algorithm 1 with simulation mode 0, 1, and 2. Simulation mode 0 is the case discussed above and the algorithm returns reasonable gain values. Running coordinate descent using simulation mode 1 also returns reasonable control parameters, as confirmed by running simulation with these parameters (not shown here). The optimal control parameters from optimization with simulation mode 2 look more extreme if compared with those from mode 0, and the simulation results (not shown here) using these gains show oscillatory accelerations/forces, indicating possibly extreme parameter values. This is likely due to the stochasticity from the simulation of measurement noise. In other words, localization method (such as bisection used in this section) may be not suitable when there is random noise in the simulator since the objective function itself is stochastic. Given more time, we could instead implement stochastic gradient descent to minimize the expected/average tracking loss. The reported gains for lookahead controller in the previous subsection is from running algorithm 1 with simulation mode 0.

Figure 5, Figure 6, Figure 7 shows the trajectory of gains and loss using three different simulation modes: mode 0, 1, 2. Overall, the algorithm converges within just a few iterations. The run time is around 5 minutes on average.

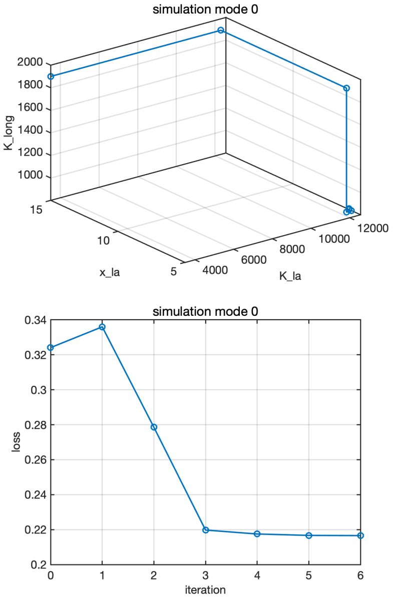

Figure 5: Sim mode 0

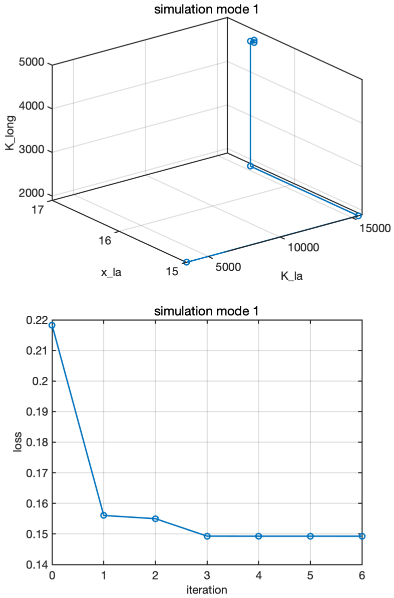

Figure 6: Sim mode 1

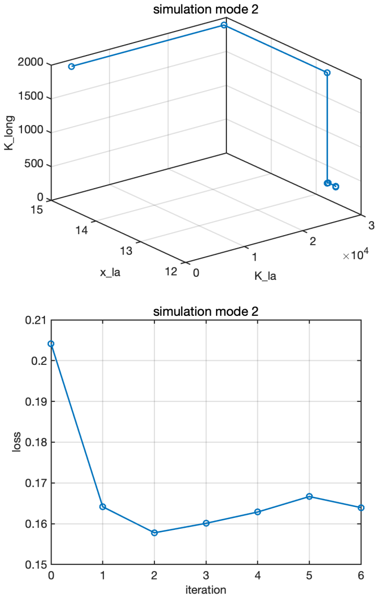

Figure 7: Sim mode 2

3.2 Controller B

The second method for controlling the lateral dynamics of the vehicle uses the principles of linear-quadratic regulators (LQR) to solve for the optimal controller. Dynamics equations are re-derived with relaxed dynamics. For this section, the longitudinal controller is decoupled, with the gain being the same as was found above.

