MEEN 689: Automotive Engineering II

Project 2: All Wheel Steer Race Car

Allen George Philip

Michiel Ashley

Akshay Sarvesh

Executive Summary

Objectives

1. Design an all-wheel steering (AWS) controller to aid a race car in traversing an oval track as fast as possible.
2. Maintain system stability: stay on the track while moving in the proper direction.

Methods

1. Initially the two steering inputs were set equal (with appropriate sign) and a PID controller that drove the lateral error to zero was used. (Open exPID.mo to view this approach)
2. It was assumed that sufficient system identification has been performed and an accurate GPS/IMU has been installed such that the controller can reasonably estimate the vehicle's position and tire slip angles.
3. The system dynamics were formulated such that the state variables are vehicle slip angle, yaw rate, lateral tracking error, and heading error.
4. The 3 system inputs were front steering angle, rear steering angle and front wheel torque.
5. A linear quadratic regulator (LQR) was chosen to regulate the desired system states.

Results

1. The LQR AWS controller drove the race car around the track in 240 sec.
2. The maximum lateral error from the center of the track was 8 m.
    a. after transients decayed (on laps 2 and greater) the maximum error was reduced.
3. After the initial acceleration on the 1st straight away, the speed of the race car was 15 m/s on the straights and 11 m/s on the curves.
4. The initially designed PID controller had similar results, but they are not captured in this report.

Dymola Model

1. To get the same results in this report, run:
    a. File: ex3.mo (exPID.mo for the PID approach)
    b. Model: simpVeh8
    c. If you want to see the animation – our car starts at x=0 and y=-200m

Table of Contents

List of Symbols

Symbols	Explanations	Units
m	Mass of the vehicle	kg
I	Moment of inertia of the vehicle about the yaw axis	$kg\text{-}m^2$
F_y	Lateral force on the tire	N
v_x	Longitudinal velocity of the vehicle center of mass	m/s
v_y	Lateral velocity of the vehicle center of mass	m/s
$\dot{v}_y$	Lateral acceleration of the vehicle center of mass	m/s^2
v_c	Critical velocity	m/s
ω	Yaw rate of the vehicle	rad/s
$\dot{\omega}$	Time derivative of ω	rad/s^2
β	Angle between body x axis and vehicle velocity	rad
$\dot{\beta}$	Time derivative of β	rad/s^2
α	Angle between tire x axis and tire velocity	rad
α_{max}	Maximum angle between tire x axis and tire velocity	rad
φ	Angle between vehicle heading vector and path tangential vector	rad
$\dot{\varphi}$	Time derivative of φ	rad/s
δ_f	Angle between body x axis and front tire x axis (positive cw)	rad
δ_r	Angle between body x axis and rear tire x axis (positive ccw)	rad
l_f	Distance between vehicle center of mass and front axle	m
l_r	Distance between vehicle center of mass and rear axle	m
r	Lateral error of the vehicle from the center of the track	m
$\dot{r}$	Time derivative of r	m/s
C_α	Cornering stiffness	N/rad
$C_{\alpha f}$	Cornering stiffness of the front tire	N/rad
$C_{\alpha f}$	Cornering stiffness of the rear tire	N/rad

Introduction

Dynamics of vehicles are critical to their design, and development, and they can be captured using mathematical models of varying complexities. This is essential to engineers as it allows them to understand the fundamental behavior of a vehicle before actually developing it and performing real time experiments on it. Moreover, simulations based on these models provide a good insight into the control strategies that must be implemented in order to achieve a desired task. There are different approaches to model the dynamics of a vehicle, and therefore, deciding on what approach to choose has to be taken after considering the required level of detail, accuracy, computational cost, availability of resources, and so on. In this paper, a simple bicycle model was used to derive the dynamic equations of an all-wheel steer vehicle, and an LQR controller was used to control the trajectory of vehicle to follow a predefined oval track. The modeling and simulation were performed using the Dymola software.

All Wheel Steer Race Car Dynamics

A bicycle model was used to develop the dynamics of the vehicle. Since the controller was developed for a known race car, a model-based control approach was chosen. The car's dynamics are non-linear, so to use linear control theory the appropriate assumptions must be made. The equation of motion is:

$$m\left(\dot{v}_y + \omega v_x\right) = -\beta(\mathrm{C}_{\alpha\mathrm{f}} + \mathrm{C}_{\alpha\mathrm{r}}) - \frac{\omega}{v_x}\left(\mathrm{C}_{\alpha\mathrm{f}} l_f + \mathrm{C}_{\alpha\mathrm{r}} l_r\right) + \mathrm{C}_{\alpha\mathrm{f}}\delta_f + \mathrm{C}_{\alpha\mathrm{r}}\delta_r \tag{1}$$

$$I\dot{\omega} + \left(\mathrm{C}_{\alpha\mathrm{f}} l_f + \mathrm{C}_{\alpha\mathrm{r}} l_r\right)\beta + \frac{\omega}{v_x}\left(\mathrm{C}_{\alpha\mathrm{f}} l_f^2 + \mathrm{C}_{\alpha\mathrm{r}} l_r^2\right) = \ l_f \mathrm{C}_{\alpha\mathrm{f}}\delta_f - l_r \mathrm{C}_{\alpha\mathrm{r}}\delta_r \tag{2}$$

For a description of the constants, see list of symbols. For a small slip angle, the linearizing assumptions is

$$\beta = \tan^{-1}\left(\frac{v_y}{v_x}\right) \approx \frac{v_y}{v_x} \tag{3}$$

The lateral force on the tire is a nonlinear function of the slip angle, but for small slip angles it is approximately linear

$$F_y = \ C_\alpha \frac{\alpha}{\alpha_{max}\sqrt{1 + \left(\frac{\alpha}{\alpha_{max}}\right)^2}} \approx C_\alpha \alpha \tag{4}$$

In the nonlinear force equation, the terms are normalized by α_{max} to make the non-linear curve more realistic by giving a maximum force for a maximum slip angle, although even this does not quite match the highly complex nature of tire forces.

