

Lateral Control using a MobileEye Camera for Lane Keeping Assist

Adityen Sudhakaran
DC 2017.003

M.Sc. Automotive Technology
Department of Mechanical Engineering
Dynamics & Control
Eindhoven University of Technology
Eindhoven, 5600 MB, The Netherlands
a.sudhakaran@student.tue.nl

February 9, 2017

Abstract

Automated technologies aim to improve the automotive industry by improving traffic flow, making the roads and the vehicles safe. One of the technologies that can enable vehicles to drive autonomously is a lane keeping assist (LKA) system that uses sensor and vehicle information to keep the vehicle in the lane. This is achieved by controlling the vehicle's steering angle. The aim of this project is to develop, implement, and test a lane keeping controller using information acquired from a MobilEye camera mounted on the vehicle's windshield. A linearised lateral vehicle model is developed and a controller is designed such that the vehicle maintains its position in the centre of the detected lane and orientation with respect to the lane. The resulting controller is implemented and tested in a prototype research vehicle that is equipped with a MobilEye camera. Results show that the MobilEye camera provides sufficient information to be utilised for lane keeping only if the controller is activated on highway scenarios where clear lane markings are visible on both sides of the vehicle.

Keywords - Lane Keeping Assist, MobilEye camera, autonomous vehicles, lateral control

Contents

1	Introduction	4
1.1	MobilEye Camera	5
1.2	Literature Review	8
1.3	Outline	8
2	Lateral Vehicle Dynamics	9
3	Controller Design	13
3.1	Desired Trajectories and Error Definition	14
4	Simulink Simulations	20
4.1	Combined Lateral Deviation and Relative Heading Simulation	20
5	Practical Implementation and Testing	23
5.1	Test 1 - Straight Road	24
5.2	Test 2 - Curved Road	26
5.3	Test 3 - Combined Curved and Straight Road	29
6	Discussion and Conclusion	33
6.1	Future Work	34
6.2	Conclusion	34
A	Derivation of Lateral Vehicle Model	37
B	MobilEye Connections	40
C	Model Verification	41
D	Simulink Simulations	44
D.1	Lateral Deviation Simulation	44
D.2	Relative Heading Simulation	46

Nomenclature

α_1	front tyre side slip angle [rad]
α_2	rear tyre side slip angle [rad]
$\ddot{\psi}$	Yaw Acceleration of vehicle [rad/s^2]
δ	Steering angle of the vehicle [rad]
$\dot{\psi}_h$	Derivative of relative heading between vehicle and lane
$\dot{\psi}$	Yaw Rate of vehicle [rad/s]
$\dot{\psi}_{des}$	Desired Yaw Rate of vehicle [rad/s]
ℓ	MobilEye look-ahead distance [m]
ψ_h	Relative Heading between vehicle and lane [rad]
$\psi_{h,des}$	Desired relative Heading between vehicle and lane [rad]
ρ	Curvature of road [m^{-1}]
a_x	Acceleration of vehicle in longitudinal direction [m/s^2]
a_y	Acceleration of vehicle in lateral direction [m/s^2]
$a_{y,des}$	Desired acceleration of vehicle in lateral direction [m/s^2]
c_f	front tyre cornering stiffness [N/rad]
c_r	rear tyre cornering stiffness [N/rad]
DLL	Distance to Left Lane [m]
DRL	Distance to Right Lane [m]
F_y	The total forces in the lateral direction [N]
I	Yaw Moment of Inertia about vertical (z-) axis [kgm^2]
ITS	Intelligent Transportation Systems
l	Wheelbase of vehicle [m]
LCL	Lane Confidence of Left Lane
LCR	Lane Confidence of Right Lane
LKA	Lane Keeping Assist

M	The Moment about vehicle Centre of Gravity
m	Mass of vehicle [kg]
R	Radius of Curvature [m]
T	Sample Time in Discrete Time Domain [s]
v_x	Velocity of vehicle in longitudinal direction [m/s]
v_y	Velocity of vehicle in lateral direction [m/s]
x	Position of vehicle in longitudinal direction
y	Position of vehicle in lateral direction [m]
y_{des}	Desired position of vehicle in lateral direction [m]

Chapter 1

Introduction

Today's roads include various types of vehicles that contribute to a global transportation network. As the need for transportation continues, there is an increase in road traffic which has led to negative impacts in terms of a congested road network, air pollution, and high fuel consumption. This steady increase in the number of vehicles on the roads has resulted in road traffic injuries. The WHO reports more than a million fatalities worldwide due to traffic-related problems [2, 10]. Road fatalities can be prevented by implementing safe technologies in vehicles that automate the driving tasks, assisting the driver. Since road accidents remain a major cause of death and injury, intelligent transport systems (*ITS*) play an important role in making vehicles and roads safer [4]. Statistics from the U.S. Transportation Research Board indicate that approximately 40 percent of all road fatalities occur when a single vehicle departs from the road and crashes. Additionally, sleepy and inattentive drivers are significantly more likely to be in a fatal accident compared to drivers with other performance-related factors [9]. This provides a great need for automated systems to control the steering of the vehicle such that the vehicle does not deviate from its current lane. In the context of this report, the vehicle will be classified as a SAE Level 2 automated vehicle [7] and as an "intelligent vehicle" that performs certain aspects of driving either autonomously or assists the driver to perform his/her driving functions more effectively, all resulting in enhanced safety, efficiency, and environmental impact [5].

The advances in vehicle technology, control systems, automotive sensing, and robotics have had a tremendous impact on automating vehicle technology to make it safer for drivers. One example of automated vehicle technology is a Lane Keeping Assist (*LKA*) system that automatically prevents the host vehicle from deviating from its current lane. This is made possible by controlling the steering angle of the vehicle while monitoring lane and vehicle parameters using sensor information. The importance of sensors for automated driving is critical in ensuring safety on the road. In order to automate systems, there must be sensors that provide reliable information that can be used with control algorithms that help keep the vehicle in the lane. In this report, a single MobilEye camera will be used as the primary sensor mounted at the top-centre of the windscreen, typically behind the rear view mirror.

The goal of this project is to utilise information from a MobilEye camera to create a simple Lane Keeping Assist (*LKA*) controller. The vehicle and camera will be modelled, simulated, and a simple controller will be designed and tested that will control the steering angle of a prototype research vehicle. The goal is to ensure that by controlling the vehicle steering using vehicle and camera information, it is possible to keep the vehicle within its lane.

1.1 MobilEye Camera

The MobilEye camera used is the C2-270 Driver Assistance System, seen in Figure 1.1. The vision sensor unit (camera) is mounted just behind the rear view mirror such that the maximum forward-facing view of the road is achieved. The MobilEye camera is connected via a CAN-bus connection



Figure 1.1: C2-270 MobilEye Camera

to the processor that is also installed in the vehicle. The parameters from the MobilEye camera are then extracted and used in the algorithms on MATLAB/Simulink. The MobilEye camera does not guarantee a 100% accuracy in detecting vehicles or lanes. The MobilEye camera manual explicitly states that the driver is to maintain full concentration on the road at all times. The system does have limitations as it is intended for paved roads with "clear lane markings" [8]. Any obstruction to the camera view also affects the functionality of the camera so a clear camera view is required at all times. These limitations are unavoidable as they are sensor limitations and must be obeyed before activation of the lane keeping controller.

Table 1.1: List of Utilised MobilEye Parameters

No.	Parameter	Definition	Range	Units
1.	Lane Confidence Left, LCL	The confidence of the camera in the detection of the left lane	0 – 3	-
2.	Lane Confidence Right, LCR	The confidence of the camera in the detection of the right lane	0 – 3	-
3.	Distance to Left Lane, DLL	The measured distance to the left lane	-40 : 0	m
4.	Distance to Right Lane, DRL	The measured distance to the right lane	0 : 40	m
5.	Lane Curvature, ρ	Curvature of the road	-0.12 : 0.12	m^{-1}
6.	Lane Heading, ψ_h	The relative heading between the lane and the vehicle	-1 : 1	rad

The use of the camera in this project is to detect lane markers on a road with a certain level of confidence, provide the distance to the lanes with respect to the location of the camera on the vehicle, detect the curvature and the relative heading of the lane with respect to the vehicle. The parameters utilised in this report are listed in Table 1.1. The use of these parameters to define the local position of the vehicle in the lane will be described in detail in the following chapter.

An additional feature that can be extracted from the MobilEye parameters is the lane equation. This is the equation that describes the lateral position of the detected lane as a function of the longitudinal distance. This equation is extracted from the MobilEye CAN bus protocol document [13]. Assuming an eagle eye view of the lane, the MobilEye lane equation is a quadratic function

that describes the curvature of the two detected lanes.

$$\begin{aligned} \text{LeftLane} &= \rho x^2 + \psi_h x + DLL \\ \text{RightLane} &= \rho x^2 + \psi_h x + DRL \end{aligned} \tag{1.1}$$

where ρ is the curvature of the road, ψ_h is the lane heading, DLL is the distance to the left lane,

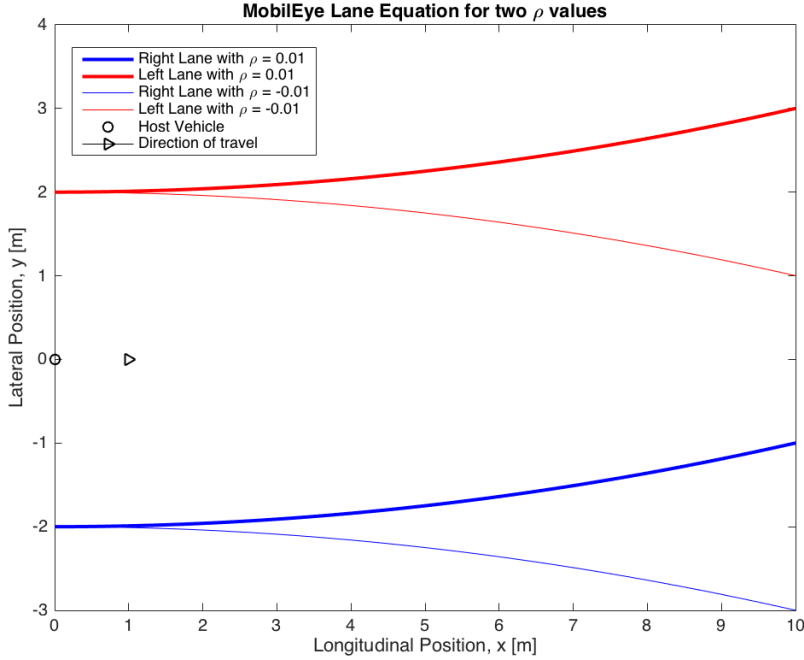


Figure 1.2: Lane Equation with positive and negative ρ values

DRL is the distance to the right lane, and x is a point in the longitudinal direction of the vehicle at which the lane is detected. The curvature, ρ , determines how sharp the curve in the road is and the sign of ρ determines which direction the road is curving. For positive values of ρ the lane is curving to the left (vehicle has to steer left), and for negative values of ρ the lane is curving to the right. For two different ρ values, and a constant ψ_h of 0.0001 rad in the road, the lane equation is visualised in Figure 1.2 above.

With the host vehicle positioned at point $(0, 0)$ in the figure below and facing the positive x -direction (towards the right), if the camera detects a curvature of $+0.01 \text{ m}^{-1}$, heading of 0.0001 rad , and a distance to the lanes as 2 m , then the equation of the road would be as seen by the thick red line and the thick blue line. If the camera detects a curvature of -0.01 m^{-1} , heading of 0.0001 rad , and a distance to the lanes as 2 m , then the equation of the road would be the thinner red line and the thinner blue line. In this manner, the road equation is determined and the road is "detected" by the camera. The values of ρ , ψ_h , DLL , and DRL are chosen at random to illustrate the lane equation. In reality the ρ values are much smaller to represent a wider curve in the road and the distance to the lanes are smaller to represent a narrower lane. The values are exaggerated for the figure.

The curvature value is also used to determine the Radius, R of the curve that the vehicle will be taking [13]. The radius of the curve is used for calculating the desired yaw rate of the vehicle, and further explained in chapter 3. The formula for the radius calculation is taken from the MobilEye CAN bus Protocol document provided with the camera product. The Radius, R , is calculated

using the formula

$$R = 1/2\rho \tag{1.2}$$

For example, a curvature of 0.01 m^{-1} used previously, the radius would be 50 m. This can be interpreted as a road that is the circumference of a circle with a radius of 50 m. The vehicle will be required to perform steady state cornering on that curve. At the next measurement, the curvature might change and as a result the radius will change too. The smaller the curvature value, the larger the radius of the curve in the road.

The magnitude and sign of the relative heading, ψ_h , determines in which direction the lane is pointing relative to the vehicle. It can be considered as the relative yaw angle between the vehicle and the road. If ψ_h is 0, it means the orientation of the lane and the orientation of the vehicle is the same. i.e. the lane and the vehicle are pointing in the same direction. If the value of ψ_h is positive, it means the vehicle is oriented towards the right. If the vehicle were to continue on it's current trajectory in a straight line without any change in rotation, it would approach the right lane. The camera perceives the lane to be pointing left. In order to illustrate this, assume the road is a straight road with no curves, as shown in figure 1.3, below. The host vehicle is at the centre of the lane. At the left of figure 1.3 the host vehicle is shown to have 0 rad relative heading. The vehicle in the middle of the figure is facing the left lane and will have negative relative heading because it perceives that the lane is facing right. Similarly, the vehicle on the right of figure 1.3 is facing the right lane and perceives the lane is facing left and will therefore result in a positive relative heading.

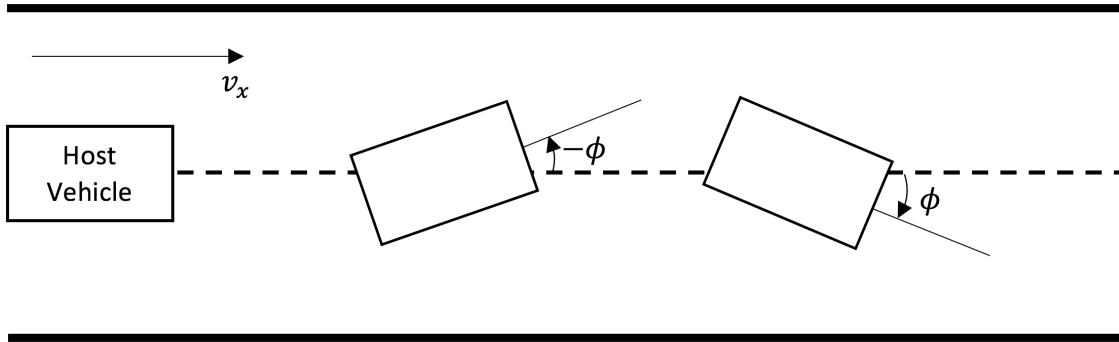


Figure 1.3: Lane Equation with positive, negative, and zero ψ_h values

This is important to include the relative heading in the model because the orientation of the vehicle also matters, not just the lateral position, when it comes to the position of the vehicle in the lane. This has an effect on how much steering needs to be done in order to get back to the desired centre of the lane.

The MobilEye camera appears to be a good vision sensor that provides sufficient information required for LKA. Based on this information, along with vehicle information, a vehicle model can be created. The camera does some processing of the detected image and provides data via CAN bus. The disadvantage of using a MobilEye camera is that the sensor operates as a black box. The actual look ahead distance of the camera and the time delay between when the image is detected and when the data is sent out by the camera are not known. This is a problem when designing an optimal controller as it is important to know the time delay of the camera processing. Therefore, a look ahead distance is assumed for this project and used as a way to include the delay in the image processing of the camera.

1.2 Literature Review

There are a variety of lane keeping controllers that have been presented in various scenarios. However, it is important to note what type of sensor is used with the designed controller. The current state of lateral control takes two paths. One that follows the path following model that utilises sensors to detect the vehicle's immediate surrounding and identify a path that the vehicle has to follow around various objects. The other path involves using the infrastructure and sensors to identify lane markings and use that as the reference path. The type of sensor and controllers vary accordingly and many variations can be made depending on cost constraint, time, etc.

For example, the camera model utilised in this report is presented in [6]. The paper presents a lateral control system based on vision sensors and equations that extract essential information from the sensors. The paper also studies the effect of time-delays of the vision sensor. The controllers presented in this paper are PI controllers but the data extracted from the camera are filtered. In using the MobilEye camera, only some of the data needs to be filtered as the camera itself does some processing of the image. In addition, the system presented in [12] not only uses a camera mounted on the windshield but also the radar to determine the path forward. This is particularly relevant in platooning scenarios.

On the other side, with more focus on vehicle model and control, the controller synthesised in [1] also focuses on the stability of lateral platoons so a heavy emphasis is placed on the controller design and performance. An LQR controller is synthesised and presented. Following that multiple vehicles are analysed for a platoon of infinite length and the controller was implemented and tested in a platoon of 2 vehicles.

It can be concluded that the current state of lateral control depends on the user's cost constraint, and access to hardware required to model the vehicle, sensors, and then design a controller. The utilisation of sophisticated sensors such as LIDAR to plan a path could significantly increase the cost required for the overall system. In comparison a camera works out to be relatively cheaper. While the selection of the required sensor may vary, it is paramount that the sensor be accurate enough to give the correct information at all times while driving such that the designed controller can provide safe automated technology.

1.3 Outline

The following chapter describes the lateral dynamics that was used to simulate the host vehicle. Chapter 3 describes the controller that was designed based on the vehicle model to control the steering angle of the host vehicle, along with the desired trajectories for the vehicle states. Chapter 4 presents the simulation that was run on MATLAB/Simulink using the designed controller and the results of the simulation. Chapter 5 presents the implementation of the controller in a prototype research vehicle and results of tests conducted in the vehicle. Finally, chapter 6 discusses the results and makes recommendations for future work, along with concluding comments.