3.2.1 Nonlinear Bicycle Model with Linear Tires

We begin with the nonlinear bicycle model, assuming a linear tire models.

$$\begin{aligned}
F_{yf} &\approx -C_{\alpha f}\frac{U_y + ar}{U_x} + C_{\alpha f}\delta, \\
F_{yr} &\approx -C_{\alpha r}\frac{U_y - br}{U_x}, \\
F_{rr} &= f_{rr}mg, \\
F_d &= \frac{1}{2}\rho C_d A U_x^2, \\
F_g &= mg\sin\theta_r, \\
\dot{U}_x &= \frac{1}{m}(F_{xr} + F_{xf}\cos\delta - F_{yf}\sin\delta - F_{rr} - F_d - F_g) + rU_y, \\
\dot{U}_y &= \frac{1}{m}(F_{yf}\cos\delta + F_{yr} + F_{xf}\sin\delta) - rU_x, \\
\dot{r} &= \frac{1}{I_z}(aF_{yf}\cos\delta + aF_{xf}\sin\delta - bF_{yr}), \\
\dot{s} &= \frac{1}{1 - e\kappa}(U_x\cos\Delta\psi - U_y\sin\Delta\psi), \\
\dot{e} &= U_y\cos\Delta\psi + U_x\sin\Delta\psi, \\
\ddot{e} &= \dot{U}_y\cos\Delta\psi + \dot{U}_x\sin\Delta\psi - U_y\sin\Delta\psi\Delta\dot{\psi} + U_x\cos\Delta\psi\Delta\dot{\psi}, \\
\Delta\dot{\psi} &= r - \kappa\dot{s}, \\
\Delta\ddot{\psi} &= \dot{r} - \dot{\kappa}\dot{s} - \kappa\ddot{s},
\end{aligned} \tag{5}$$

where F_{yf} is the front tire lateral force and F_{yr} is the rear tire lateral force, $C_{\alpha,\cdot}$ is the linear tire coefficient, U_x and U_y are the longitudinal and lateral velocities with respect to the vehicle body frame, a and b are the distances to the front and rear axles from the center of mass, δ is the steer angle, m is the vehicle mass and g the acceleration due to gravity, $f_r r$ is a rolling resistance coefficient, ρ is air density, $C_d A$ is a known aerodynamic parameter, F_{xf} and F_{xr} are longitudinal forces generated by the front and rear tires, r is the yaw rate, s and e are the path distance and lateral error, κ is the path curvature, and $\Delta\psi$ is the yaw angle error with respect to the path.

3.2.2 Assumptions

To simplify this model for controller design we make the following assumptions:

1. Longitudinal controller maintains U_x at the desired speed $U_{x,d}$
2. Small angle approximation for δ and $\Delta\psi$
3. $F_x(s) = m\dot{U}_{x,d} + F_{rr} + F_d + F_g = m\frac{dU_{x,d}(s)}{ds}U_{x,d}(s) + F_{rr} + F_d + F_g$, and $F_{xf} = d_f F_x$
4. $e\kappa(s)$ is negligible
5. $\dot{s} = U_{x,d}(s)$

The first assumption is made such that we can approximately know the U_x quantity such that we do not have to consider the longitudinal dynamics in the controller design. The small angle approximation is also made to simplify some of the trigonometric nonlinearities, and will be a good assumption when the vehicle is close to the path and if the controller is not extremely aggressive. The third assumption provides us with an estimation for the longitudinal force that is being applied through the tires, and is based on the desired velocity profile being tracked perfectly on a straight path where $U_y = 0$. The assumption that $e\kappa(s)$ is negligible simplifies a nonlinearity and is valid when the curvature is small and the path is being tracked well. Finally, since we assume the longitudinal speed is equal to the desired speed, we also assume that this corresponds to the rate that the vehicle traverses the path with respect to s. This assumption is valid under assumption 2 and 4 and when U_y is also small.