The linearized EOM is put into a state space form similar to that of [2] on page 301 except with two steering inputs as shown on page 311 in [2]:

$$\begin{bmatrix}\dot{\beta}\\ \dot{\omega}\end{bmatrix} = \begin{bmatrix}\frac{-(C_{\alpha f}+C_{\alpha r})}{mv_x} & \frac{-(C_{\alpha f}l_f - C_{\alpha r}l_r)}{mv_x^2}-1\\ \frac{-(C_{\alpha f}l_f - C_{\alpha r}l_r)}{I} & \frac{-(C_{\alpha f}l_f^2 + C_{\alpha r}l_r^2)}{Iv_x}\end{bmatrix}\begin{bmatrix}\beta\\ \omega\end{bmatrix} + \begin{bmatrix}\frac{C_{\alpha f}}{mv_x} & \frac{C_{\alpha r}}{mv_x}\\ \frac{C_{\alpha f}l_f}{I} & \frac{-C_{\alpha r}l_r}{I}\end{bmatrix}\begin{bmatrix}\delta_f\\ \delta_r\end{bmatrix} \tag{5}$$

$$A = \begin{bmatrix}\frac{-(C_{\alpha f}+C_{\alpha r})}{mv_x} & \frac{-(C_{\alpha f}l_f - C_{\alpha r}l_r)}{mv_x^2}-1\\ \frac{-(C_{\alpha f}l_f - C_{\alpha r}l_r)}{I} & \frac{-(C_{\alpha f}l_f^2 + C_{\alpha r}l_r^2)}{Iv_x}\end{bmatrix} \qquad B = \begin{bmatrix}\frac{C_{\alpha f}}{mv_x} & \frac{C_{\alpha r}}{mv_x}\\ \frac{C_{\alpha f}l_f}{I} & \frac{-C_{\alpha r}l_r}{I}\end{bmatrix} \tag{6}$$

Here v_x is assumed to be constant, although for the race car it won't be. This is a sufficient linearizing assumption since there is significant speed control. Additionally if the speed is going to vary a method of "gain switching" can be used.

Initially in the control design process we thought about defining the velocity as a state variable, however we want access to β, the side-slip, which will help keep the race car stable. This is explained in [1] in the section "Electronic Stability Control" where the author indicates that a side-slip greater than a critical side-slip will lead to oversteer and the car "spinning out". The relationship between side-slip, yaw, and velocity prevents reformulating the system matrices to include velocity as a state variable in the linearized system.

The stability of the system can be checked by finding the eigenvalues of A, by following the same approach as in [2], then the stability can be determined as a function of vehicle speed. The condition for stable forward driving is

$$C_{\alpha r}l_r < C_{\alpha f}l_f \tag{7}$$

With a speed less than the critical speed of

$$v < v_c = \sqrt{\frac{C_{\alpha f}C_{\alpha r}\left(l_f + l_r\right)^2}{m\left(C_{\alpha f}l_f - C_{\alpha r}l_r\right)}} \tag{8}$$

Control Development

The chosen method of control is largely inspired by [3], where the author makes the rear wheel steering angle proportional and smaller than the front steering angle.

The control objective is to make the race car traverse the track as fast as possible – this means that the car must remain on the track and remain stable. The system dynamics are formulated such that these objectives are met by defining two more states. This is based on the method in [1] in the section “Lateral Offset Steering Control”.

More System States

1. $\beta \coloneqq$ side-slip angle
   a. Minimizing the vehicle side-slip will help maintain the stability of the car. On a curve, if the side slip is zero then the direction of travel is tangential to the path - which is what we want.
2. $\omega \coloneqq$ vehicle yaw rate
   a. On a curve, this state should not be driven to zero, instead it should be compared to a desired yaw rate. For a straight line, as in the example in [1], the yaw rate should be driven to zero. This wasn’t realized until later and although the method works, this is limiting the control.
3. $r \coloneqq$ lateral error from center of track
   a. Driving this state to zero will allow the race car to follow the track
4. $\psi \coloneqq$ heading error: angle between vehicle heading vector and path tangential vector

Which gives the full system:

$$\begin{bmatrix} \dot{\beta} \\ \dot{\omega} \\ \dot{r} \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{-(C_{\alpha f} + C_{\alpha r})}{m v_x} & \frac{-(C_{\alpha f} l_f - C_{\alpha r} l_r)}{m v_x^2} - 1 & 0 & 0 \\ \frac{-(C_{\alpha f} l_f - C_{\alpha r} l_r)}{I} & \frac{-(C_{\alpha f} l_f^2 + C_{\alpha r} l_r^2)}{I v_x} & 0 & 0 \\ v_x & 0 & 0 & v_x \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ \omega \\ r \\ \psi \end{bmatrix} + \begin{bmatrix} \frac{C_{\alpha f}}{m v_x} & \frac{C_{\alpha r}}{m v_x} \\ \frac{C_{\alpha f} l_f}{I} & \frac{-C_{\alpha r} l_r}{I} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_f \\ \delta_r \end{bmatrix} \quad (9)$$