Chapter 2

Lateral Vehicle Dynamics

In order to design a controller for the lateral control of the vehicle, an accurate model of the lateral dynamics is created. A linear single track model is used to show the lateral forces and moments on the front and rear tyres. The figure below shows the lateral vehicle model taken from a set of Vehicle Dynamics lecture notes, [3]. The two front tyres are represented as one front tyre which generates a total lateral force F_{y1} . The same is done for the two rear tyres which generates a lateral force F_{y2} . The adapted SAE coordinate system is adopted to model and simulate the lateral dynamics of the vehicle.

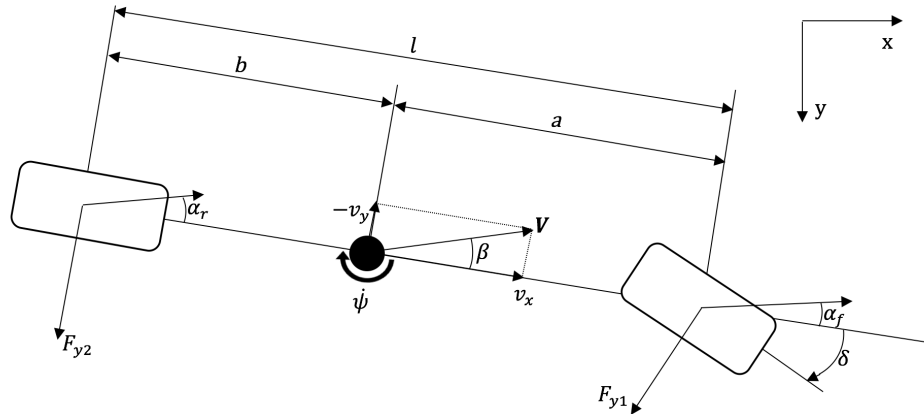


Figure 2.1: Linear Vehicle Model

From figure 2.1 above, two of the main equations of motion are considered, the sum of forces in the y-direction ($F_{y,i}$) and the moments in the z-direction, about the centre of gravity (M_z) and can be written as

$$\Sigma F_{y,i} = ma_y = m(\dot{v}_y + v_x\dot{\psi}) = F_{y1} + F_{y2} \quad (2.1)$$

$$\Sigma M_z = I_{zz}\ddot{\psi} = aF_{y1} - bF_{y2} \quad (2.2)$$

where m is the mass of the vehicle, a_y is the lateral acceleration caused by the lateral motion (v_y) and rotational motion ($\dot{\psi}$), v_x is the longitudinal velocity, I_{zz} is the vehicle yaw moment of inertia about the vertical (z-) axis of the vehicle, and $\ddot{\psi}$ is the yaw acceleration.

In addition to the vehicle dynamics model, to make the model represent the actual vehicle, a vision-based and curvature-based model is added from [6] and two more states are added to the system to represent the vehicle's position and orientation in the lane determined from the camera sensor. This is shown in figure 2.2 below.

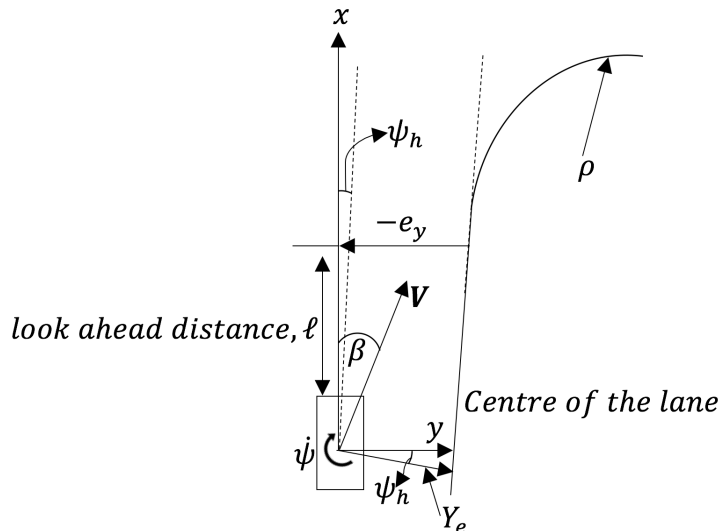


Figure 2.2: Vision Model

The first additional state is the lateral position of the vehicle within the lanes, and the second additional state is the relative heading between the vehicle and the lane. These parameters can be measured using the MobilEye camera. Therefore, two more equations are added to (2.1) and (2.2) to make the simulation closer to what the real vehicle would experience. From figure 2.2 above,

$$v_y = \beta v_x + \psi_h v_x \quad (2.3)$$

$$\dot{\psi}_h = \dot{\psi} - v_x \rho \quad (2.4)$$

where β is the vehicle side slip angle about the centre of gravity, v_y is the velocity in the lateral direction, ψ_h is the relative heading, and ρ is the road curvature.

The vision-based model equations show the change in what the camera would detect as the vehicle moves in the lateral direction within the lane [6]. If the vehicle is at one side of the road then the detected relative heading value would be a certain measured value. However, if the vehicle begins to move closer to the centre of the lane, then the perceived relative heading changes. Even though the road parameters is the same, the camera will detect a different value because the position and orientation of the vehicle has changed. This is included in the simulation with the equations (2.3) and (2.4). The vision model also adds the look ahead distance of the camera ℓ to the system as it is important to translate the position at which the camera detects the lane to the centre of gravity of the vehicle. The look-ahead distance, ℓ , is defined as the distance from the camera sensor to the look-ahead point. From figure 2.2, the perceived error in lateral position is defined as e_y . This is the position of the vehicle at the point that the camera detects the lane and is described in detail in the next chapter. However, that is not the actual position of the vehicle. The centre of gravity of the vehicle is at a distance Y_e away from the centre of the lane. However, this value is only valid on the straight region of the road. When a curvature is detected, then the lateral position of the vehicle, Y_e needs to account for the perceived curvature as the relative heading is now with respect to a curved lane. Assuming the change in curvature is linear ($\dot{\rho} = \text{constant}$) and slow, as derived

in [6], the error in lateral position of the centre of mass of the vehicle, Y_e , from figure 2.2, is defined by the equation

$$Y_e = (e_y + \ell \tan(\psi_h) + \frac{\ell^2}{2v_x} \rho) \cos(\psi_h) \quad (2.5)$$

Assuming small values for ψ_h and that $\sin(-\psi_h) = -\sin(\psi_h) = \psi_h$ and also $\cos(-\psi_h) = 1$, and measuring from the distance of the centre of gravity to the look ahead point,

$$Y_e = e_y + (a + l_c + \ell) \psi_h + \frac{\ell^2}{2v_x} \rho \quad (2.6)$$

where a is the distance from the front axle to the centre of the mass of the vehicle, l_c is the distance from the front axle to the camera sensor, ℓ is the distance from the camera sensor to the look ahead point, v_x is the vehicle longitudinal velocity, and ρ is the actual curvature of the road.

Since it is not possible to determine ℓ for the MobilEye camera, it can be assumed to be changing with the longitudinal velocity of the vehicle in m/s , every second. For example, if the vehicle is being driven at $20 m/s$ then the look-ahead is assumed to be $20 m$. The vision-based model adds curvature, ρ , as an input to the model. However this is only to **simulate** the perceived change in road parameters as viewed by a vision sensor. The curvature, ρ , is **not** a controlled input. The only input that can be controlled and sent to the actual vehicle is the steering angle, δ .

From (2.1) and (2.2), we can derive the states $\dot{\psi}$ and β . In addition, the equations (2.3) and (2.4) add the relative heading ψ_h and lateral position y as states to the model. Choosing these parameters as the states of the vehicle give an accurate representation of the vehicle that uses a vision sensor. The dynamics of the vehicle can then be written in the general state space form

$$\begin{aligned} \dot{\underline{x}} &= \underline{A}\underline{x} + \underline{B}u \\ \underline{y} &= \underline{C}\underline{x} + \underline{D}u \end{aligned} \quad (2.7)$$

where $\dot{\underline{x}}$ represents a vector of the derivatives of the state variables, \underline{x} is a vector of the state variables $[\dot{\psi} \quad \beta \quad \psi_h \quad y]^T$, \underline{y} is a vector of the measured outputs $[\dot{\psi} \quad a_y \quad \psi_h \quad y]^T$. The outputs are the measured parameters that are determined from the vehicle sensor measurements or measurements made by the MobilEye camera. The yaw rate, $\dot{\psi}$, and the lateral acceleration, a_y , are measured using the sensors in the vehicle. The relative heading, ψ_h , and lateral position, y , are measured using the detected lane information from MobilEye camera. In case of the simulation, they are the outputs of the vehicle model and will be used as feedback terms for the controller. Since the vehicle side slip can not be measured in the vehicle, the lateral acceleration of the vehicle will be one of the measured outputs. Using (2.3), an expression of lateral acceleration a_y is derived in terms of the states. This is also done in the simulation so that the simulation more accurately represents the vehicle.

The following state space model is created for the lateral dynamics and vision dynamics of the vehicle.

$$\begin{bmatrix} \ddot{\psi} \\ \dot{\beta} \\ \dot{\psi}_h \\ v_y \end{bmatrix} = \begin{bmatrix} \frac{-a^2 c_f - b^2 c_r}{I v_x} & \frac{-a c_f + b c_r}{I} & 0 & 0 \\ \frac{-a c_f + b c_r - m v_x^2}{m v_x^2} & \frac{-c_f - c_r}{m v_x} & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & v_x & v_x & 0 \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \beta \\ \psi_h \\ y \end{bmatrix} + \begin{bmatrix} \frac{a c_f}{I} & 0 \\ \frac{c_f}{m v_x} & 0 \\ 0 & -v_x \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \rho \end{bmatrix} \quad (2.8)$$

$$\begin{bmatrix} \dot{\psi} \\ a_y \\ \psi_h \\ y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{-a c_f + b c_r}{m v_x} & \frac{-c_f - c_r + m a_x}{m} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \beta \\ \psi_h \\ y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{c_f}{m} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \rho \end{bmatrix} \quad (2.9)$$

Table 2.1: Estimated Vehicle Parameters for Vehicle Model

No.	Parameter	Value	Units
1.	Mass, m	1380	kg
2.	Wheelbase, l	2.70	m
3.	Distance from CoG to front axle, a	1.62	m
4.	Distance from CoG to rear axle, b	1.08	m
5.	Yaw Moment of Inertia, I	$2.6611 * 10^3$	kgm^2
6.	Front wheel cornering stiffness, c_f	77000	N/rad
7.	Rear wheel cornering stiffness, c_r	130000	N/rad

For the full derivation of the state space model, see Appendix A.

The vehicle parameters have been measured or estimated and are listed in Table 2.1. These can be adapted to suit the type of vehicle that needs to be modelled. The parameters represent a 2010 Toyota Prius. The parameters listed in Table 2.1, along with the state space model are implemented in MATLAB/Simulink, and run in discrete time with a sample time, T of 0.01 s.

The model created in MATLAB/Simulink is tested by placing known inputs of δ and ρ to check the response of the model. A constant velocity of 20 m/s is chosen at the start of each simulation and maintained throughout the simulation because that is the expected average velocity of the vehicle while the controller is active. Known inputs of $\delta = 0 \text{ rad}$, step input of $\delta = 0.001 \text{ rad}$, and a sine input of $\delta = 0.01 \sin(3t) \text{ rad}$. The results of the model test is shown in Appendix C. The model can be considered valid as the simulated vehicle behaves in a similar fashion to what would be expected of an actual vehicle given the known steering angle, δ , input. The linear vehicle model and camera model is valid up to lateral accelerations of 3 m/s^2 as the vehicle behaves linearly only during low lateral accelerations. With a verified linear vehicle model, a controller can be designed.

Chapter 3

Controller Design

The lane keeping controller that needs to be designed to control the steering of the vehicle needs to satisfy two objectives: the lateral position error needs to go to zero, and the relative heading should also go to zero so that the vehicle is close to the centre of the lane and facing in the same direction as the lane. In order to keep the controller simple, as the focus of this report is on the data output of the MobilEye to create an accurate vehicle model, a combination of P and PD controllers was chosen as the controller. This is because valid reference points can be created for each of the outputs of the vehicle model. Therefore, with the created references, along with sensors available to measure the outputs (from camera and vehicle) variable, a set of gains can be used to determine the steering angle, δ , to control the vehicle.

The lateral vehicle dynamics model from the previous chapter was implemented in Simulink. The state space model is repeated below for convenience.

$$\begin{aligned}\dot{\underline{x}} &= \underline{A}\underline{x} + \underline{B}u \\ \underline{y} &= \underline{C}\underline{x} + \underline{D}u\end{aligned}\tag{3.1}$$

$$\begin{bmatrix} \ddot{\psi} \\ \dot{\beta} \\ \dot{\psi}_h \\ v_y \end{bmatrix} = \begin{bmatrix} \frac{-a^2c_f - b^2c_r}{Iv_x} & \frac{-ac_f + bc_r}{I} & 0 & 0 \\ \frac{-ac_f + bc_r - mv_x^2}{mv_x^2} & \frac{-c_f - c_r}{mv_x} & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & v_x & v_x & 0 \end{bmatrix} \begin{bmatrix} \psi \\ \beta \\ \psi_h \\ y \end{bmatrix} + \begin{bmatrix} \frac{ac_f}{I} & 0 \\ \frac{c_f}{mv_x} & 0 \\ 0 & -v_x \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \rho \end{bmatrix}\tag{3.2}$$

$$\begin{bmatrix} \dot{\psi} \\ a_y \\ \psi_h \\ y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{-ac_f + bc_r}{mv_x} & \frac{-c_f - c_r + ma_x}{m} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \psi \\ \beta \\ \psi_h \\ y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{c_f}{m} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \rho \end{bmatrix}\tag{3.3}$$

The first check is to find the eigenvalues of the \underline{A} matrix. A constant velocity, v_x , of 20 m/s is chosen because that would be the average velocity of the vehicle during most test scenarios. At this velocity, there are two eigenvalues at 0 and a pair of complex eigenvalues with negative real parts. It is observed that the two eigenvalues at 0, remain at 0 for all v_x . The two complex eigenvalues in the LHP get closer to 0 as $v_x \rightarrow 0$ and as v_x increases beyond 30m/s. The complex eigenvalues are related to the dynamics of the vehicle. The eigenvalues at 0 are related to the lateral position and relative heading added to the system. The important observation is that there are no right half plane poles, which means the open loop is stable. The only concern are the two poles at 0. It is also found that for all v_x between 1m/s – 30m/s, which is the expected range for the velocity of the vehicle with an active LKA controller, there are always two eigenvalues at 0 and two eigenvalues in the left half plane.

The second check is to determine the controllability and observability of the system to find out if a controller can be designed for the system and if the internal dynamics of the system can be

observed based on the inputs and external outputs. To check for controllability, the rank of the controllability matrix, C , must equal the number of states, $n = 4$. To check for observability, the rank of the observability matrix, O , must equal the number of states, $n = 4$. The controllability matrix is determined as follows

$$\text{rank}(C) = \text{rank} \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix}$$

The observability matrix is determined as follows

$$\text{rank}(O) = \text{rank} \begin{bmatrix} C & CA & CA^2 & \dots & CA^{n-1} \end{bmatrix}^T$$

Using the A , B , and C matrices from the state space model, the $\text{rank}(C)$ and the $\text{rank}(O)$ are found to be 4. Therefore the system is controllable and observable.

Now that the system is found to be controllable, a controller can be designed. It was determined that because appropriate reference trajectories can be created for the output variables of the system, and the outputs of the model can be measured in the vehicle using sensors, a set of proportional and damping gains can be used to control the errors in the variables. In order to do this, the desired trajectories and errors need to be defined.

3.1 Desired Trajectories and Error Definition

The desired trajectories for $\dot{\psi}_{des}$ and $a_{y,des}$ are calculated based on the assumption that the road curvature is changing slowly, and that the front wheels are following the same curved trajectory because a bicycle model is used to describe the lateral dynamics [11]. In addition, from the MobilEye camera, the parameters listed in Table 1.1 are extracted and utilised for the reference trajectories for y_{des} and $\psi_{h,des}$.

The $\dot{\psi}_{des}$ reference is calculated based on the curvature of the road, ρ , using the following equation

$$\dot{\psi}_{des} = \frac{v_x}{R} \quad (3.4)$$

where R is the radius of curvature calculated from equation (1.2). Since the radius is calculated based on curvature of the road, if the road is straight, then the curvature value is 0 [m^{-1}]. According to (1.2), this would provide a radius of ∞ . This is accounted for in the simulation and when implementing in the vehicle by setting $\dot{\psi}_{des}$ to 0 if the detected curvature is 0 m^{-1} . The measured value for $\dot{\psi}$ comes from the yaw rate sensor of the vehicle. The error in yaw rate is then defined as

$$e_{\dot{\psi}} = \dot{\psi} - \dot{\psi}_{des} \quad (3.5)$$

where $\dot{\psi}_{des}$ is calculated from (3.4) and $\dot{\psi}$ is measured from the vehicle.