3.2.3 Linear Model

Under the previously stated assumptions we simplify the nonlinear bicycle model to obtain the linear model

$$\begin{aligned} F_{yf} &\approx -\frac{C_{\alpha f}}{U_{x,d}(s)}U_y - \frac{aC_{\alpha f}}{U_{x,d}(s)}r + C_{\alpha f}\delta, \\ F_{yr} &\approx -\frac{C_{\alpha r}}{U_{x,d}(s)}U_y + \frac{bC_{\alpha r}}{U_{x,d}(s)}r, \\ \dot{U}_y &= \frac{1}{m}(F_{yf} + F_{yr} + d_f F_x(s)\delta) - rU_{x,d}(s), \\ \dot{r} &= \frac{1}{I_z}(aF_{yf} + ad_f F_x(s)\delta - bF_{yr}), \\ \dot{e} &= U_y + U_{x,d}(s)\Delta\psi, \\ \ddot{e} &= \dot{U}_y + \frac{dU_{x,d}(s)}{ds}U_{x,d}(s)\Delta\psi + U_{x,d}(s)r - U_{x,d}(s)^2\kappa(s), \\ \Delta\dot{\psi} &= r - \kappa(s)U_{x,d}(s), \\ \Delta\ddot{\psi} &= \dot{r} - \frac{d\kappa(s)}{ds}U_{x,d}(s)^2 - \kappa(s)\frac{dU_{x,d}(s)}{ds}U_{x,d}(s). \end{aligned} \tag{6}$$

3.2.4 State Space Representation

The linear model (6) can equivalently be written in state space form as the linear, time-varying model

$$\dot{x} = A(s)x + B(s)u + d(s), \tag{7}$$

where $x = \begin{bmatrix} e & \dot{e} & \Delta\psi & \Delta\dot{\psi} \end{bmatrix}^T$, $u = \begin{bmatrix} \delta \end{bmatrix}$, and

$$\begin{gathered} A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{C_{\alpha f}+C_{\alpha r}}{mU_{x,d}(s)} & \frac{C_{\alpha f}+C_{\alpha r}}{m} + \frac{dU_{x,d}(s)}{ds}U_{x,d}(s) & -\frac{aC_{\alpha f}-bC_{\alpha r}}{mU_{x,d}(s)} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{aC_{\alpha f}-bC_{\alpha r}}{I_z U_{x,d}(s)} & \frac{aC_{\alpha f}-bC_{\alpha r}}{I_z} & -\frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{I_z U_{x,d}(s)} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{C_{\alpha f}+d_f F_x(s)}{m} \\ 0 \\ \frac{aC_{\alpha f}+ad_f F_x(s)}{I_z} \end{bmatrix} \\ d = \begin{bmatrix} 0 \\ -\frac{aC_{\alpha f}-bC_{\alpha r}}{m}\kappa(s) - U_{x,d}(s)^2\kappa(s) \\ 0 \\ -\frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{I_z}\kappa(s) - \frac{d\kappa(s)}{ds}U_{x,d}(s)^2 - \kappa(s)\frac{dU_{x,d}(s)}{ds}U_{x,d}(s) \end{bmatrix} \end{gathered} \tag{8}$$

Note that under an additional assumption that the speed of the vehicle is constant the A and B matrices become time-invariant.

3.2.5 LQR Controller

To control this LTV system we choose to use a time-varying linear quadratic regulator. In particular, we choose to solve the problem using s as the independent variable instead of t, and therefore modify the state space model (7) to by using the fact that under the stated assumptions $\frac{dx}{dt} = \frac{dx}{ds}U_{x,d}(s)$ such that we have

$$\frac{dx}{ds} = A_s(s)x(s) + B_s(s)u(s) + d_s(s), \quad A_s(s) = \frac{1}{U_{x,d}(s)}A(s), \quad B_s(s) = \frac{1}{U_{x,d}(s)}B(s), \quad d_s(s) = \frac{1}{U_{x,d}(s)}d(s). \tag{9}$$

We then formulate the finite-horizon quadratic cost function

$$J(\mathbf{x}_0) = \frac{1}{2}\int_0^{s_f} \left[x(s)^T Q x(s) + u(s)^T R u(s)\right] ds, \tag{10}$$

which can be minimized with the s-varying control law $u(s) = k(s) + K(s)x(s)$ where

$$\begin{aligned} K(s) &= -R^{-1}B(s)^T P(s), \\ k(s) &= -R^{-1}B(s)^T p(s), \end{aligned} \tag{11}$$

and where P and p are solutions to the Riccati differential equations

$$\begin{aligned} -\frac{dP}{ds} &= Q - P(s)B(s)R^{-1}B(s)^T P(s) + P(s)A(s) + A(s)^T P(s), \\ -\frac{dp}{ds} &= -P(s)B(s)R^{-1}B(s)^T p(s) + A(s)^T p(s) + P(s)d(s), \end{aligned} \tag{12}$$

with boundary condition $P(s_f) = 0$ and $p(s_f) = 0$.