Plugging in the nominal values for the system constants the eigenvalues of A are:

$$\lambda_1 = 0 \qquad \lambda_2 = -0.2 \qquad \lambda_3 = 0 \qquad \lambda_4 = -0.2 \quad (10)$$

Which makes sense because a step input to the steering will cause the yaw angle and radial error to grow unbounded whereas the yaw rate and side slip will asymptotically approach a constant value without oscillation. Thus, a stabilizing controller is needed.

Linear Quadratic Regulator

The gains for full state feedback regulator were chosen using the LQR method as discussed in [4].

The Q matrix, which can be informally thought of as the weights associated with the penalty of each system state not being at the desired value is chosen as such:

$$Q = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{11}$$

1. We chose to have the largest weight on the side-slip angle, because we really want the system to remain stable followed by the lateral tracking error since we want the race car to follow the track.
2. The two states with the lowest weight are the heading angle error and yaw rate, with heading error being more important than yaw rate.

Since this is a race car, control inputs are not as limited compared to a passenger car. For example, high fuel expenditure is not a problem in racing compared with passenger cars. However, for this system with steering as input, it is understood that a high cost on control input will decrease the gain and help keep the system stable. So, the chosen R value is chosen larger than the values in the Q matrix at $R = 2000$.

Letting MATLAB compute the gain matrix using the "LQR" call:

$$K = \begin{bmatrix} 1.7430 & 0.7711 & 0.0696 & 2.4872 \\ -0.5308 & -0.5437 & -0.0126 & -0.9398 \end{bmatrix} \tag{12}$$

The control input is then

$$u = K \cdot x \tag{13}$$

Problems in the Control Design

1. The first immediately obvious problem in the control design is the choice to drive the yaw rate to zero. During the curves the desired yaw rate it not zero.
2. In the state space form the system appears linear however, along the curves the calculation of the radial error and heading error utilized trigonometric functions (non-linear)

Modelica Model

The modelica model of the all-wheel steer vehicle at the top level, is illustrated in the below figure. All the parameter values that were used in this model will be provided in the appendix section.

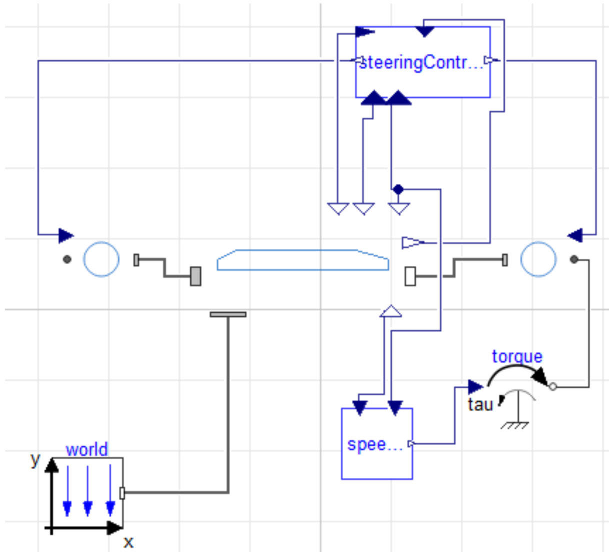

Fig. 1 Modelica Model of the AWS Vehicle

The main components of the vehicle model are the vehicle body, the simple wheel, the steering controller, and the speed controller. The wheel frames are connected to the body frame and has a flange on which the driving torque can be applied. The wheel also takes in steering inputs from the steering controller. The speed controller uses the longitudinal velocity of the vehicle, as well as its absolute x position on the track to output the required torque to the front wheel. The steering controller uses the global x and y positions of the vehicle, the slip angle of the vehicle, as well as the yaw angle of the vehicle to determine the necessary front and rear steering angles. Now we take a closer look at the individual components.

The vehicle body represents the geometry of the vehicle and its center of mass, as well as provide important information about various states that are utilized by the other components. The below figure illustrates the vehicle body model.

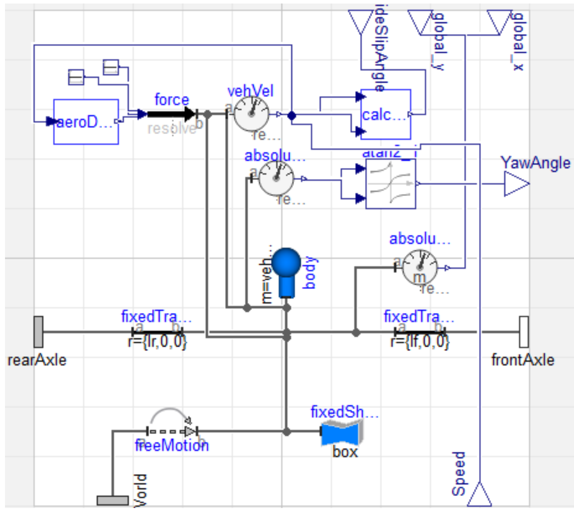

Fig. 2 Modelica Model of the Vehicle Body

The body mass is connected to two translations that represents the distance between the center of mass of the vehicle to the front axle, and the rear axle. The frame of the body mass is also extended out so that it may be connected the world frame outside through a free translation. Moreover, an aero dynamic drag is modeled based on the longitudinal velocity of the center of mass of the vehicle. Using different sensors and calculations, as shown in Fig. 2, the absolute x, and y positions of the center of mass, the yaw, and slip angles, and the longitudinal speed of the vehicle are published.