The $a_{y,des}$ reference is calculated based on the desired yaw rate, $\dot{\psi}_{des}$, from (3.4). This is utilised because a significant lateral acceleration is desired specifically during cornering. On a straight road, the vehicle is required to have minimal rotation and minimal lateral movement so lateral acceleration is not desired. The lateral position error and relative heading errors are expected to dominate on a straight road. It is also not desired for the vehicle to accelerate in the lateral direction. Therefore, it is determined that $a_{y,des}$ be 0 m/s^{-1} on a straight road. However, during cornering, $a_{y,des}$ will follow the equation

$$a_{y,des} = v_x \dot{\psi}_{des} \quad (3.6)$$

where, $\dot{\psi}_{des}$ is the desired yaw rate calculated from (3.4). The actual lateral acceleration measurement comes from the vehicle sensor. The error in lateral acceleration can then be defined as

$$e_{a_y} = a_y - a_{y,des} \quad (3.7)$$

where $a_{y,des}$ is calculated from (3.6) and a_y is measured using the vehicle sensor.

Before going on to the desired relative heading, $\psi_{h,des}$, the desired lateral position y_{des} is first computed. Using information from the MobilEye camera, and the lane equation as described in equation (1.1) and figure 1.2, the equation of the left and right lanes are computed. It follows that the desired lateral position y_{des} is the trajectory of the centreline between the two lanes and is therefore,

$$y_{des} = \rho x^2 + \psi_h x + 0 \quad (3.8)$$

where 0 is the centre of the lane.

The y -position of the vehicle in the lane is then computed by measuring the distances to both the lanes at the look-ahead distance using the MobilEye camera. The formula gives the distance of the vehicle with respect to the centre of the lane.

$$y = \frac{|DLL| - |DRL|}{2} \quad (3.9)$$

The error in lateral position is then described as

$$e_y = y - y_{des} \quad (3.10)$$

However, this is not the actual lateral position of the vehicle. The actual lateral position of the vehicle at the centre of gravity uses the error calculated in (3.10), the look ahead distance, and the road curvature, and is described in (2.6). It is repeated below for convenience.

$$Y_e = e_y + (a + l_c + \ell)\psi_h + \frac{\ell^2}{2v_x}\rho \quad (3.11)$$

where a is the distance from the front axle to the centre of the mass of the vehicle, l_c is the distance from the front axle to the camera sensor, ℓ is the distance from the camera sensor to the look ahead point, v_x is the vehicle longitudinal velocity, and ρ is the actual curvature of the road.

From MobilEye data, it is observed that there are sudden losses in distances to the lane values even with the lane confidence is 2. In order to account for this, a virtual lane is implemented. If the MobilEye camera loses one of the lanes then a virtual lane is created to replace the lost lane. Several measurements were made by manually driving the vehicle on properly marked lanes. The driver was asked to stay in approximately the middle of the lane and the distance to both lanes were measured. This test was performed on a highway and several country roads and city roads. It was determined that the average distance to a detected lane is around 1.6 m. Therefore a virtual lane is created at 1.6 m away from the vehicle. This is utilised so that the lane keeping controller is still active on roads that only have a one lane marking in the middle of the road. For example, on roads where one side has a lane marking but the other side has a curb, rather than a lane marking. There are instances where the MobilEye detects the curb as a lane marking and gives a value for the distance to the lane. However, the confidence of that measurement drops. The virtual lane allows the lane keeping controller to be active and not require manual override from the driver. This increases the scope of MobilEye camera as it can be used for LKA on roads with only one proper lane marking. For the simulation, the virtual lane is not included as the lateral position is determined from the vehicle model. However, it is added to the vehicle when implementing the controller in the vehicle's Simulink scheme.

Now the desired relative heading, $\psi_{h,des}$, can be calculated based on the lateral deviation of the host vehicle from the centre of the lane. This is because if a controller is used to control only the y -position of the vehicle then at the centre of the lane, the controller will not steer the vehicle, regardless of the heading. This will result in oscillations that could increase causing the vehicle to

steer into the next lane, which is undesired and dangerous. The desired heading at the centre of the lane needs to be zero meaning the vehicle is required to be oriented tangential to the lane. Therefore, the relative heading of the host vehicle needs to be based on the lateral position. This is achieved by creating $\psi_{h,des}$ as a function of the error in lateral position. The maximum ψ_h was determined based on measurements taken with the test vehicle. The vehicle was oriented at different lateral positions on a lane and the heading was measured. This was included in the equations to determine the desired heading. This covers a range of possibilities of position and orientation that the vehicle could be at when on the road.

The lane is divided into three sections as illustrated in Figure 3.1. The section in the middle of

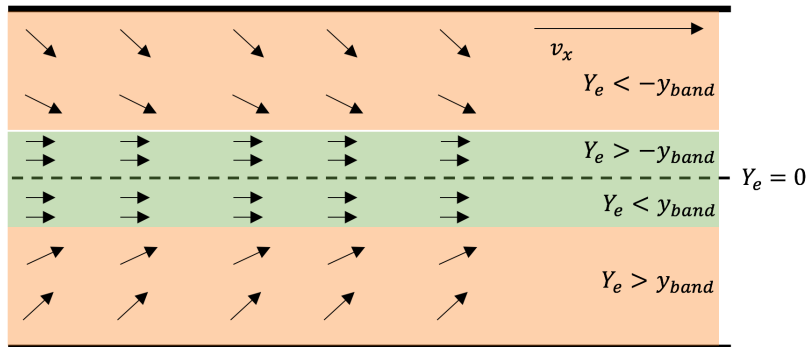


Figure 3.1: Regions of the lane defined by lateral position error

the lane (green band) represents a region of a tuneable width in lateral distance, i.e. y_{band} m at either side of the centre of the lane. Between these limits, the desired relative heading, $\psi_{h,des}$, is 0 rad. Within these limits it is desired that the vehicle be oriented in the same direction as the lane. If the vehicle drifts slightly away from the centre of the lane but is still within the band then the vehicle is to only correct for lateral position error and not heading error. This prevents unnecessary rotation and oscillations of the vehicle. The direction of the arrow represents the desired orientation of the vehicle if it were in that lateral position in the lane. The horizontal arrows indicate 0 rad desired relative heading.

Beyond these limits ($|Y_e| > y_{band}$, orange bands in figure 3.1), $\psi_{h,des}$ is a quadratic function of lateral position. A coefficient $\psi_{h,slope}$ is added in front of the expression to determine how much the vehicle is required to rotate within the lane. The higher the $\psi_{h,slope}$ value, the more the vehicle will be required to rotate towards the centre of the lane. This is a tuneable parameter that is created to penalise the relative heading error. This value is kept at a low magnitude as excessive rotation of the vehicle in the lane is not desired, especially at high speeds. The equation describing the desired relative heading $\psi_{h,des}$ is

$$\psi_{h,des} = \begin{cases} 0, & -y_{band} \leq Y_e \leq y_{band} \\ \psi_{h,slope}((Y_e + y_{band})^2), & Y_e > y_{band} \\ -\psi_{h,slope}((Y_e - y_{band})^2), & Y_e < -y_{band} \end{cases} \quad (3.12)$$

The y_{band} is varied based on the speed of the vehicle and the curvature of the road. At high speeds, a wider band is implemented because rotation of the vehicle is not desired. If at higher speeds, a significant curvature is detected, then y_{band} is reduced slightly to allow for e_{ψ_h} to build up that would assist the vehicle to steer around the approaching corner. At lower speeds, a narrower band is implemented because more rotation is required as higher curvature values are expected. This is further elaborated when the controller is implemented in the vehicle for testing. Assigning the desired heading the values of the above function provides for a smoother driving style. It also

prevents oscillations at the centre of the lane. This also mimics the driving style of a human driver. The actual relative lane heading ψ_h is MobilEye measurement. The error in relative heading is then defined as

$$e_{\psi_h} = \psi_h - \psi_{h,des} \quad (3.13)$$

where $\psi_{h,des}$ is calculated from (3.12) and ψ_h is measured from the MobilEye camera.

Using the above errors in the output variables, the steering angle of the vehicle can be controlled. A block scheme of the full control system is shown in Figure 3.2 below.

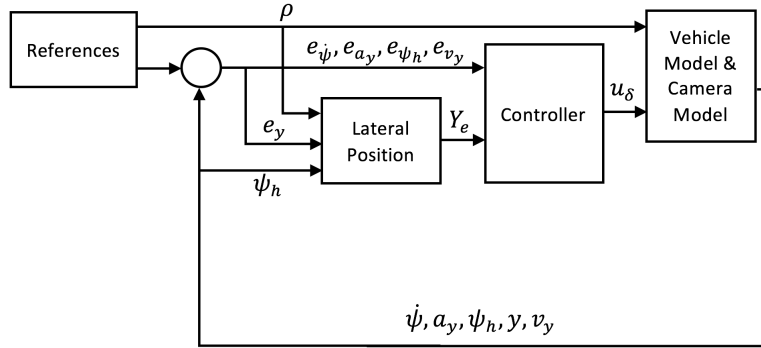


Figure 3.2: Block Scheme for Simulation

For the simulation, there are two inputs to the model, u_{δ} and ρ . The curvature, ρ , is the actual curvature of the road and is not a controlled input. The only controlled input to the model is the steering angle δ . However, as described in the previous chapter, the effect of the curved road and the look ahead distance on the actual lateral position of the vehicle needs to be simulated. This is computed in the 'Lateral Position' block just before the controller using equation (3.11).

The controller can be designed as a weighted sum of the errors in lateral position Y_e from (3.11), relative heading from (3.13), yaw rate from (3.5) and lateral acceleration from (3.7). The controller for the steering angle, δ , is shown in the figure below where, K_{P1} , K_D , K_{P2} , and PD are gains

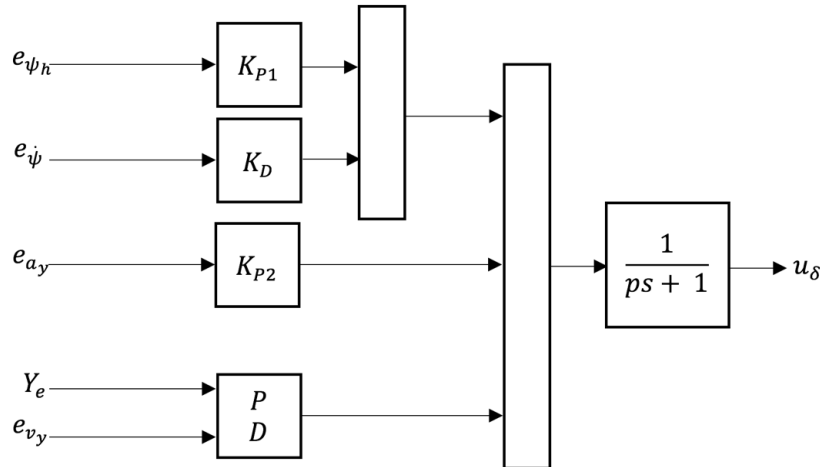


Figure 3.3: Control Scheme for Simulation to get δ input

that are manually tuned, and p is a tuneable parameter for a first order filter. Since the relative

heading can be considered as the yaw angle between the vehicle and the lane the relative heading, ψ_h , and the yaw rate, $\dot{\psi}$, form a PD controller. The gain for error in relative heading is K_{P1} , the gain for error in yaw rate is K_D . For the lateral acceleration, a_y , a simple proportional gain K_{P2} is used. The error in lateral position, Y_e , is controlled using a PD controller with the proportional gain on lateral position and a derivative gain for the derivative of lateral position error. This can be considered as the error in lateral velocity v_y . Therefore, the error in lateral position Y_e and the lateral velocity, v_y , form a PD controller. It is predicted that if only a P controller is used to control the lateral position then high oscillations in lateral position will be expected and this is not desired, nor is it considered safe. Therefore, v_y is extracted as an output from the model and added to the existing 4 outputs. Therefore the output vector, y , from (3.3), changes from $[\dot{\psi} \quad a_y \quad \psi_h \quad y]^T$ to $[\dot{\psi} \quad a_y \quad \psi_h \quad y \quad v_y]^T$. This is done only for the simulation. The primary goal is to control the steering angle such that errors in lateral position and relative heading go to 0 rad. The controller needs to control the error in position such that the vehicle goes to the centre of the lane. In addition, it needs to control the orientation such that the vehicle is oriented in the same direction as the lane. The low pass filter is added to provide a smoother steering input to the vehicle and to remove the expected noise from the vehicle sensors. The response of the system is checked after the set of gains is implemented. This can be seen in the figure below. Given the initial conditions

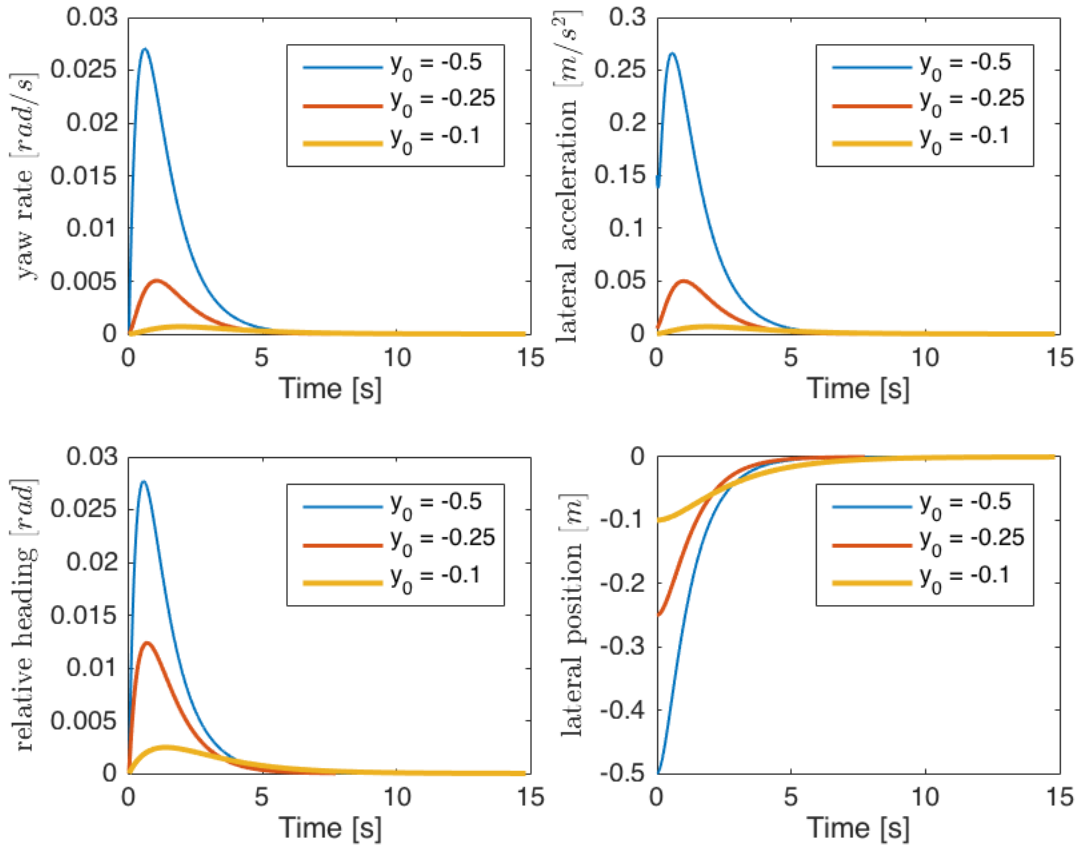


Figure 3.4: Response of system with given initial conditions

$[\dot{\psi} = 0 \quad \beta = 0 \quad \psi_h = 0]^T$, the response is plotted in Figure 3.4. The initial conditions of the lateral position are varied from $y_0 = -0.5$ m to $y_0 = -0.25$ m and $y_0 = -0.1$ m. The initial

conditions state that the vehicle is y_0 m away from the centre of the lane towards the left lane and is facing in the same direction as the lane ($\psi_h = 0$). All the states gradually converge to the equilibrium point $(0,0,0,0)$ in approximately 6 seconds and the set of gains determined will bring the vehicle back to the centre of the lane. There is a change in relative heading, ψ_h (bottom left in figure 3.4), which increases and then settles back to 0 rad . This is expected because the vehicle rotates to face the centre of the lane. As the lateral position reaches 0 m , the centre of the lane, the heading also goes to 0 rad . This means that the vehicle corrects itself by rotating back to the same direction as the lane. The system can be considered stable as the closed loop eigenvalues are all in the negative left half plane. Various scenarios can now be simulated on Simulink.

Chapter 4

Simulink Simulations

For the simulation in MATLAB/Simulink, the same block and control schemes of figures 3.2 and 3.3 is used. The criteria for evaluation during the simulation are as follows:

Table 4.1: Criteria to evaluate simulations

No.	Criteria
1.	The vehicle must get back to the centre of the lane
2.	The vehicle should not oscillate about the desired lateral position set-point
3.	The vehicle should orient itself with the lane
4.	The controller should work at expected driving speeds (10 - 20 m/s)
5.	The steering angle values should be small at high speeds

Multiple simulations are run to show various situations that a vehicle might encounter when on the road. One simulation is presented in this chapter while two others are presented in Appendix *D*.

The vehicle is simulated at 3 constant velocities of 10 *m/s*, 15 *m/s*, and 20 *m/s*. The controller is expected to be active during these driving velocities and therefore these are ideal velocities to be simulated. In addition, lateral offsets and relative heading offsets are chosen at values that are expected during a real driving scenario as initial conditions for the simulation. For example, the vehicle will not be in the centre of the lane when the controller is activated or the vehicle may be in the centre of the lane but could be oriented facing one of the lanes.

4.1 Combined Lateral Deviation and Relative Heading Simulation

This simulation is conducted to simulate what the vehicle would do if there is both a lateral offset **and** a relative heading offset. This means that at the start of the simulation, the vehicle is not at the centre of the lane and is facing one of the lanes. This would require that the vehicle correct both laterally and rotationally to get back to the centre of the lane. The lateral offset of 0.2 *m* is chosen which means the vehicle will be closer to the right lane. The vehicle will have a positive initial heading of $\psi_{h,0} = 0.002 \text{ rad}$ which means it will also be facing the right lane (away from the centre of the lane).