3.2.6 Performance

In this section we demonstrate the performance of the proposed LQR controller against a couple of simple variants. In each case we simulate the system with the nonlinear bicycle model and perfect state knowledge, and implement a simple longitudinal controller that is the same in each case. We kept the weighting matrices the constant throughout, namely

$$\begin{aligned} Q &= \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 20 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} \\ R &= 100 \end{aligned} \tag{13}$$

First, in Figure 8 we see the results of the proposed LQR controller under the nonlinear simulation (sim_mode = 0). We can see that the steering angle is reasonable, and the lateral error e seems to stay below 10 cm after recovering from the initial offset. However, while this method computes the gains offline and then gain schedules for real-time implementation, this option was not suitable for this assignment where the entire controller was implemented online.

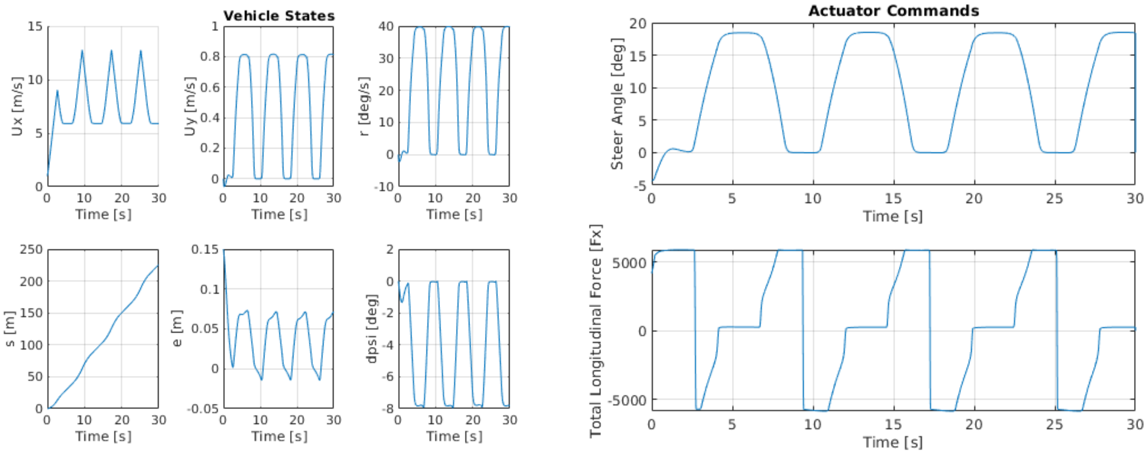

Figure 8: Simulation of nonlinear bicycle model with time-varying LQR controller discussed above.

In Figure 9 we implement the same LQR controller as before, but without the feedforward component (to see its effect on the control). We see that in this case the lateral error increases to be almost twice as large at the peaks. This shows that the feedforward term is quite useful in decreasing the tracking error.