The simple wheel model is crucial to simulate the interactions between the tires of the vehicle, and the road. The modelica model of the simple wheel is shown below.

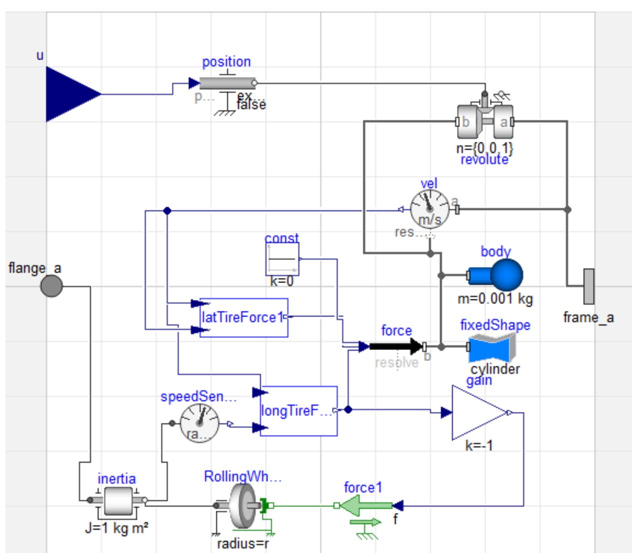

Fig. 3 Modelica Model of the Simple Wheel

The simple wheel model receives the steering angle inputs from the steering controller outside, and steers the wheel accordingly using the position model that is connected to the flange of the revolute joint. In addition, it uses the velocity at the axle resolved in the wheel frame to calculate the lateral force from the road on the wheel, and also, the longitudinal velocity of the axle and the rate of rotation of the wheel to calculate the longitudinal force from the road to the wheel.

Since the same wheel model is used for both the front and rear axles, a critical detail that should not be overlooked is that the wheel is attached to a positive frame at the rear axle, and a negative frame at the front axle. Hence, at the rear axle, a positive steering angle would steer the wheel counterclockwise, and at the front wheel, a positive steering angle would steer the wheel in the clockwise direction. This means that a positive steering angle produces opposite steer on the front and rear wheels

The steering controller plays a very crucial role in keeping the vehicle on the track. The below figure represents the steering controller.

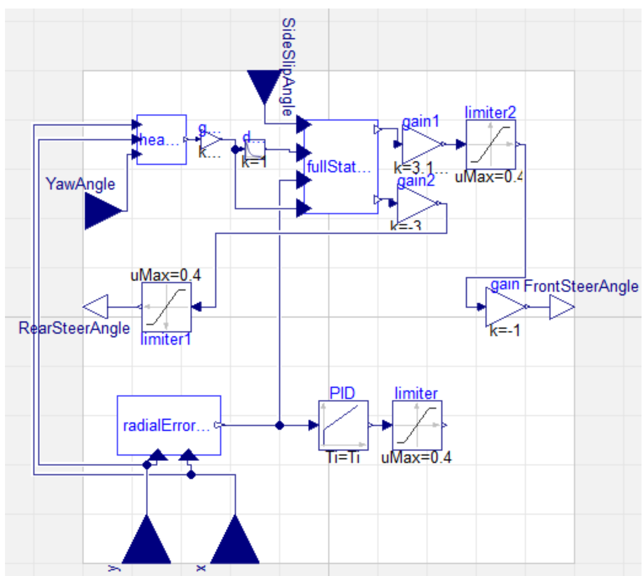

Fig. 4 Modelica Model of the All-Wheel Steering Controller

The all-wheel steering controller uses the global x, and y position of the vehicle to calculate the lateral error from the center of the track. The global x, and y positions, as well as the yaw angle of the vehicle is used to obtain the heading angle of the vehicle. The difference between the heading angle, as well as the desired heading angle which will be tangential to the track, is used to find the heading angle error. The side slip angle, the lateral error from the center of the track, the heading angle error, and its first derivative, are all fed into the gain matrix K from eq. 12. The outputs are then passed through a gain of -3.14/180, and are then taken to be the front and rear wheel steering angles as shown in Fig. 4. The steering angles are constrained to have a maximum absolute value of 0.4 rad.

The speed controller plays a crucial role in maintaining the vehicle velocity. The modelica model for the speed controller is shown in the below figure.

```
model SpeedController
  "Use the global positon of the car to determine the speed setpoint"
  Modelica.Blocks.Interfaces.RealInput x ;
  Modelica.Blocks.Interfaces.RealInput speed ;
  Modelica.Blocks.Interfaces.RealOutput DriveWheelTorque
    ;
    parameter Real p=60 "p-gain for speed control";
    parameter Modelica.Units.SI.Velocity StraightDesiredSpeed = 13 "p-gain for speed control";
    parameter Modelica.Units.SI.Velocity CurveDesiredSpeed = 11 "p-gain for speed control";
algorithm
  if abs(x) < 300 then DriveWheelTorque := p*(StraightDesiredSpeed-speed);
  else DriveWheelTorque := p*(CurveDesiredSpeed-speed);
  end if;

end SpeedController;
```

Fig 5. Modelica Model of the Speed Controller.