Table 4.2: Parameters for combined lateral deviation and relative heading simulation

No.	Parameter	Value	Units
1.	Acceleration, a_x	0	m/s^2
2.	Curvature, ρ	0	m^{-1}
3.	Initial Heading, $\psi_{h,0}$	0.002	rad

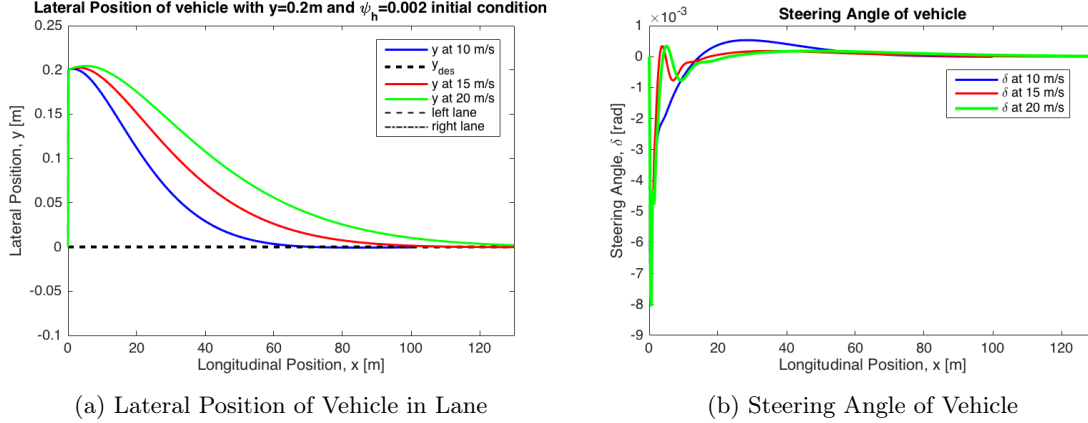


Figure 4.1: Simulation Results for Combined Lateral Deviation & Relative Heading Simulation

From the simulation results shown in figures 4.1 and 4.2, it can be seen that the vehicle corrects itself back to the centre of the lane and corrects its orientation to that of the lane. From the lateral position plot (4.1a), it can be seen that the vehicle increases in lateral position at the start of the simulation. This is expected due to the initial conditions provided. The vehicle needs to travel that extra distance in the lateral direction because the initial heading specifies that the vehicle is facing the edge of the lane rather than the centre of the lane. In order to correct itself for position and heading, the steering angle plot (figure 4.1b) shows that the vehicle has a maximum value of $-0.008 \text{ rad} = -0.458 \text{ deg}$ wheel angle = -8.7 deg steering wheel angle. This is quite a harsh steering value for a vehicle travelling at 20 m/s . The vehicle corrects itself immediately. Once again, this is due to the nature of the controller that is determined as a set of gains. Initially, the error in lateral position and relative heading is very high and as a result, the steering angle required to correct the vehicle is also high. As the velocity of the vehicle increases, and the simulation is run, the vehicle travels a larger amount in the lateral direction. This can be seen in figure (figure 4.1a) as the green line, representing the vehicle at 20 m/s , moves further than the blue line, representing the vehicle at 10 m/s . As the error builds, the controller calculates a higher steering angle in order to quickly correct the vehicle such that it goes back to the centre of the lane.

From the relative heading plot (top left in figure 4.2), it can be seen that the initial relative heading values (0 - 0.5 s) show that the vehicle is facing the right lane. However, the values go to 0 rad and the relative heading values then become negative. This means that, due to the controller, the vehicle rotates such that the vehicle is now going towards the centre of the lane. As a result, the lane is perceived as facing towards the right. These values settle to 0 rad as the vehicle reaches the centre of the lane indicating that the vehicle smoothly goes back to the centre of the lane and no oscillations occur. The lateral acceleration plot (bottom plot in figure 4.2) shows that the lateral acceleration experienced by the vehicle is approximately -0.4 m/s^2 . This is a relatively high value for lateral acceleration. However, this is expected as the steering angle value of the vehicle is also high. This is due to the nature of the controller and the need for the controller to correct for the initial high error values.

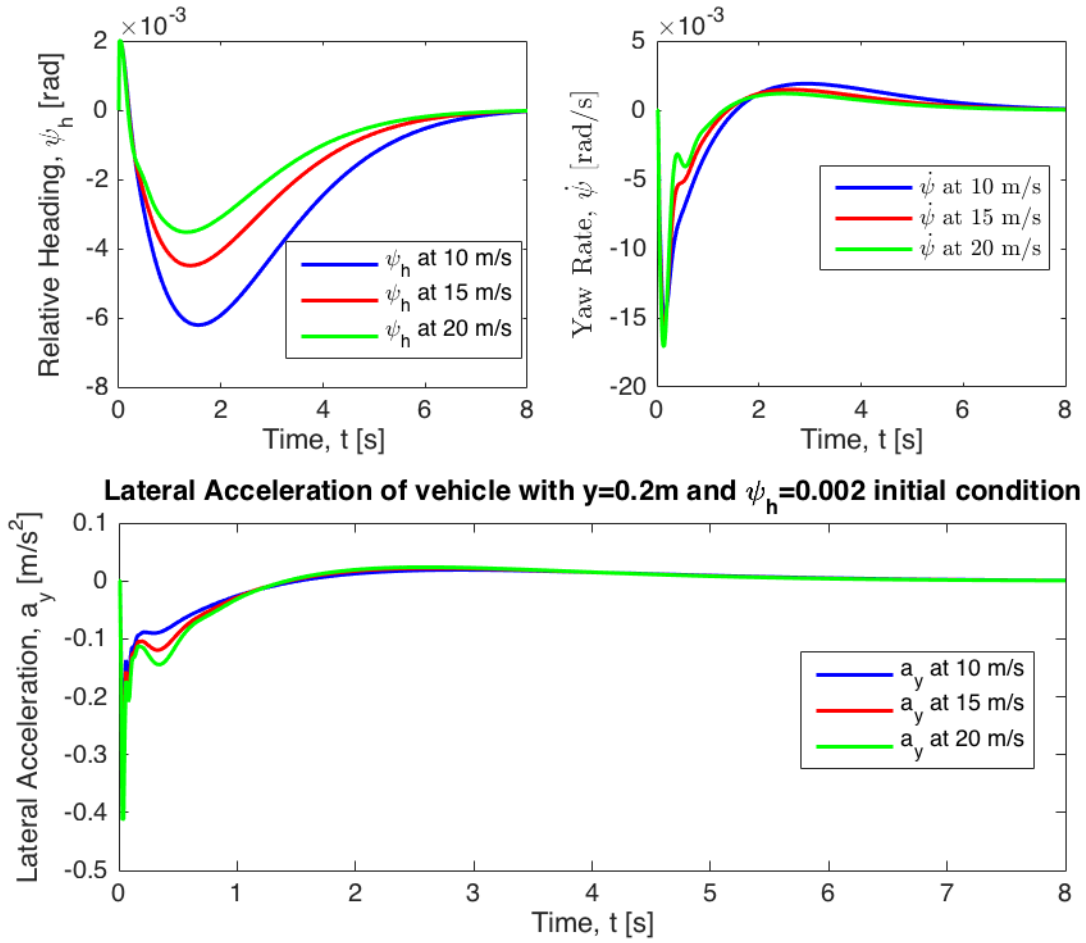


Figure 4.2: Relative Heading, Yaw Rate, and Lateral Acceleration Plots Combined Lateral Position and Relative Heading Simulation

From the above simulation, and the additional simulations presented in Appendix D, it can be seen that the controller steers the vehicle back to the centre of the lane, given the initial conditions. The simulations were run with constant gain values for all three velocities. It can be concluded that static gains are not ideal for the controller. The gains need to be speed dependent as large corrections are not desired at high speeds. As a result, the gains are made speed dependent so that at high speeds, the gains are lowered to avoid large corrections and the excessive steering angles achieved during the simulation. Large corrections could lead to significant changes in position and rotation which is dangerous and not desired. The gains at higher speeds are also tuned manually. In addition, it can also be concluded that the vehicle model created and simulated, along with the controller, can control the vehicle steering angle to keep a vehicle at the centre of the lane. The next step is to implement the controller in a test vehicle and test on a road with clear lane markings.

Chapter 5

Practical Implementation and Testing

The testing of the controller was done on a prototype research vehicle that belongs to one of the student teams (ATeam Eindhoven) at the university. The vehicle is a Toyota Prius, pictured below, currently being used as a platform for the research and development of automated and cooperative driving technologies.



Figure 5.1: Research Vehicle used for Testing Purposes

The vehicle is equipped with a MobilEye camera that is mounted to the windscreen. The camera has a CAN bus connection with a dSPACE real-time target machine that is onboard the vehicle. For the detailed CAN bus connections to and from the MobilEye camera, please refer to Appendix B. Via this CAN bus connection, signals from the camera can be imported to the real-time target machine where a Simulink model that contains the *I/O* connections, and signal processing and control algorithms runs in real time. The vehicle is also equipped with a MOVE-box which connects the

dSPACE real time target machine with the CAN bus of the car and controls the steering actuation of the vehicle. A steering angle value can be sent from the real time target machine (output of the controller, δ) and the MOVE-box actuates the steering by that value. After the controller is integrated into the Simulink scheme of the vehicle, the model is built by generating C code that is uploaded to the real-time target machine that is on-board the vehicle.

The testing was conducted on the A270-N270 highway between Eindhoven and Helmond. Necessary precautions were taken to ensure that the vehicle would not steer more than a maximum value causing the vehicle to steer out of the lane by placing saturation blocks on the output of the controller. The limits to the steering angle are speed dependent. It is also ensured that the controller is only activated when the lane confidence is greater than 2 on at least one of the lanes. Due to safety protocols implemented in the MOVE-box, while the controller is activated, the driver is able to override the system at any time by simply steering manually. If, while the controller is active, the driver determines that the situation is unsafe and needs to override, then he/she may takeover steering at any time to ensure their safety and the safety of the passengers. The controller would need to be reactivated manually.

Some changes were made from the simulation to when the controller was implemented in the vehicle. The output v_y was not used as the vehicle does not have a sensor implemented to measure it. Instead the derivative of the error in lateral position, Y_e , was taken as the error in lateral velocity for the damping term in the PD controller to control lateral position. This is obtained by taking the discrete derivative of the lateral position error, Y_e , and using that in the simulink scheme of the vehicle. The gains for the controller were made speed dependent as it was determined in the Simulations chapter that static gains were not desired. This was due to the excessive steering that was caused at higher velocities. While the steering angles were high, they were not excessive enough to be considered dangerous. It is desired that the controller provide a smooth steering angle to the vehicle. Therefore, the gains were lowered at high speeds so the corrections would be slower. This tradeoff was acceptable and considered safe. The gains were manually tuned once again at various longitudinal velocities to determine what the gain would be for a given range of velocities of the vehicle.

5.1 Test 1 - Straight Road

This section describes the test setup and results of the first test performed with the LKA controller implemented in the vehicle. In this test, the LKA controller is activated on a relatively straight road for approximately 85 seconds of testing where there are clear lane markings on both sides of the vehicle.

The figure below, (figure 5.2), shows the test conditions for first test that was conducted on a relatively straight road. The top graph (blue line) shows the velocity of the vehicle during the test. The lower graph (red line in figure 5.2) shows the curvature of the road as detected by the MobilEye camera. The velocity is varying because the throttling and braking of the vehicle was manually controlled by the driver and is also affected by the traffic on the road during the test. Even though the model created for the simulation is a constant velocity model, the velocity for the test was only slightly varying. It is not possible to keep constant velocity on the road due to traffic conditions, and due to the vehicle being driven manually during the test. The curvature data shows values less than $0.5 * 10^{-4} m^{-1}$, which means the radius of the road is larger than 10000 m which would mean that the road is relatively straight, but not arrow straight. During this time it is expected that the vehicle will not steer too much as the road is relatively straight. The only steering required would be to correct the vehicle's lateral position around the centre of the lane.

The figure below,(figure 5.3), shows the lateral position and the **steering wheel** angle during

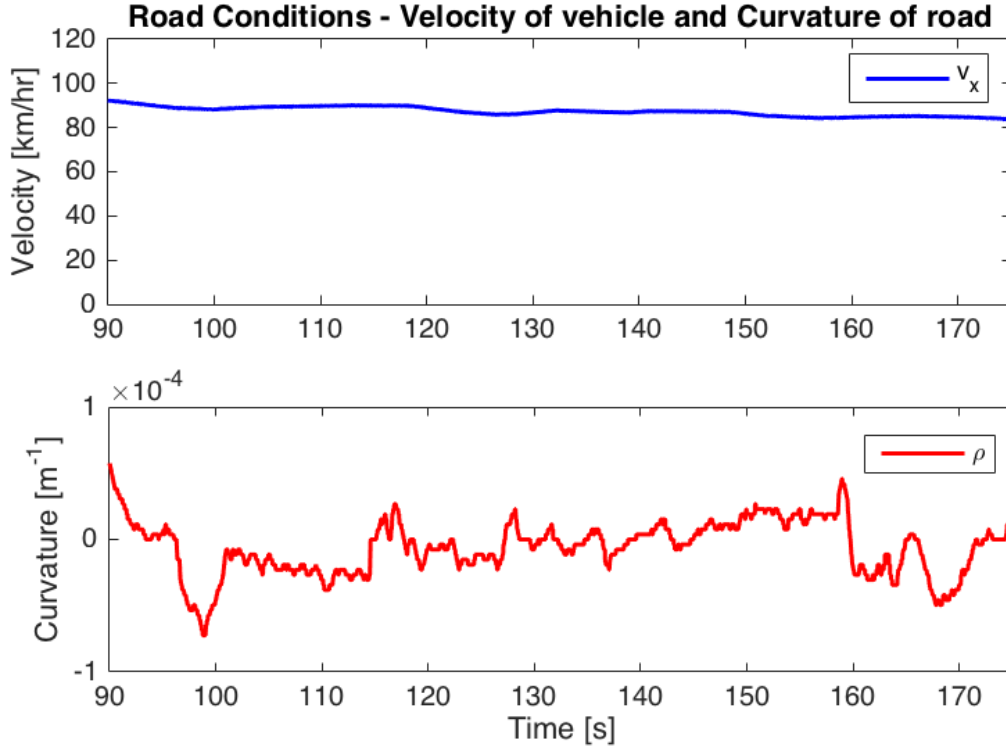


Figure 5.2: Test 1 - Test Conditions

this test. The lateral position graph (top plot in figure 5.3) shows the vehicle’s lateral position as measured with the camera, and the vehicle’s desired lateral position. In addition the plot shows the measured distance to the right and left lanes. The plot shows that the vehicle is a maximum of 0.2 m away from the centre of the lane for most of the test. The vehicle is still within the heading band as described in section 3.1. Within this region the desired relative heading $\psi_{h,des}$ is 0 rad . Since the lane is relatively straight, it is expected that the vehicle is only correcting for position. This can be seen over the course of the test data as the lateral position y (blue line) goes closer to the desired lateral position y_{des} (dashed red line). However, due to the fact that the lane is not arrow straight, it can be seen that the vehicle does not always remain at the centre of the lane. This is expected due to the changing nature of the road over the course of the test run. However, this result is acceptable as the vehicle never drifts beyond 0.2 m in the lateral direction. This is considered acceptable as the vehicle remains close to the centre of the lane and because this mimics the driving style of a human being.

The bottom plot in figure 5.3 represents the steering wheel angle of the vehicle and the steering wheel angle sent by the controller to the vehicle. The red line represents the steering angle as sent by the controller to the vehicle. The blue dashed line represents the steering angle measured by the vehicle steering wheel angle sensor. This is the raw measurement from the sensor. The δ from the controller remains around 0 rad as the vehicle’s lateral position is near the centre of the lane. However, due to the change in the lane conditions (small changes in curvature), the vehicle starts to drift away from the centre and the vehicle is required to steer again. This is due to the error in lateral position that builds up and the controller responds to that. These steering angle values are very small which ensures the car does not make sudden movements in the lateral direction, potentially causing an accident. The steering wheel angle goes to a maximum of 0.05 rad so the vehicle was required to move the steering wheel by around 3 deg . This is a small change in steering wheel angle required to correct the vehicle’s lateral position. At approximately 90 km/h , this steering correction

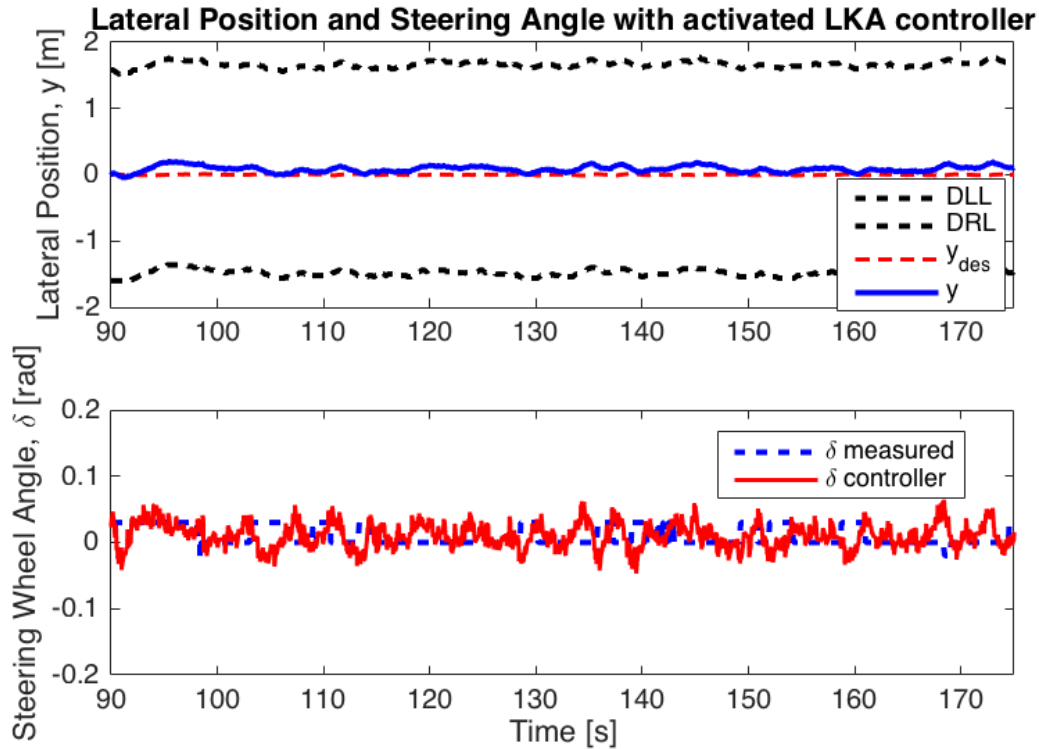


Figure 5.3: Test 1 - Lateral Position and Steering Angle Results

is enough to laterally correct the vehicle's position. The steering angle appears to be bumpy because a new steering angle value is calculated every time step (0.01 s). However, it is not possible for the actuator of the steering angle to steer the vehicle at that frequency.