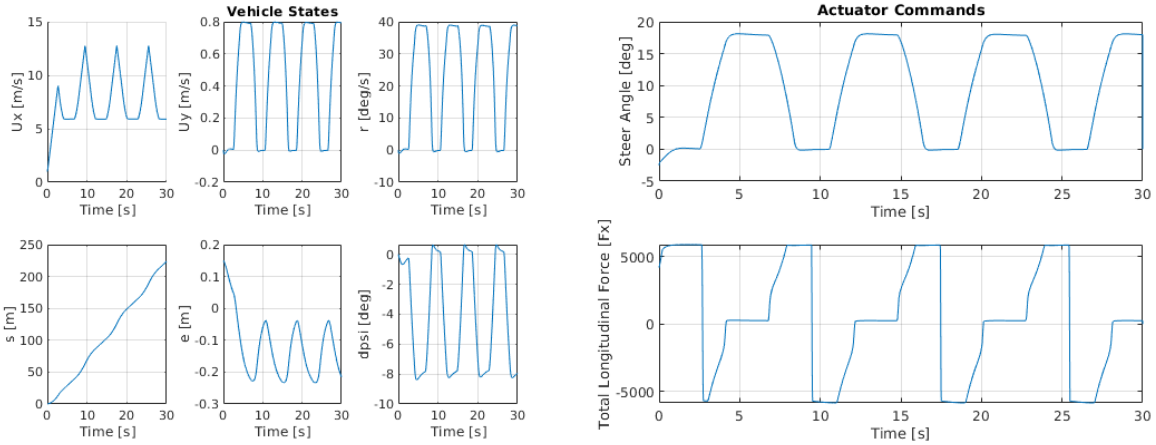

Figure 9: Simulation of nonlinear bicycle model with time-varying LQR controller discussed above, but without the feedforward terms ($k(s) = 0$).

3.2.7 Reformulation

To maximize the performance and minimize the execution time of the controller, a few assumptions were made on the LTV LQR with feed-forward design. In particular we make the assumption that the vehicle moves at a constant velocity of 10 m/s when designing the feedback gains K. In this case the LQR controller synthesis simplifies because the system matrix A is time-invariant. Therefore the infinite-horizon controller is computed once for the system, providing both the gain matrix

$$K = \begin{bmatrix} -0.316 & -0.280 & -1.663 & -0.090 \end{bmatrix} \tag{14}$$

and the solution to the Ricatti equation P. These values are then used to recalculate the feed-forward term at each time step by

$$k(s) = -R^{-1}B(s)^T(PBR^{-1}B^T - A^T)^{+}Pd \tag{15}$$

where the superscript + represents the Moore-Penrose inverse.

Figure 10 shows the results from this controller. In this case we see only a small enlargement in the tracking error. This suggests that the time-varying speed of the vehicle plays a small role in determining the steering angle.

This controller was then run on the high-fidelity simulation (sim_mode = 3). Figure 11 shows the vehicle state and actuator commands for this experiment. The actuator dynamics and weight transfer are seen in the slight variation between the commanded and actual commands. The addition of sensor noise causes noticeable variation the lateral tracking error, but does not reduce performance. There is no loss of performance due to the 5-second delay at the beginning of the simulation.

The accelerations from this test are shown in Figure 12. The total accelerations do not exceed $5m/s^2$, with only small portions of the test exceeding the desired $4m/s^2$.

4 Conclusion

This project developed three different controllers to allow a vehicle to track a given path. A speed controller was developed using feedback/feedforward, and showed that the desired speed and acceleration profile could be well tracked. In addition, two different lateral speed controllers were developed. The first used a lookahead controller, where tuning parameters including gains and the lookahead distance were determined using a coordinate descent with bisection approach. This controller showed good tracking performance, but became susceptible to peaks in acceleration upwards of $5m/s^2$ for the high-fidelity simulation. The second lateral controller used an LQR-based approach. A nominal speed was chosen to reduce the LTV system to an LTI system, allowing for a constant set of feeback gains to be chosen. Due to the affine nature of the dynamic,