The speed controller uses a simple algorithm where conditional statements are used to ensure that the vehicle tries to achieve a desired speed of 13 m/s over the straight away, and 11 m/s on the curved parts of the track. However, for the simulation, 15m/s was the desired straight away speed.

Simulation Results

The vehicle is initially at rest, and a speed controller is used to provide torque to the front wheels. The all-wheel steering controller controls the steering angle of both the front and rear wheels to transverse the track. The parameter values that were used for the simulation can be obtained from the appendix section. Once the simulation is run, the following results are obtained.

The x and y coordinates of the vehicle is show in the below figure.

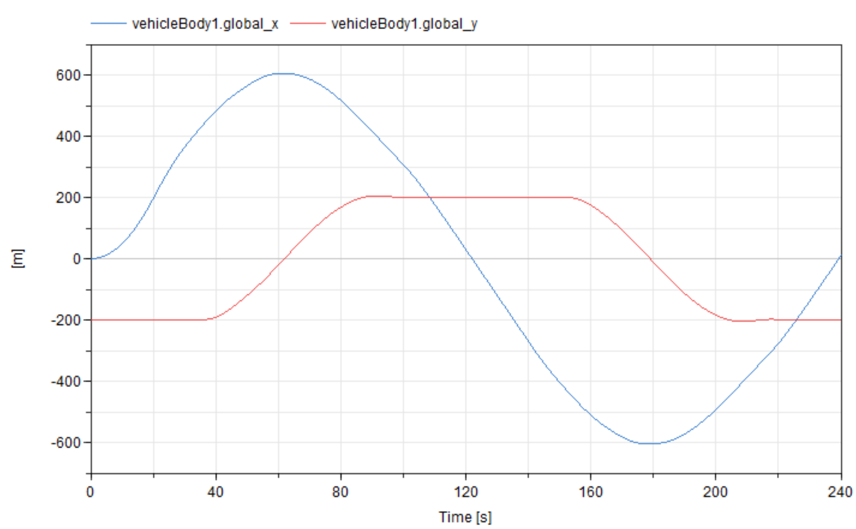

Fig. 6 Global x and y positions of the Vehicle

It can be seen that the vehicle has very little to no oscillations in the global y direction as it travels the circular path. Moreover, it has to be noted that, since the gains associated with the errors to be minimized are only proportional, the oscillations are slow to die down. The overall trajectory was close to what was required, and the lap was completed in approximately 240 s. The following figure shows the path taken by the vehicle.

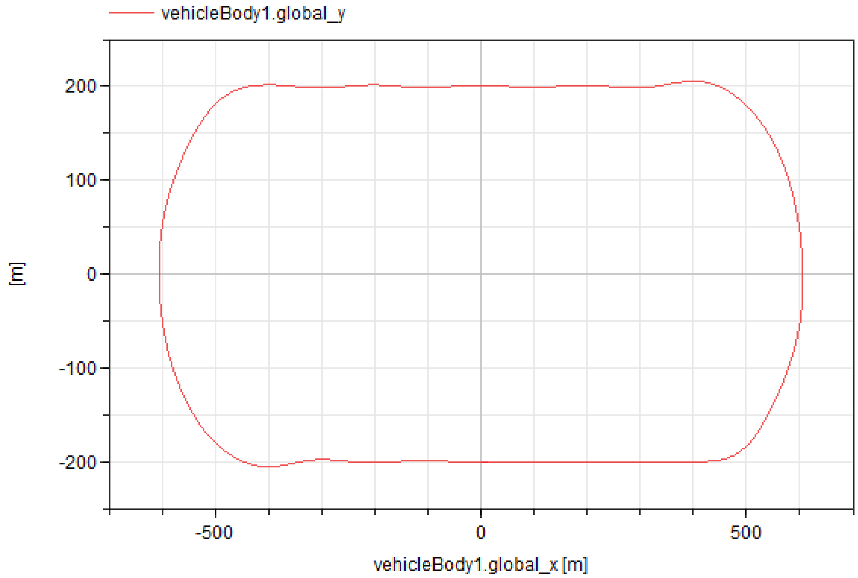

Fig. 7 Path Followed by the Vehicle

The vehicle velocity in the longitudinal and the lateral direction will be now analyzed.