From the performed test, it can be concluded that the vehicle does not deviate away from the centre of the lane. However, due to the constantly changing lane conditions, there is an error of approximately 0.2 m in the lateral direction. This is acceptable as it is close to a human driving style. The steering angle corrections are not high and are considered realistic. At 90 km/h a steering wheel angle of approximately 3 deg is performed to correct the vehicle's position. There appears to be a difference between the sensor reading and the controller output for the steering wheel angle. The calculations for the controller are taking place at 100 Hz because that is the frequency of the real-time target machine. However, the steering actuator and the steering wheel angle sensor does not work at that frequency. Therefore, these differences are seen. However, small changes of $3\text{-}4\text{ deg}$ is performed by the steering wheel actuator and can be measured. In addition, it was not possible to keep the velocity constant as the testing was conducted on roads with traffic. Also, the vehicle was driven manually. Despite this, the controller was able to calculate a steering angle value to send to the vehicle.

With the LKA controller working on relatively straight road conditions, it can be tested on curved regions of the highway.

5.2 Test 2 - Curved Road

This section describes the conditions and test results of the second test performed with the LKA controller implemented in the vehicle. In this test, the LKA controller is activated on a curved

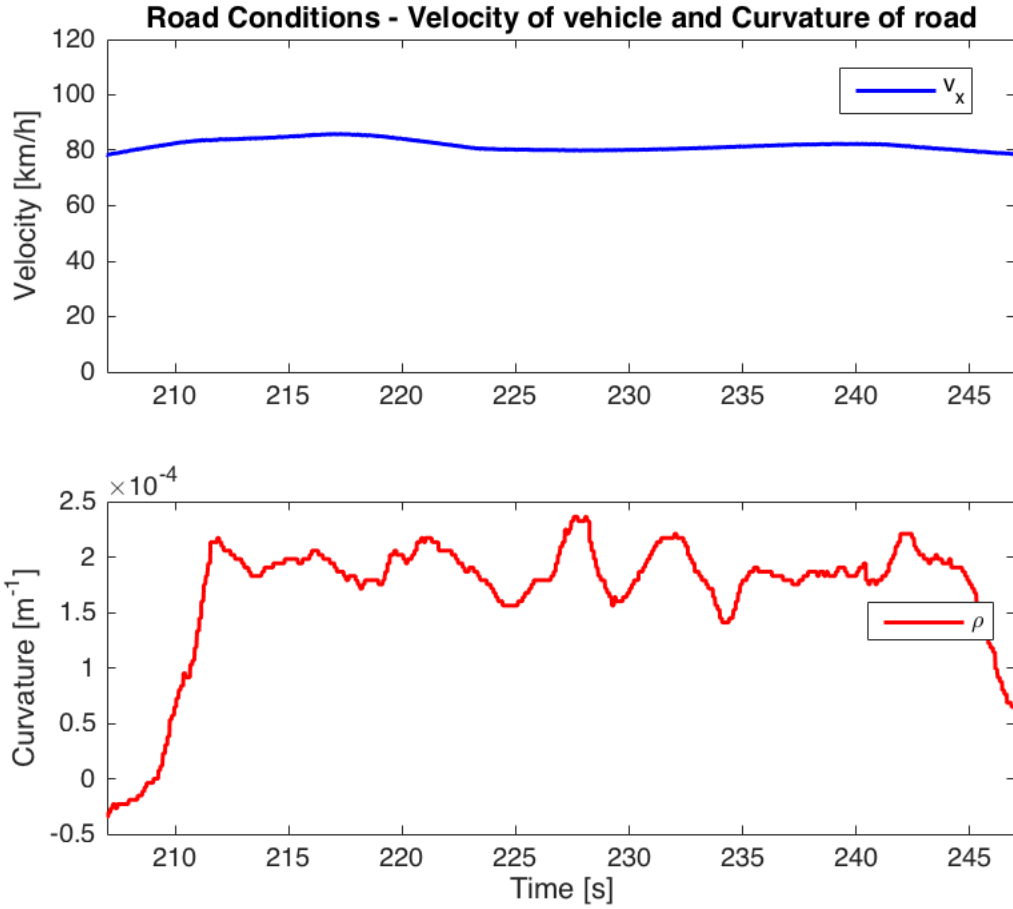


Figure 5.4: Test 2 - Test Conditions

section of the road.

The figure above, (figure 5.4), shows the test conditions during the second test. The velocity of the vehicle (top plot in figure 5.4) remains around 80 km/h . It is kept as constant as possible. However, due to the changing traffic conditions, the velocity of the vehicle had to be adapted. The curvature values (bottom plot in figure 5.4) show that the curvature of the road is around $2 \times 10^{-4} \text{ m}^{-1}$, which corresponds to a radius of 2500 m . The positive curvature values indicate that the MobilEye detects a curve in the left direction with respect to the vehicle. Positive curvature values mean that the vehicle will be required to steer left and negative curvature values mean that the vehicle will be required to steer to the right

While this might only be a slight curve, it is significant on a highway. It is important for the controller to steer the vehicle along the centreline of the road. As the curvature increases, $\dot{\psi}_{des}$ increases in magnitude creating a desired yaw rate for the vehicle to perform. This adds rotational motion to the vehicle that causes the vehicle to steer. In addition the lateral reference point, y_{des} , begins to curve as it is defined by the road equation (*see* Section 1.1) which includes a curvature term. The positive values of curvature move the y_{des} values more to the left with respect to the vehicle's position and as a result an error in lateral position is also created. This is then added to

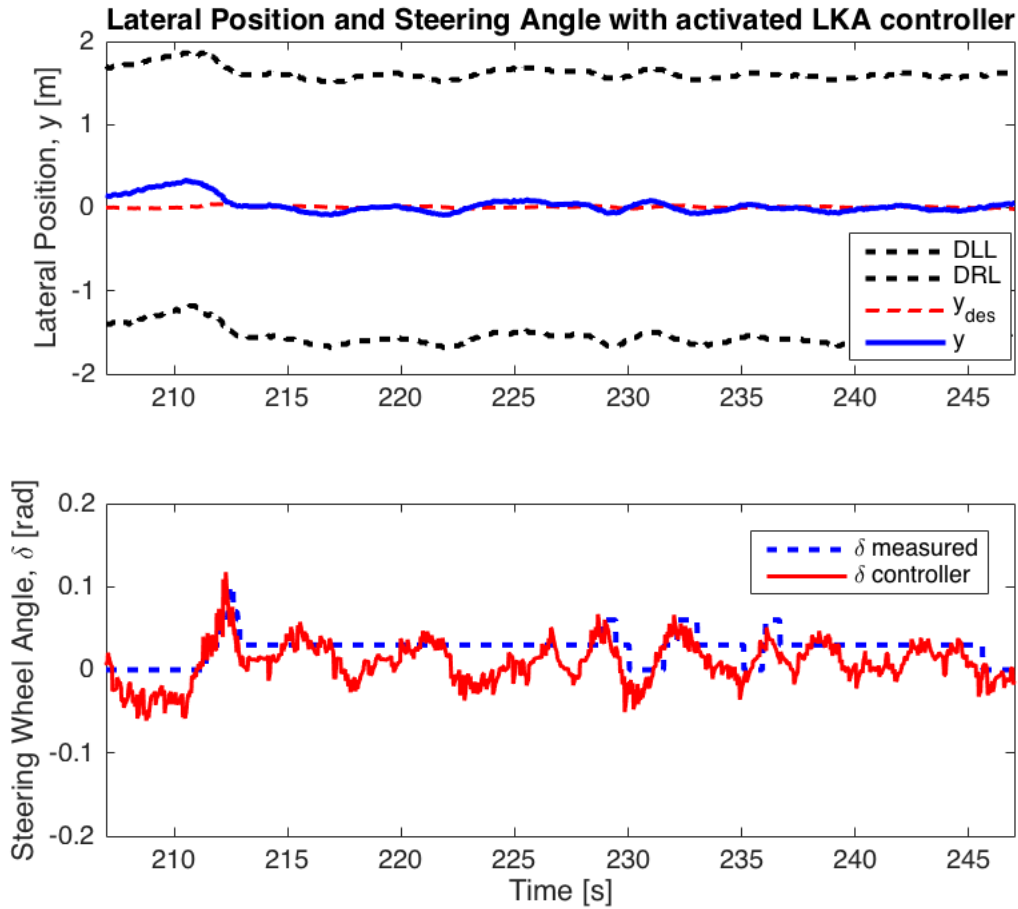


Figure 5.5: Test 2 - Lateral Position and Steering Angle Results

the $\dot{\psi}$ to create the total lateral motion required to navigate a curved road.

The figure above, (figure 5.5), shows the lateral position and **steering wheel** angle results from the second test, performed on a curved road. As the vehicle approaches the corner at the start of the figure (before the 210 s mark), it can be seen that the distance to the left lane starts to increase and the distance to right lane starts to decrease. As the left turn begins, the vehicle takes approximately a second to detect the curve and begin correcting for it. However, since the vehicle is already towards the left of the centreline of the lane, the error in lateral position goes to 0 m. The road is still curving and the desired lateral position is constantly moving left as the camera detects the lanes moving in that direction. This requires a high steering wheel angle as to correct for the corner takes place at the 212 second mark. The steering angle goes to a peak value of 0.1 rad and then decreases to between around 0.05 rad. Looking at the lateral position of the vehicle in the top graph in the figure above, it can be seen that the vehicle turns along the curve and also remains at the centre of the lane. The steering wheel angle plot (bottom plot in figure 5.5) shows that the vehicle steers and the measured steering wheel angle is a constant value above 0 rad for most of the curved region of the road. However, the controller data (red line in the steering wheel angle plot in figure 5.5) shows changes in the steering wheel angle that is required of the vehicle. The gradual nature of the curved road proved a suitable test region for the LKA controller and the vehicle does correct for both lateral and rota-

tion error resulting in a smooth curve with the vehicle remaining very close to the centre of the lane.

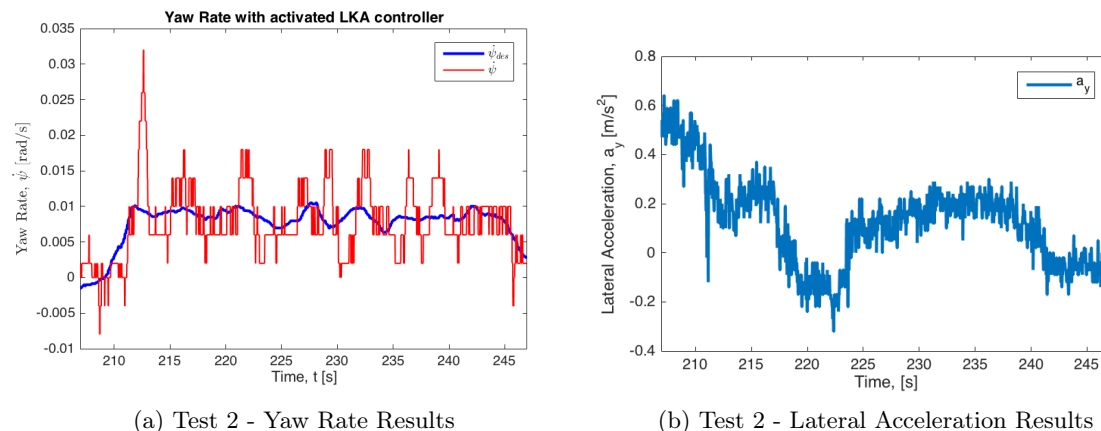


Figure 5.6: Test 2 - Yaw Rate and Lateral Acceleration Measurements

The figure above, (figure 5.6), shows the yaw rate and the lateral acceleration plots during the curved road. Since steady state cornering is assumed, the desired yaw rate, $\dot{\psi}_{des}$ (blue line in figure 5.6a), stays around 0.01 rad/s . There are small changes and that is expected as the detected curvature values are also changing. However, assuming steady state cornering provides a suitable estimate of the yaw rate of the vehicle. This is visible in the yaw rate plot (figure 5.6a) as the desired yaw rate and the measured yaw rate are around the same value (excluding sensor noise). The measured lateral acceleration, shown in figure 5.6b, shows that a maximum lateral acceleration is only measured at the beginning when the steering is maximum due to the detected curvature in the road. Ignoring the sensor noise, the maximum lateral acceleration achieved during cornering reaches 0.6 m/s^2 . This is acceptable as there is no jerk in the lateral direction providing smooth cornering.

The cornering test results show that the vehicle is capable of taking a corner with steady state yaw rate and lateral position. The vehicle does take the corner and maintains its position in the centre of the lane. There is a steady steering input to the vehicle through the corner which means the vehicle is steering through the corner. This test shows that the controller is capable of taking corners on the highway.

5.3 Test 3 - Combined Curved and Straight Road

This section describes the conditions and test results of the third test performed with the LKA controller implemented in the vehicle. In this test, the LKA controller is active as the vehicle is at the end of a curved region of the road and is approaching a straight section.

The figure above, (figure 5.7), shows the velocity (top plot) and curvature (bottom plot) values during the third test. The curvature values show that the values are negative (between 120 and 140 seconds) indicating a right turn, and then the curvature values go to 0 m^{-1} indicating that the road straightens as recorded directly from the MobilEye camera.

The test results from the third test are presented in figure 5.8. The lateral position, y and **steering wheel** angle, δ are presented. From the top plot in figure 5.8, the desired lateral position (red line) is shown, along with the measured position (blue line) and the distance to both the lanes (black lines). It can be seen that the vehicle begins at approximately 0.3 m away from the centre of the lane while still on the curved region. This is evident as there is a steering wheel angle of approximately

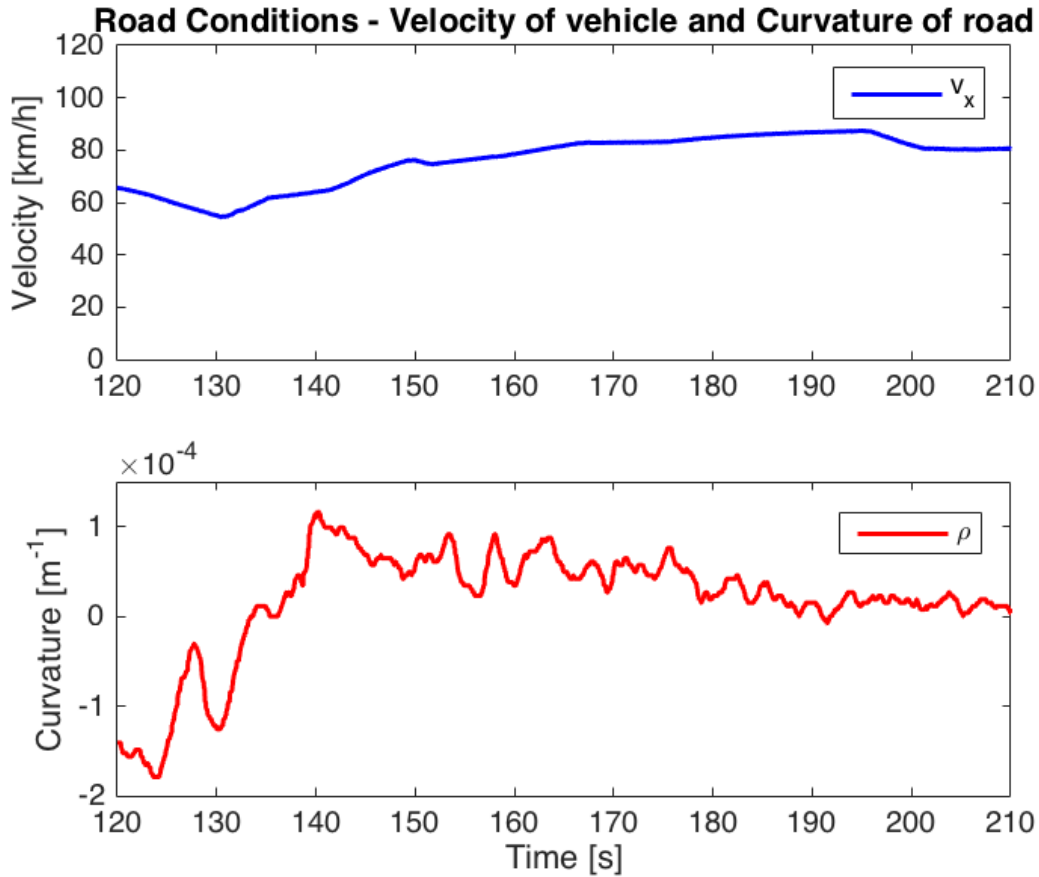


Figure 5.7: Test 3 - Test Conditions

0.1 *rad*. The vehicle then detects the curvature going to $0m^{-1}$ (between 130 - 140 s). The vehicle then begins correcting for this change by setting the steering wheel angle to 0 *rad*. As the road is no longer a corner, the vehicle begins correcting lateral position to go towards the centre of the lane. As the vehicle corrects its position, the road continues curving very slightly (140 s to 160 s). During this time the vehicle the moves across the lane to approximately 0.1 m away from the centre of the lane towards the opposite lane. In this period, the curvature values are shown to be positive. This is because the vehicle has crossed the centre of the lane and is oriented in such a way that the lane appears to be curved to the left. The road then begins to straighten and the lateral position of the vehicle goes to the centre of the lane. During this time the steering angle does not increase beyond 0.05 *rad* to correct for the curve in the road. The steering angle then goes to 0 *rad*. The lateral position of the vehicle is also approximately 0 m so no corrections are required and the vehicle maintains its position in the lane.