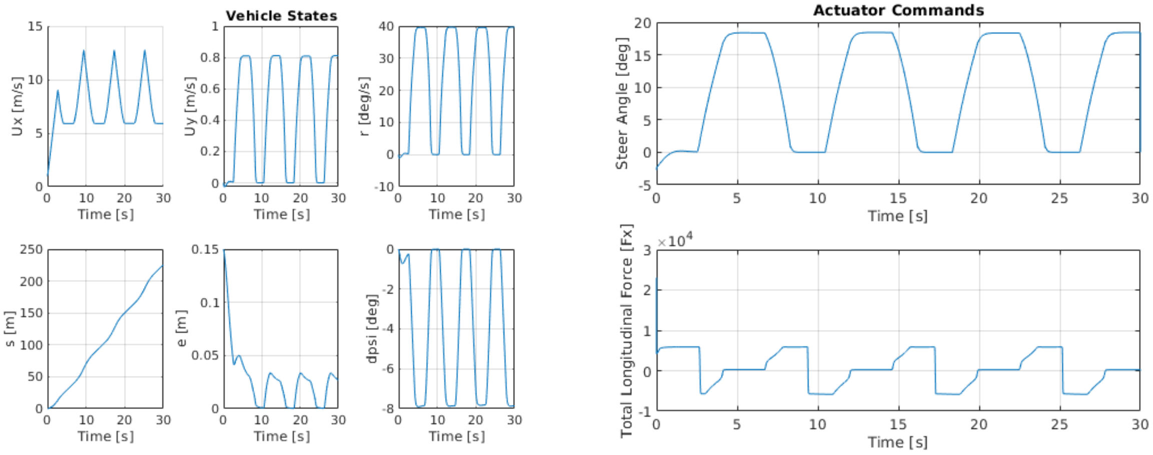

Figure 10: Simulation of nonlinear bicycle model with time-invariant LQR controller based on the state space model assuming a constant longitudinal velocity $U_x = 10$.

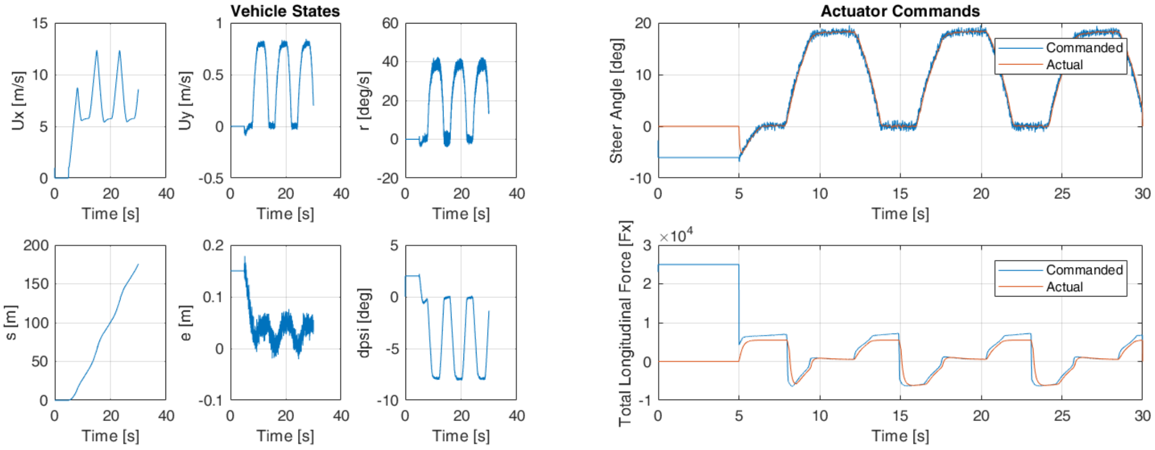

Figure 11: Simulation of nonlinear bicycle model with time-invariant LQR controller in the high-fidelity simulation

a feedforward gain was also required and calculated at each time-step by using the solution to the Ricatti equation from the LTI LQR problem. While the reduction to an LTI system caused an increase in the tracking performance, the reduced computational load was found to be suitable for real-time applications of this controller.

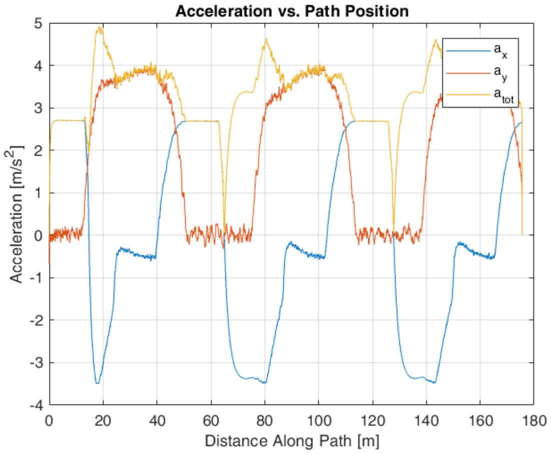

Figure 12: Vehicle accelerations for the time-invariant LQR controller with feed-forward