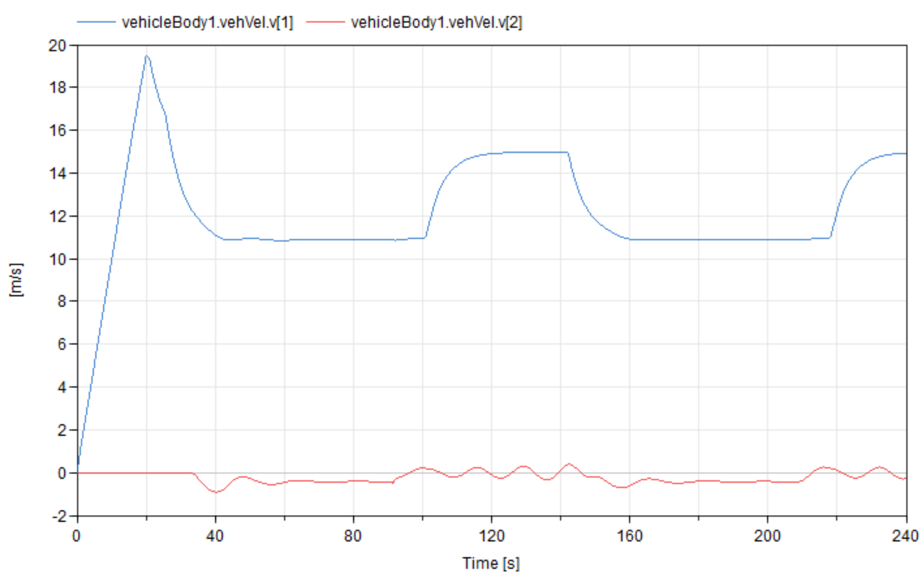

Fig. 8 Longitudinal and Lateral Velocity of the Vehicle

Ideally, the longitudinal velocity must smoothly increase to 15 m/s over the straight away, and smoothly reduce to 11 m/s over the curves, while maintaining a 0 lateral velocity. The simulation results from Fig. 8 indicates that the vehicle mostly follows this pattern. However, there are some overshoots in the longitudinal velocitiy, and some minor oscillations in the lateral velocity. This is because, the speed controller is defined for just the straight away and the curves. In addition, the state space representation assumes that the longitudinal velocity is constant. Whenever a sudden transition occurs, instabilities are introduced into the system. It has to be noted that the oscillations associated with the lateral velocity are indeed much smaller than the longitudinal velocity, however, they are slow to decay because similar to the previous result, the steering controls are only proportional to the errors of the states. A gain matrix that provides more system damping would reduce the oscillation.

The front and rear steering will now be analyzed.

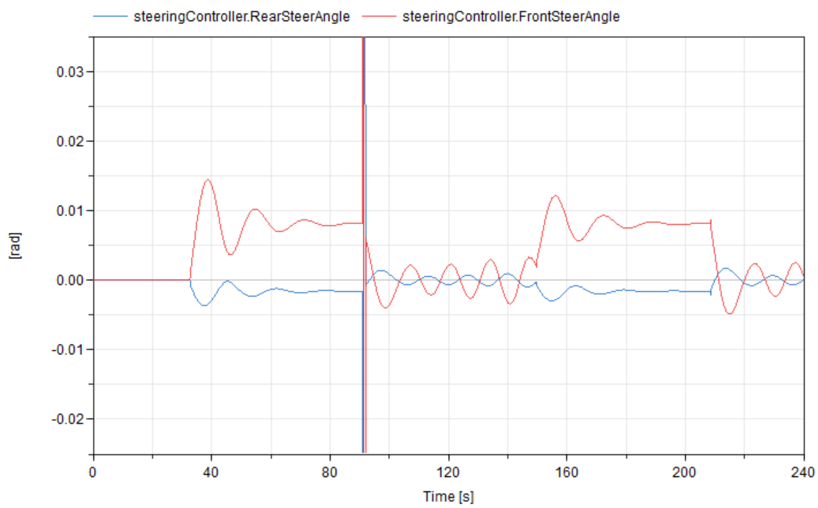

Fig. 9 Front and Rear Steering Angles of the Vehicle

From Fig. 9, it can be found that rear wheel for the most part was steered in the opposite direction as the front wheel, but with a smaller magnitude so as to reduce the side slip. In fact, this is what was expected, based on the control development section of the paper. However, it has to be noted that there are some small oscillations. These oscillations are proportional to the linear combination of all the states' errors. The lack of damping in the system causes the car to oscillate as soon as it hits the first curve, and then decay slowly.

We also expect that if the formulation of the state variables was better (yaw rate not equal to zero through the curve) and if the car were to begin to lose control that the wheels would potentially be steered in opposite directions as needed to reduce the side slip.

The states of the vehicle that were controlled, in order to transverse the desired trajectory was the vehicle side slip angle, the vehicle yaw rate, the lateral error from the center of the track, and finally the heading error. These states will now be analyzed in their respective order.

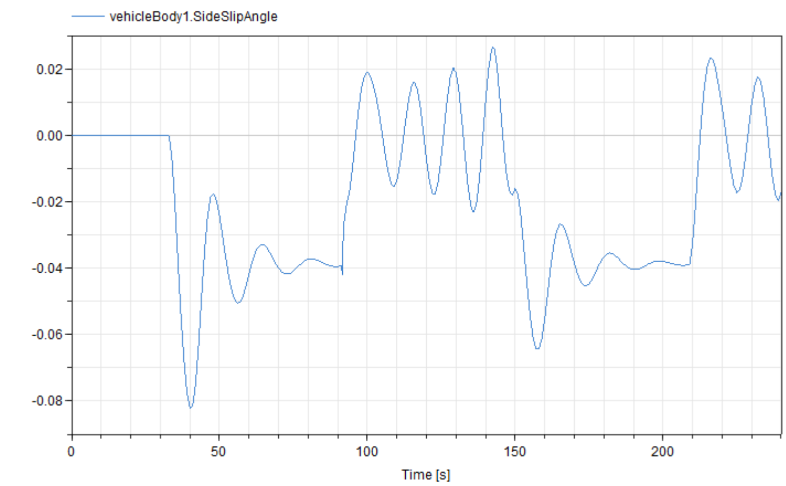

Fig. 10 Side Slip Angle [rad] of the Vehicle

The side slip angle must ideally be zero (for an all-wheel steer vehicle) and the vehicle's velocity must be tangent to the path. Hence, the steering controller decides inputs that would minimize this value. In fact, the largest penalty for the controller was associated with this state. Fig. 10 shows that the maximum side slip was 4.58 degrees (0.08 rad). These disturbances have a direct correlation with the oscillations that were found in the steering angles as well as the actual trajectory of the vehicle.