The figure below (figure 5.9) shows the yaw rate plot during this test. At the beginning it can be seen that there is a desired yaw rate as the vehicle is at the end of a curved road. However, as the camera begins detecting the straightening of the road, the desired yaw rate goes to 0 *rad/s*. This is ideal as no rotation is required on a straight road. The measured yaw rate also follows the same set of values, as seen by the red line in figure 5.9.

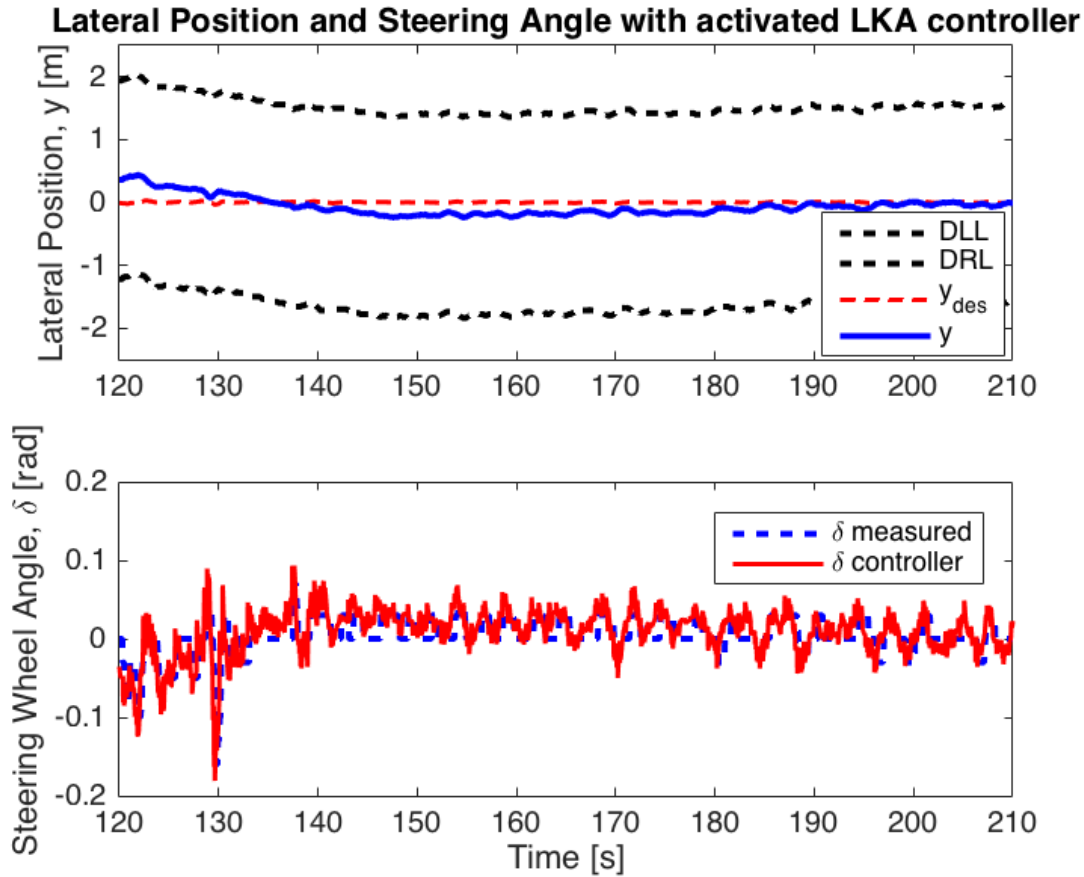


Figure 5.8: Test 3 - Lateral Position and Steering Angle Results

The third test performed showed that the vehicle was able to correct its position as a curved road straightens out without oscillations in lateral position and a relatively low steering wheel angle. This is similar to a human driver and is therefore considered acceptable.

From the simulation results we see the ideal situation of a vehicle. It is possible for the steering angle to go to 0 as the lateral position goes to 0. However, in real test cases, this is not always the case as roads are not often arrow straight. Even on relatively straight roads, the steering angle values are changing because the road curvature is changing slightly. This creates the discrepancy between the simulation results and the test results. Furthermore, since the tests were conducted on the highway, various traffic situations needed to be accounted for. For example, a constant speed could not be maintained as there were other vehicles around. It is not possible to keep constant velocity on the road due to traffic conditions, and due to the vehicle being driven manually during the test. Therefore there is a difference between the simulation setup and the test setup. However, between two samples, the velocity is not changing. Therefore, for each sample, the controller was working with one velocity value. The steering angle plots from the controller appear to be bumpy because a new steering angle value is calculated every time step (0.01 s). However, it is not possible for the actuator of the steering angle to steer the vehicle at that frequency. Furthermore, the linear model was considered in this project. This model assumes low slip and low steering angles which limits the application of the model to highway scenarios. However, the linearised vehicle dynamics and the vision model work well as a starting point as small angles are often sufficient for highway

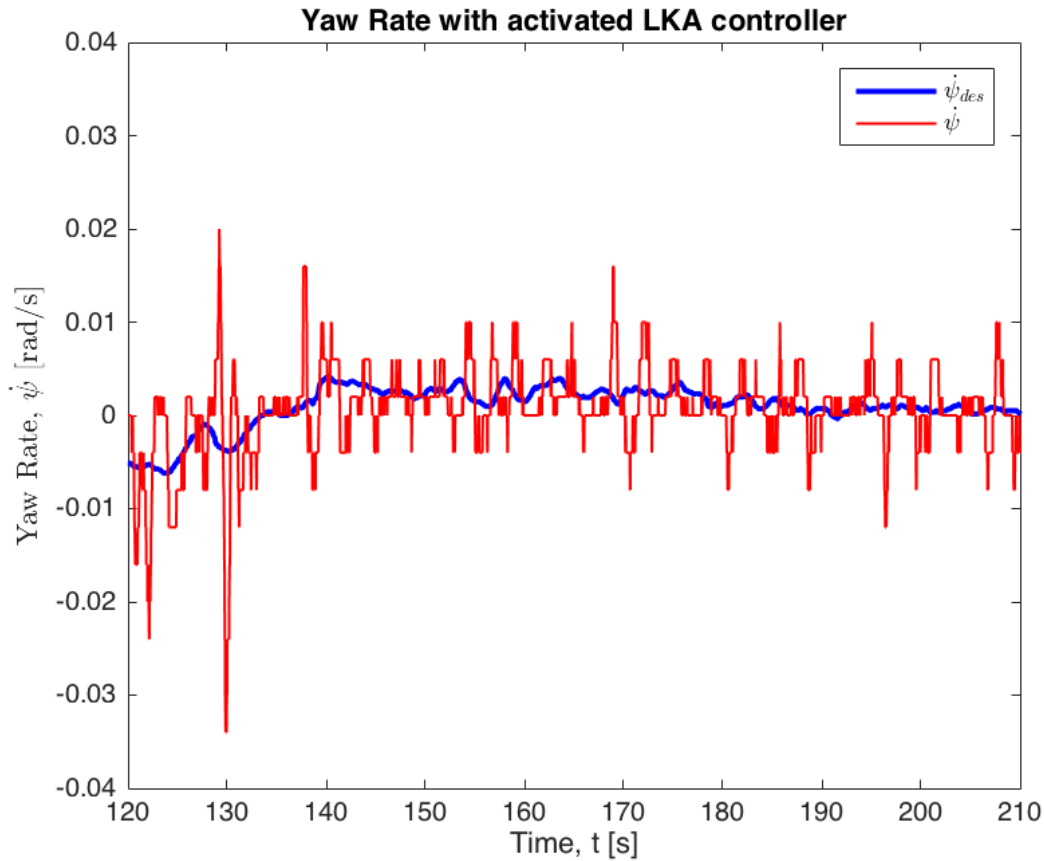


Figure 5.9: Test 3 - Lateral Position and Steering Angle Results

driving and the MobilEye works well during these small changes.

There were some changes to the controller as the transition was made from simulation to testing. The gains utilised for the simulation were updated to be speed and curvature dependent. At high speeds the gains were lowered as too many corrections is not desired. This would cause the vehicle to oscillate within the lane. This could potentially cause the vehicle to exit the lane onto the next lane, or perhaps off the road. This is not desired and could lead to an accident. Therefore, it was important that the corrections at high speeds be small. In addition the vision model was used as works with small changes in curvature. This scenario proved best for the performance of the MobilEye camera as it works at 10 Hz. This means that the MobilEye camera gives a new value every 0.1 seconds. The frequency of the MobilEye camera is a limitation to the performance of the system and could explain why the vehicle has some difficulty in taking sharper (high curvature) corners. Therefore, the vision model is an accurate model of the MobilEye camera.

Chapter 6

Discussion and Conclusion

The controller worked on the highway scenarios smoothly without sudden jerks or oscillations. The highest steering angle is achieved the moment after the controller is activated as the initial conditions of the vehicle are random and could be anything when driving in real life. There will be a build up of error in lateral position and relative heading and the moment the controller is activated, the vehicle will begin correcting itself. This would cause the sudden steering. This is acceptable as the gains are made speed dependent and also due to hardware limitations. The highway scenario is the optimal scenario for getting the most reliable data from the MobilEye camera. During the several tests conducted on the A270-N270 highway, the vehicle was always between 70 km/h and 100 km/h , and the controller did not fail to correct the position of the vehicle in the lane. When encountering curved roads, the controller was able to steer the vehicle without letting the vehicle exit the lane. At no point did the vehicle get beyond 0.3 m away from the centre of the lane. The highway scenario included one traffic light intersection. There are lane markings through the intersection that determine which lane the vehicle should follow. However, it was noticed that the lane confidence of the lane would drop to 0 during that period. This was a temporary drop in distance to lane readings and was accounted for in the controller with the implementation of the virtual lane and the vehicle was able to navigate the traffic light intersection without any problems.

As an extra test, the same controller that was used during the highway tests was activated while the vehicle was in an urban environment. The urban environment included sharper corners with radius of between 1000 m - 1250 m . This appeared to be a problem for the camera as it could not keep up with the rapid changes in curvature. Despite the fact that there were clearly visible lane markings on either end of the lane, the camera would not keep up with the change in curvature because the camera only gives one value of curvature every 0.1 seconds. During this situation, the controller calculated a steering angle based on this curvature value and would hold that value till the next curvature value was received. This could cause the vehicle to steer for the corner, then hold that steering angle value until the next set of values were received, and then make a correction again. This proved too slow and the vehicle would move in the lateral direction closer to the outside edge of the lane by a significant amount. This is not desired as it could potentially lead to an accident. During every test performed for this corner, a manual override needed to be done because the curvature values were not updating quickly enough. This is also expected as the vision model that was used required small linear changes in curvature.

There were situations where the camera would detect the side curb as a lane. While the confidence would switch from 1 to 2 (out of a maximum of 3), the distance to the lane also varied significantly. During certain tests the distance to the curb would be measured as 0.86 m , or 2.16 m , and was not consistent. Therefore, the virtual lane was implemented and utilised in this situation. However, during instances where the confidence was 2, the controller would use the actual distance to the lane value and because that was 2 m or 0.8 m , depending on the situation, and the car would

suddenly jerk in the lateral direction. This is accounted for again with the implementation of the virtual lane. During these periods, the virtual lane of 1.6 m was used to allow the controller to be active when only one proper lane is detected.

Based on the tests performed with the MobilEye camera and the designed controller it is recommended that the camera with this controller be used only on highway scenarios where clear lane markings are visible at all times and where there are not significantly high curvatures of the road. The bicycle model is a good representation of the dynamics of the vehicle. However, it is still a linearised model and the vehicle does have nonlinear behaviour in the lateral direction. This will need to be accounted for in the future. In addition, the vision model provides an accurate representation of the camera dynamics to be modelled. Despite having limitations such as slow changes in curvature, the model provides an accurate representation of a single MobilEye camera.

6.1 Future Work

Many potential problems while lane keeping using the MobilEye camera were overcome with the designed controller, as can be seen in the test results. However, there is still potential for further improvements and changes.

One of the important things to consider is the nonlinear behaviour of the vehicle in the lateral direction. This would improve the automated functionality of the vehicle as more complicated lateral manoeuvres could be performed by the vehicle. In addition, a possible combination of sensor information could be added to the data from the MobilEye that could assist with better lane information. For example, Maps data could provide the desired route that could help with where the vehicle needs to be in terms lateral and longitudinal position. The problem with this is that accurate map information is required. An alternative is that sensors could be mounted on the vehicle on the front corners to detect possible obstacles or possible ends of the road (curbs, sidewalk, etc.) to determine the limits of the road in the lateral direction. This, added to the MobilEye information, would improve the ability of the vehicle to drive on roads where lanes are not clear or where there are no lanes on the road. This would benefit the overall autonomous functionality of the vehicle.

If the MobilEye is used for LKA as the only sensor, then a few improvements could be made to the data from the MobilEye camera before it goes to the controller. The sudden switch between distance to the lane measurements when the camera loses one of the lane is not ideal so a filtering of the distance to the lane measurements would need to be done. In addition, an estimator could be implemented to estimate the distance to the lane, if the MobilEye camera is not confident of the current distance to the lane measurement. However, this should only be applied to situations where the MobilEye camera is confident of one of the lanes but not of the other. This would benefit the performance of the controller as no sudden steering would be done and a smoother drive would be possible.

6.2 Conclusion

A linearised vehicle dynamics model was created to simulate a vehicle. A vision model was added to the vehicle dynamics model to simulate a vehicle that is equipped with a vision sensor, like the MobilEye camera. An LKA controller was designed, simulated, and tested using information only from the MobilEye camera. It can be concluded that the camera provides good, reliable information only on highways where it can be used as the primary sensor with the designed controller. The designed controller provides a smooth ride on the highway. However, in order to be used in urban environments, where changes in curvature are high and visible lane markings are not guaranteed, several recommendations have been made to adapt the model and the controller. The reliability of

the controller is directly determined by the ability of the sensors to provide accurate information. This provides an opportunity to add multiple sensors to the vehicle and using sensor fusion techniques to provide reliable information. It is important for both the sensor and the controller to be reliable so that safe Lane Keeping Assist technology can be implemented in vehicles for safe automated driving.

Bibliography

- [1] J. H. H. M. Alleleijn. Lateral string stability of automated vehicle platoons. *Technische Universiteit Eindhoven*, 2015.
- [2] B. van Arem, C.J.G. van Driel, and V. Ruben. The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Transactions on Intelligent Transportation Systems*, 7(4):429–436, 2006.
- [3] I. J. M. Besselink. Vehicle Dynamics 4AT000 Lecture Notes. *Technische Universiteit Eindhoven*, 2015.
- [4] European Commission. Towards a European road safety area: policy orientations on road safety 2011-2020. *Citizens' Summary: Road Safety Guidelines*, 2010(903), 2010.
- [5] A. Eskandarian. *Handbook of Intelligent Vehicles*, volume 2. Springer-Verlag London Ltd, Washington, DC, USA, 2012.
- [6] J. Hsu and M. Tomizuka. Analyses of Vision-based Lateral Control for Automated Highway System. *Vehicle System Dynamics*, 30(5):345–373, 1998.
- [7] SAE International. Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems STANDARD J3016-201609. *SAE Mobilus*, 2014.
- [8] MobilEye Technologies Limited. C2-270 Technical Installation Guide. *MobilEye Technologies Limited*, May 2012.
- [9] C. Liu and R. Subramanian. Factors Related to Fatal Single-Vehicle Run-Off-Road Crashes. Technical Report DOT HS 811 232, United States Department of Transportation, 2009.
- [10] World Health Organization. *Global Status Report on Road Safety 2013: Supporting a Decade of Action*. World Health Organization Geneva (Switzerland) 2013.
- [11] R. Rajamani. *Vehicle Dynamics and Control*. Mechanical Engineering Series. Springer US, 2005.
- [12] S. Solyom, A. Idelchi, and B. B. Salamah. Lateral control of vehicle platoons. pages 4561–4565, Oct 2013.
- [13] S. Yaron. MobilEye C2-270 & ME5 Standard CAN Protocol & AWS Extended Log Data Protocol 2. *MobilEye Technologies Limited*, 2012.

Appendix A

Derivation of Lateral Vehicle Model

In this section of the report, the state space model of the lateral vehicle model is derived. It is divided into the lateral dynamics model and the camera mode. The figure below shows the lateral vehicle model. The two front tyres are represented as one front tyre which generates a total lateral force F_{y1} . The same is done for the two rear tyres which generates a lateral force F_{y2} . The adapted SAE coordinate system is adopted to model and simulate the lateral dynamics of the vehicle.

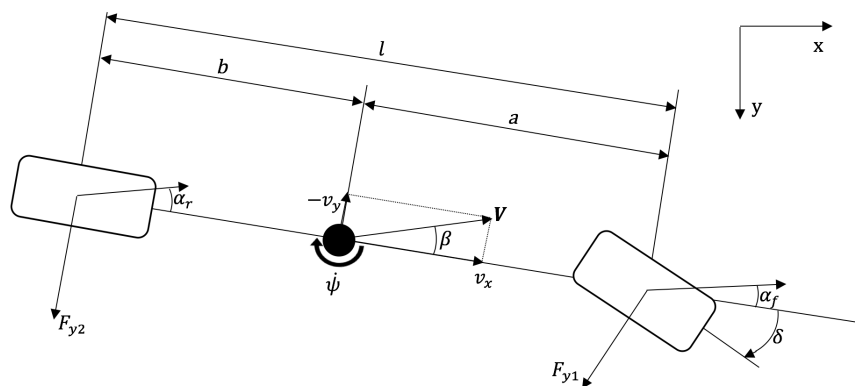


Figure A.1: Linear Vehicle Model

From figure A.1 above, two of the main equations of motion are considered, the sum of forces in the y-direction ($F_{y,i}$) and the moments in the z-direction, about the centre of gravity (M_z) and can be written as

$$\Sigma F_{y,i} = ma_y = m(\dot{v}_y + v_x\dot{\psi}) = F_{y1} + F_{y2} \quad (\text{A.1})$$

$$\Sigma M_z = I_{zz}\ddot{\psi} = aF_{y1} - bF_{y2} \quad (\text{A.2})$$

where m is the mass of the vehicle, a_y is the lateral acceleration caused by the lateral motion (v_y) and rotational motion ($\dot{\psi}$), v_x is the longitudinal velocity, I_{zz} is the vehicle yaw moment of inertia about the vertical (z-) axis of the vehicle, and $\ddot{\psi}$ is the yaw acceleration.

In addition to the vehicle dynamics model, to make the model represent the actual vehicle, a vision-based and curvature-based model is added from [6] and two more states are added to the system to represent the vehicle's position and orientation in the lane determined from the camera sensor. This is shown in figure A.2 below.

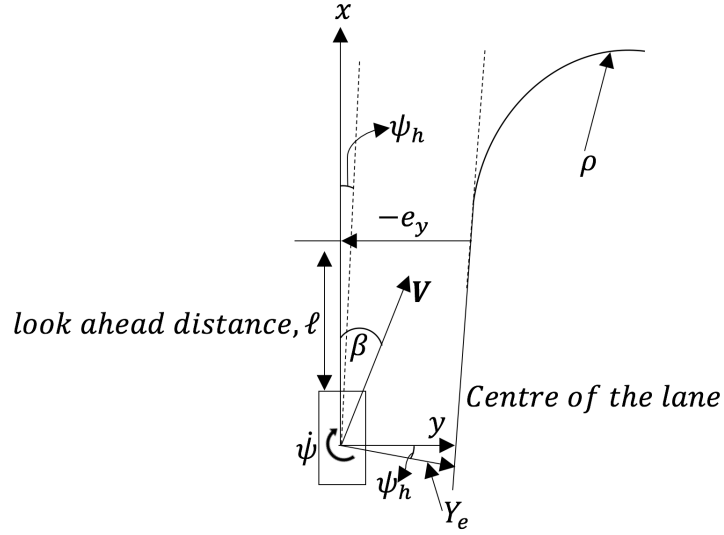


Figure A.2: Vision Model

The first is the lateral position of the vehicle in the lane with respect to the lanes, and the relative heading between the vehicle and the lane. These parameters can be measured using the MobilEye camera. Therefore, two more equations are added to (A.1) and (A.2) to make the simulation closer to what the real vehicle would experience.

$$v_y = \beta v_x + \psi_h v_x \quad (\text{A.3})$$

$$\dot{\psi}_h = \dot{\psi} - v_x \rho \quad (\text{A.4})$$

where β is the vehicle side slip angle about the centre of gravity, v_y is the velocity in the lateral direction, ψ_h is the relative heading, and ρ is the curvature of the road.

The general state space model is used to represent the dynamics of the vehicle and can be written in the following form

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} &= \mathbf{Cx} + \mathbf{Du} \end{aligned} \quad (\text{A.5})$$

The following state variables are chosen $[\dot{\psi} \quad \beta \quad \psi_h \quad y]^T$ and inputs as $[\delta \quad \rho]^T$ which means equations (A.1), (A.2), (A.3), and (A.4) need to be rewritten in terms of the state variables. The lateral forces F_1 and F_2 are a function of the wheel side slip angle, α_i , where i is the front or rear wheel. This represents the cornering characteristics for linear vehicle behaviour.

$$F_1 = c_f \alpha_1 \quad (\text{A.6})$$

$$F_2 = c_r \alpha_2 \quad (\text{A.7})$$

The tyre side slip angle can be written as

$$\alpha_1 = \delta - \beta - \frac{a\dot{\psi}}{v_x} \quad (\text{A.8})$$

$$\alpha_2 = -\beta + \frac{b\dot{\psi}}{v_x} \quad (\text{A.9})$$

The above equations can be substituted in (A.1) and (A.2)

$$m(\dot{\beta}v_x + v_x\dot{\psi}) = c_f\delta - c_f\beta - \frac{ac_f\dot{\psi}}{v_x} - c_r\beta + \frac{bc_r\dot{\psi}}{v_x} \quad (\text{A.10})$$

$$I\ddot{\psi} = ac_f\delta - ac_f\beta - \frac{a^2c_f\dot{\psi}}{v_x} + bc_r\beta - \frac{b^2c_r\dot{\psi}}{v_x} \quad (\text{A.11})$$

Rewriting A.10 and A.11 such that $\ddot{\psi}$ and $\dot{\beta}$ is on the left hand side of the equation, and the other variables β , $\dot{\psi}$, and δ are on the right hand side of the equation gives the following equations

$$\ddot{\psi} = \delta\left(\frac{ac_f}{I}\right) + \dot{\psi}\left(\frac{-a^2c_f - b^2c_r}{Iv_x}\right) + \beta\left(\frac{-ac_f + bc_r}{I}\right) \quad (\text{A.12})$$

$$\dot{\beta} = \delta\left(\frac{c_f}{mv_x}\right) + \dot{\psi}\left(\frac{-ac_f + bc_r - mv_x^2}{mv_x^2}\right) + \beta\left(\frac{-c_f - c_r}{m}\right) \quad (\text{A.13})$$

$$\dot{\psi}_h = \dot{\psi} - v_x\rho \quad (\text{A.14})$$

$$v_y = v_x\beta + v_x\psi_h \quad (\text{A.15})$$

The equations A.12, A.13, A.14, and A.15 can be written in the general state space form

$$\begin{bmatrix} \ddot{\psi} \\ \dot{\beta} \\ \dot{\psi}_h \\ v_y \end{bmatrix} = \begin{bmatrix} \frac{-a^2c_f - b^2c_r}{Iv_x} & \frac{-ac_f + bc_r}{I} & 0 & 0 \\ \frac{-ac_f + bc_r - mv_x^2}{mv_x^2} & \frac{-c_f - c_r}{mv_x} & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & v_x & v_x & 0 \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \beta \\ \psi_h \\ y \end{bmatrix} + \begin{bmatrix} \frac{ac_f}{I} & 0 \\ \frac{c_f}{mv_x} & 0 \\ 0 & -v_x \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \rho \end{bmatrix} \quad (\text{A.16})$$

The output state matrix is determined based on what measurements are possible. Since the state variable β can not be measured, the output a_y will be used. The measurements of the other state variables $\dot{\psi}$, ψ_h , and y can be measured using sensors.

$$a_y = \dot{v}_y + \dot{\psi}v_x = \dot{\beta}v_x + \dot{\psi}v_x \quad (\text{A.17})$$

Substituting the equation for $\dot{\beta}$ from A.13, and writing in the state space form gives

$$\begin{bmatrix} \dot{\psi} \\ a_y \\ \psi_h \\ y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{-ac_f + bc_r}{mv_x} & \frac{-c_f - c_r + ma_x}{m} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \beta \\ \psi_h \\ y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{c_f}{m} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \rho \end{bmatrix} \quad (\text{A.18})$$

This state space model is used as the vehicle model for the simulation and determining the controller in chapters 2 and 3.

Appendix B

MobilEye Connections

The figure below show the connections of the MobilEye camera module to the MobilEye processing unit, and eventually to the vehicle. This figure is obtained from the MobilEye installation guide.

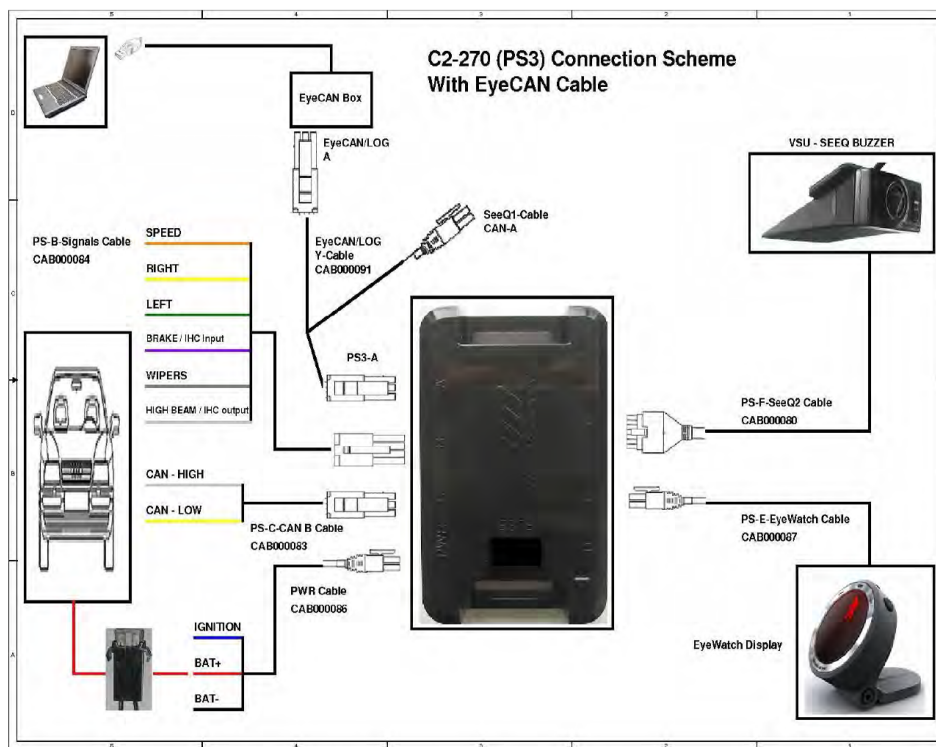


Figure B.1: MobilEye Connection Scheme

Appendix C

Model Verification

The model created in MATLAB/Simulink is tested by placing known inputs of δ and ρ to check the response of the model. A constant velocity of 20 m/s is chosen at the start of each simulation and maintained throughout the simulation because that is the expected average velocity of the vehicle while the controller is active.

First, a δ of 0 rad and ρ of 0 m^{-1} given to the model. This simulates the vehicle travelling

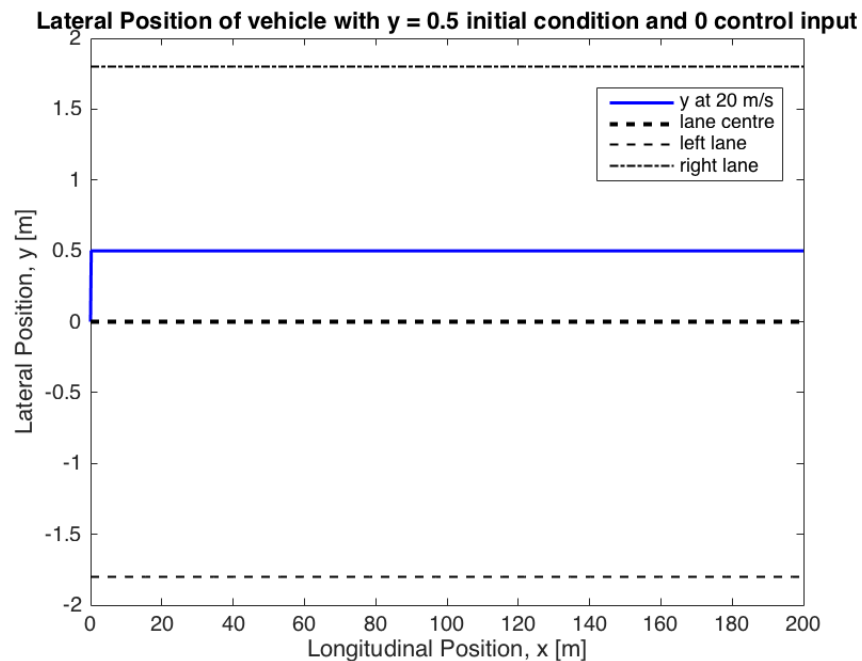


Figure C.1: Zero Control Input for Model Verification

at 20 m/s on a straight road with no steering at all. The vehicle state initial conditions are $[\dot{\psi} = 0 \quad \beta = 0 \quad \psi_h = 0 \quad y = 0.5]^T$.

The vehicle is 0.5 m away from the centre of the lane (closer to the right lane) and is oriented along the same direction as the lane. It is expected that the vehicle will remain at this lateral position in the lane and not move in the lateral direction, nor will it rotate about its vertical axis.

This can be seen in figure C.1 above. It can be seen that the vehicle remains at a constant lateral position for the 10 seconds that the simulation was run. During this time the vehicle travelled a longitudinal distance of 200 m. The other parameters, $\dot{\psi}$, a_y , and ψ_h all remained at 0 and did not change.

The second simulation performed, to test the model, was a step input for δ . Again, a straight

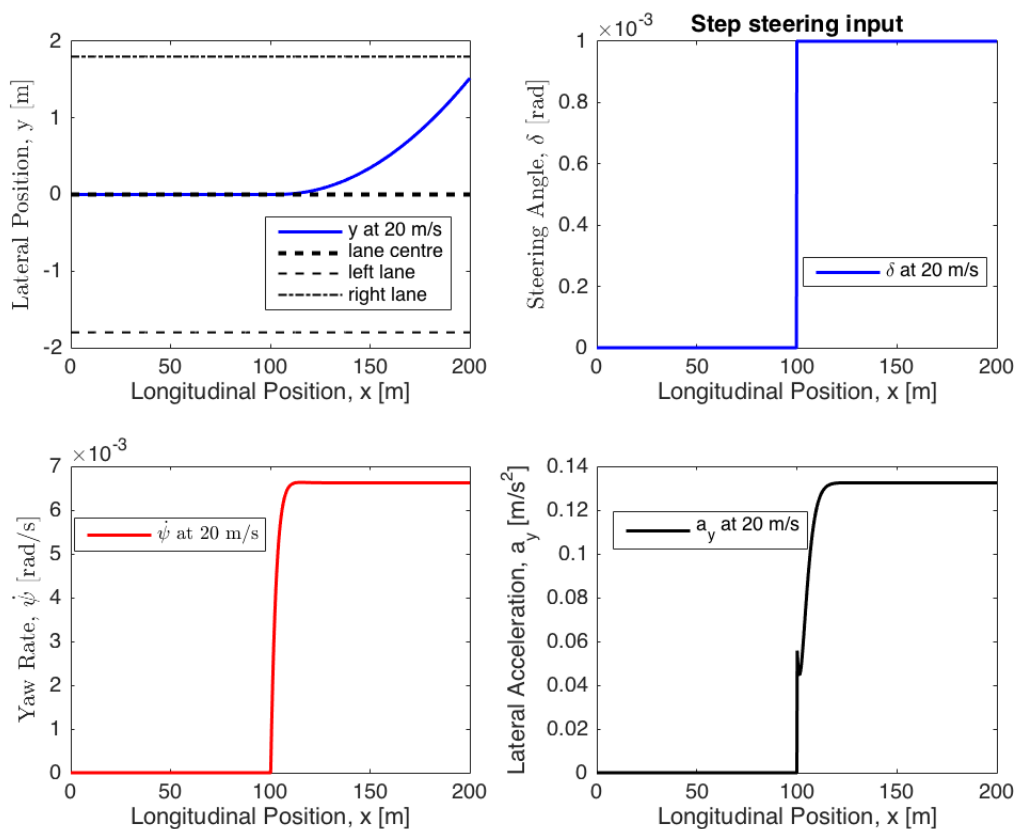


Figure C.2: Step Steering Input for Model Verification

road is simulated so ρ remains at 0 m^{-1} .

The initial conditions for this simulation are $[\dot{\psi} = 0 \quad \beta = 0 \quad \psi_h = 0 \quad y = 0]^T$. At the mid-point of the simulation (at 5 s), a step input of $\delta = 0.001 \text{ rad}$ will be applied to the system and maintained for the remainder of the simulation.

It is expected that the vehicle will follow a curved lateral path away from the centre of the lane due to the constant steering angle value. The constant steering is also expected to give a constant yaw rate. This can be seen in figure C.2 above. From the lateral position plot (top left, in Figure C.2 above), it can be seen that the vehicle does indeed move in a curved path. A constant steering input would result in that lateral motion. The yaw rate plot (bottom left, in Figure C.2 above), shows that the vehicle rotates as the steering angle increases and when the steering angle becomes a constant value of 0.01 rad , the yaw rate settles at approximately $7 * 10^{-3} \text{ rad/s}$. The step input is a sudden change in steering input to the vehicle model and as a result, the vehicle jerks in the lateral direction. This can be observed as the spike in the lateral acceleration plot (bottom right, in Figure C.2 above). This is due to the dynamics of the vehicle at this velocity.