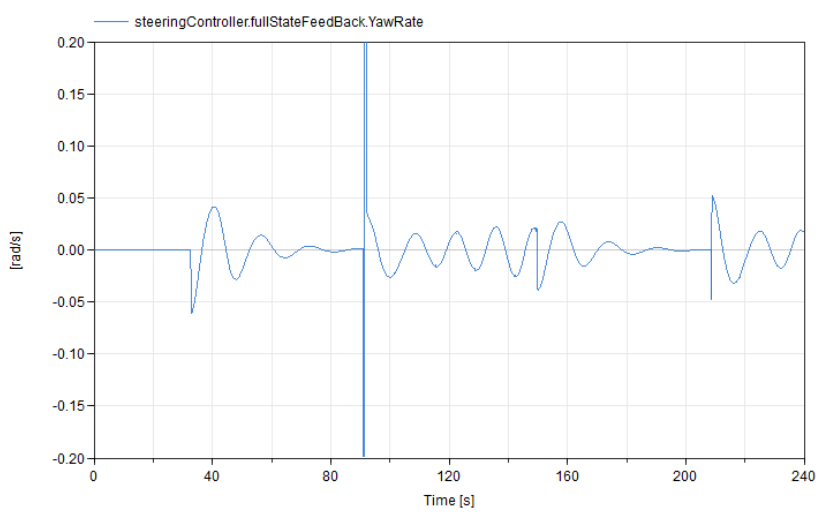

Fig. 11 Yaw Rate Error of the Vehicle

The yaw rate of the vehicle should be zero on the straight segments if its heading error and lateral error are zero. Thus, regulating yaw rate to zero is appropriate on the straights. However, on the curves, yaw rate should not be zero for steady state. Instead, the desired yaw rate should be $\omega = v_x/R$ where R is the radius of the path. Since the controller does not account for this non-zero set-point, the states: slip-angle and lateral error tend to settle at non-zero values through the curves.

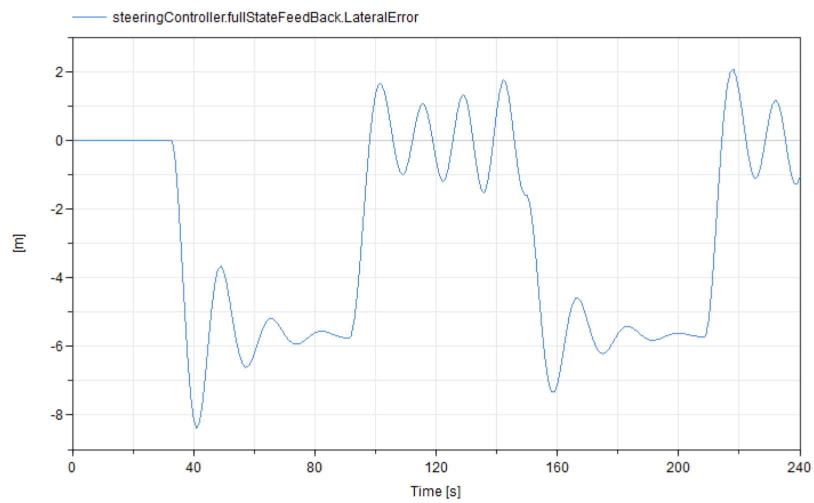

Fig. 12 Lateral Error of the Vehicle

The controller tries to reduce the lateral error down to zero. However, the error was kept between 2 and 8 meters, as seen in Fig. 12. Although the lateral error had a relatively higher importance weight, the combined effect of the errors in the other states caused the oscillations to occur. During the curves, $t \in [40,100] \cup [170,230]$, the radial error is settling to a non-zero value indicative of a problem in the control design.

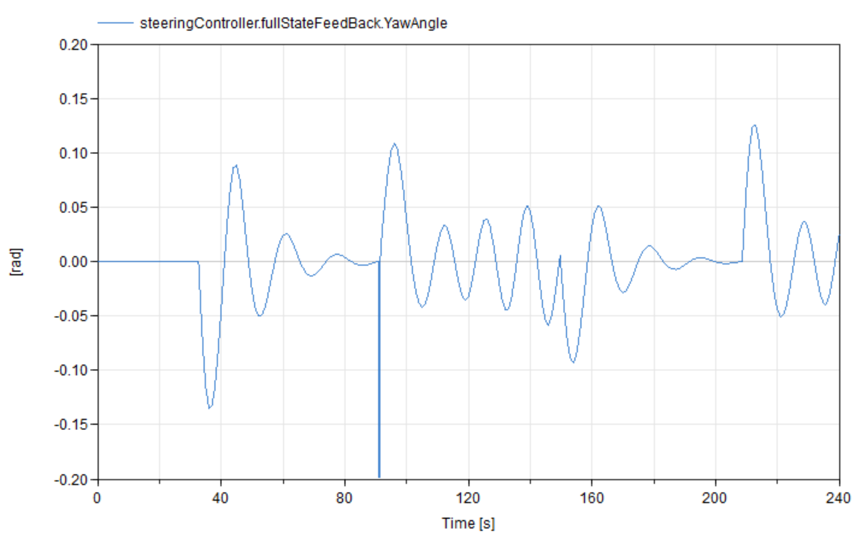

Fig. 13 Heading Angle Error of the Vehicle

The heading angle error is shown in Fig. 13. The controller tries to minimize this error. However, that causes overshoots, and therefore oscillations. It can be seen that this error varies between 0 and 0.13 rad. It has to be noted that the less important heading angle errors were generally greater than the high importance side slip angle errors. The sudden "spike" at $t \approx 88\, sec$ is due to the discontinuous calculation of the heading error as it transitions into the 2nd (top) straight segment.

Conclusion

1. The main objective of this paper was to model an all-wheel steer vehicle, and develop controllers for it so that it follows a predefined oval track in a short time, without losing control.
2. It was assumed that the car had access to a very good GPS, and IMU that would provide reasonable estimates of its position, tire slip angles, as well as the heading (yaw) angle.
3. The system dynamics were represented in the state space form such that the states were the lateral tracking error, vehicle side slip angle, the vehicle heading angle, and the yaw rate.
4. The 3 system inputs were front steering angle, rear steering angle and front wheel torque.
5. To maintain the desired system states, a linear quadratic regulator (LQR) was used to steer the vehicle.
6. A speed controller was added to adjust the torque on the front wheels.
7. The AWS controller drove the race car around the track in 240 sec.
8. From the simulation, it was found that there were very little oscillations. However, these oscillations die down slowly because of a few reasons.
    a. The state space representation assumes the vehicle longitudinal velocity to be a constant. However, this speed in reality, is not a constant, and causes disturbances.
    b. The speed controller was designed to have a different desired speed for the curves, and the straight away. However, the transition between the two is not very well described, and that induces instabilities.
    c. The controller relied on the proportional gains that corresponds to the errors. This meant that the oscillations would decay slowly since there the controller did not provide any damping effects.
9. In the future work, improvements have to be made to reduce the oscillations, and reduce the lap time. Two approach that would provide better results are as follows.
    a. Fix the problems mentioned with the controller
    b. Include a switching gains approach to accommodate the change in speed of the race car
    c. Introduce a better speed controller that has multiple modes, for a slow and smooth transition from one velocity to another.

Appendix

Table of Parameters used for Simulation

Steering PID and LQR Controller Gains

Parameter	Value	Unit
k (PID gain)	0.004	N/A
Ti (Time constant for integral term)	25	N/A
Td (Time constant for derivative term)	5	N/A
LQR Feedback Gain	Refer to eq. 12	N/A

Speed Controller

Parameter	Value	Unit
p (p gain for speed controller)	60	N/A
StraightDesiredSpeed	15	m/s
CurveDesiredSpeed	11	m/s

Simple Wheel Model

Parameter	Value	Unit
C_slip (Cornering stiffness longitudinal)	1000	N/A
C_alpha (Cornering stiffness lateral)	1000	N/A
r (Wheel radius)	0.3	m
alpha_deg_max (Max steering angle)	8	rad
omega0 (Wheel angular velocity)	40	rad/s

Vehicle Body

Parameter	Value	Unit
vehMass	1000	kg
vehInertiaXX	1000	$kg.m^2$
vehInertiaYY	1000	$kg.m^2$
vehInertiaZZ	1000	$kg.m^2$
c0 (Drag coefficient)	0	N/A
c1 (Drag coefficient for velocity)	0	N/A
c2 (Drag coefficient for velocity squared)	0	N/A
X0 (Initial X coordinate of vehicle in world frame)	0	m
Y0 (Initial Y coordinate of vehicle in world frame)	-200	m
Z0 (Initial Z coordinate of vehicle in world frame)	0	m

vx0 (Initial X velocity of vehicle in the world frame)	0	$m/_s$
vy0 (Initial Y velocity of vehicle in the world frame)	0	$m/_s$
vz0 (Initial Z velocity of vehicle in the world frame)	0	$m/_s$
lf (Distance between vehicle COM and front axle)	1	m
lr (Distance between vehicle COM and rear axle)	1	m

References

[1] T. Gordon, "Active Control of Vehicle Handling Dynamics," in *Vehicle Dynamics of Modern Passenger Cars*, 582 ed., P. Lugner, Ed., CISM International Centre for Mechanical Sciences. Springer International Publishing, 2019, pp. 205-246.

[2] G. Rill and A. A. Castro, "Lateral Dynamics," in *Road Vehicle Dynamics Fundamentals and Modeling with MATLAB*, CRC Press, 2019, pp. 297-313.

[3] M. Nagai and P. Raksincharoensak, "Advanced Chassis Control and Automated Driving," in *Vehicle Dynamics of Modern Passenger Cars*, 582 ed., P. Lugner, Ed., CISM International Centre for Mechanical Sciences. Springer International Publishing, 2019, p. 247–307.

[4] S. L. Brunton and J. N. Kutz, "Linear Control Theory," in *Data-Driven Science And Engineering Machine Learning, Dynamical Systems, and Control*, Cambridge University Press, 2019, pp. 229-266.