The third model verification simulation that was run was with $\delta = \sin(t)$ where $\sin(t)$ is a sine wave with a predefined amplitude and frequency. An amplitude of 0.01 rad and a frequency of 3 rad/s is chosen ($\delta = 0.01\sin(3t)$). This is to simulate harsh steering in one direction and steering back to correct the vehicle. The input and the results are shown in the figure C.3 below. As

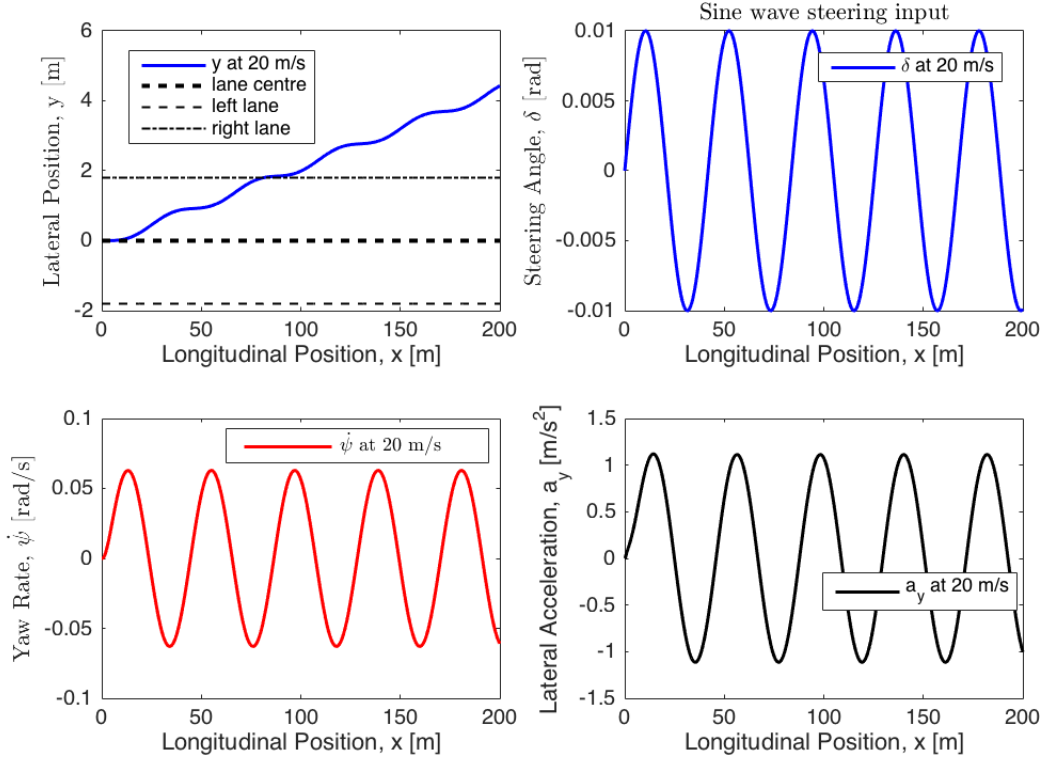


Figure C.3: Sine Steering Input for Model Verification

expected, the sine wave produces an oscillatory movement of the vehicle away from the starting position. As the vehicle steers through one time period (0 - 50 m in longitudinal position), it can be seen that the lateral position changes by 1 m. The vehicle moves 1 m in away from it's previous position. This can be interpreted as a lane change path. There are oscillations in ψ and a_y as well and this is expected as the vehicle is changing lateral position and is also rotating while doing so. The lateral acceleration reaches a peak of 1 m/s^2 . This is a fairly high value in lateral acceleration. This is expected as a steering angle of 0.01 rad is quite a harsh steering at 20 m/s .

The model can be considered valid as the simulated vehicle behaves in a similar fashion to what would be expected of an actual vehicle given the known inputs of steering angle, δ , input. The linear vehicle model and camera model is valid up to lateral accelerations of 4 m/s^2 as the vehicle behaves linearly only during low lateral accelerations. With a verified linear vehicle model, a controller can be designed.

Appendix D

Simulink Simulations

In addition to the simulink simulation presented above in Chapter 4, individual lateral offset and relative heading simulations were run. The simulation setup and results are presented in the sections below.

D.1 Lateral Deviation Simulation

For the first simulation, a straight road is simulated, with the initial condition applied to the lateral position. The host vehicle begins at 0.2 m away from the centre of the lane, closer to the right lane. The table below shows the parameters to create the scenario.

Table D.1: Parameters for lateral deviation simulation

No.	Parameter	Value	Units
1.	Acceleration, a_x	0	m/s^2
2.	Curvature, ρ	0	m^{-1}
3.	Initial Heading, ψ_{h0}	0	rad

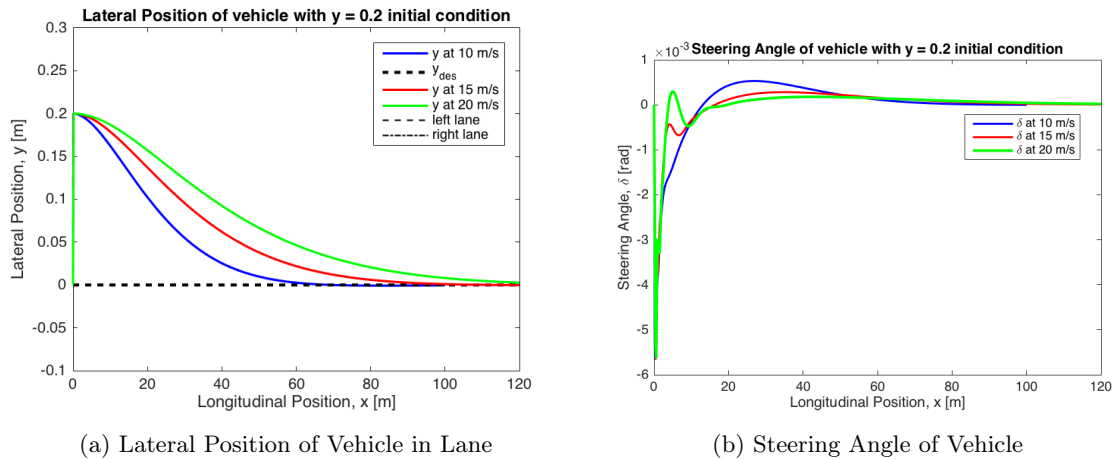


Figure D.1: Lateral Position and Steering Angle Plots for Lateral Deviation Simulation

From the simulation results in figure D.1 we can see that the vehicle corrects the lateral position relatively quickly. At lower speeds, the controller is able to achieve larger steering angles but as the speed increases, the steering angle decreases. This is expected because it does not require high steering angles in order to move the vehicle to the desired lateral direction at high speeds. The model is created with low steering, low slip, and low curvature so at high speeds, the model becomes less valid. At 20 m/s, from figure D.1b the maximum steering angle achieved is $-0.006 \text{ rad} = -0.344 \text{ deg}$ wheel angle = -6.5 deg steering wheel angle. The steering is not significant but at high speeds it will cause the car to move a greater amount in the lateral direction. This is seen as the steering angle quickly goes back to 0 rad. This appears to have a sudden change in steering angle and is explained because of the chosen controller. As the error in lateral position is high at the start of the simulation (0.2 m), the controller immediately corrects for that by steering. As the vehicle then moves in the lateral direction, at higher speeds, there is a quicker change in lateral direction and therefore the steering angle quickly goes back to 0 deg. It is not expected that there will be a sudden change of 0.2 m in the lateral position of the vehicle so this steering angle is allowed.

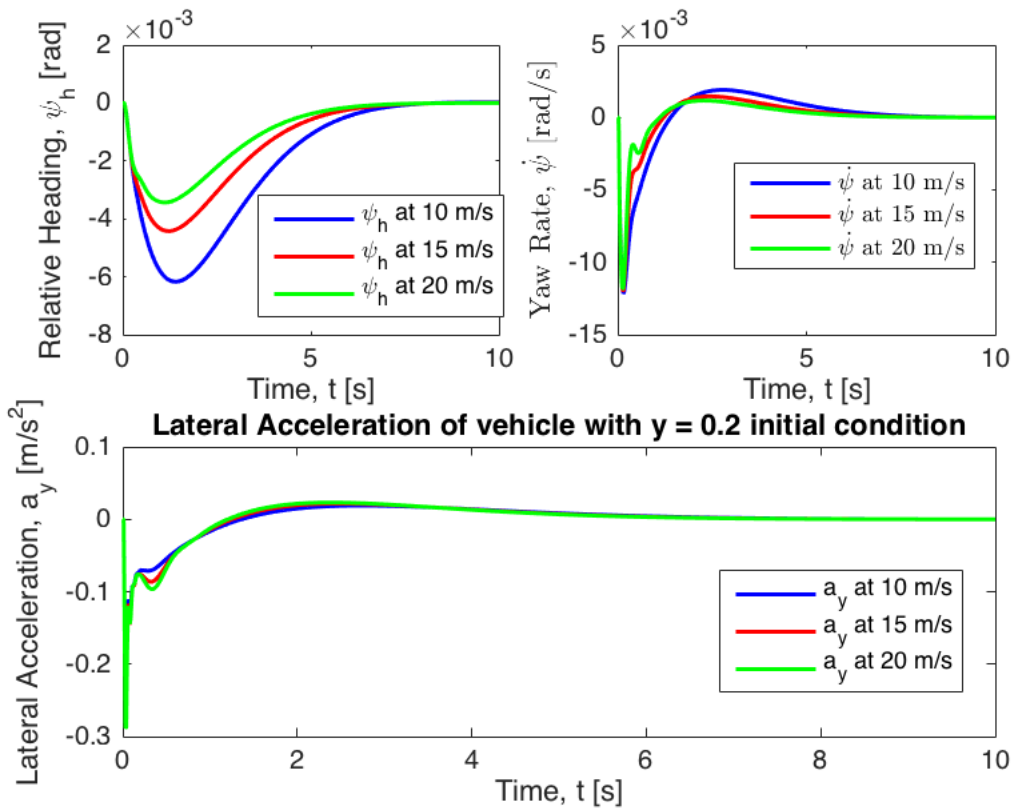


Figure D.2: Relative Heading, Yaw Rate, and Lateral Acceleration Plots Lateral Deviation Simulation

The plots for ψ_h , $\dot{\psi}$ and a_y are also provided in figure D.2 below. These states are plotted over simulation time. From the relative heading plot (top left plot in figure D.2), it can be seen that the vehicle rotates less at higher speeds due to the change in lateral position. The vehicle is moving at a faster rate towards the centre of the lane and as a result, the vehicle orientation with respect to the lane is also correcting quickly. This is good because excessive rotation is not required as that could cause the vehicle to go beyond the centre of the lane and then cause oscillations. The relative

heading can be considered as the vehicle perceiving the change in lane heading as it corrects itself within the lane. As the car turns due to the correction for lateral position, the relative heading will change because the lane will appear differently to the camera model as the vehicle moves. However, it is important to note that the relative heading goes to 0 rad as the vehicle goes to the centre of the lane. This is backed up by the yaw rate plot (top right plot in figure D.2), which shows rotation of the vehicle but then $\dot{\psi}$ goes to 0 rad/s . There is a sudden change in $\dot{\psi}$ at the start of the simulation for the vehicle at 20 m/s but for a short time. This is evident as the steering angle was also large but only for a small period of time. This is again due to the controller reacting to the sudden error in position at the start of the simulation. Despite the high speed, the controller wants to correct for lateral position quickly. The lateral acceleration plot (bottom plot in figure D.2) shows that the maximum lateral acceleration performed by the vehicle is -0.3 m/s^2 at 20 m/s longitudinal velocity. This is relatively high but still falls within the bounds of the model. However, the lateral acceleration plots at 10 m/s and 15 m/s show smoother changes over time.

D.2 Relative Heading Simulation

For the second simulation, a straight road is once again simulated, with the initial condition applied to the relative heading, instead of the lateral position. The host vehicle begins at the centre of the lane ($e_y = 0$). However, it will have an initial heading value such that the car is oriented facing the left lane. A negative heading value means that the lane is oriented to the right of the car's orientation. The vehicle will be simulated at 3 speeds, 10 m/s , 15 m/s , and 20 m/s . The table below shows the parameters to create this scenario.

Table D.2: Parameters for relative heading deviation simulation

No.	Parameter	Value	Units
1.	Acceleration, a_x	0	m/s^2
2.	Curvature, ρ	0	m^{-1}
3.	Initial Heading, $\psi_{h,0}$	-0.004	rad

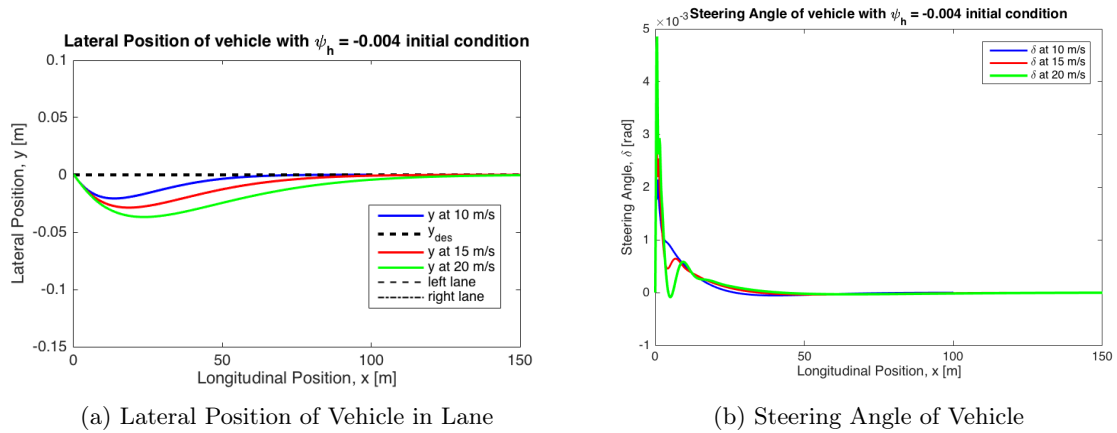


Figure D.3: Simulation Results for Relative Heading Simulation

From the relative heading simulation results, shown below in figure D.3, we can see that the vehicle corrects for relative heading to orient the car back to the orientation of the road. The vehicle covers a greater distance at higher speeds and also moves a greater distance in the lateral direction. This is

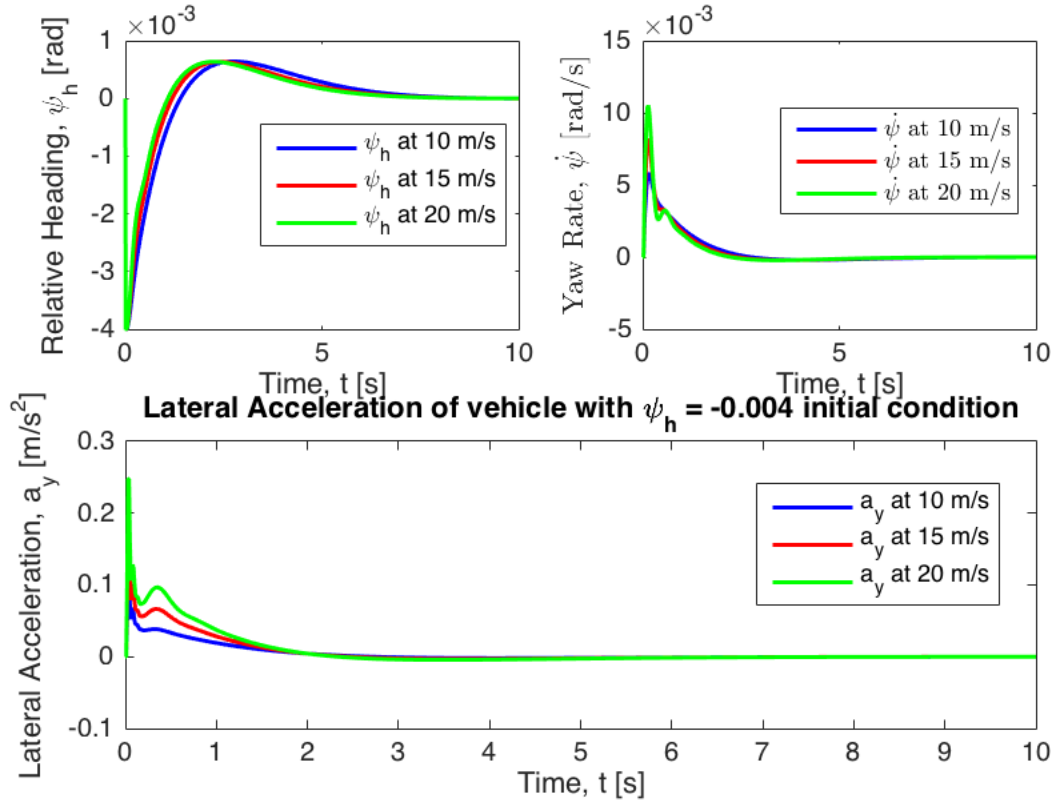


Figure D.4: Relative Heading, Yaw Rate, and Lateral Acceleration Plots Relative Heading Simulation

expected as the vehicle is at constant speed throughout the simulation. However, the steering angle is at a higher value. This is not ideal but can be explained because the rate of change of the error in lateral position is higher at 30 m/s than it is at 10 m/s . As a result, the controller corrects for the error by steering more quickly to get the vehicle back to the centre of the lane. However, the lateral position does not exceed 0.1 m , which means the car does not travel beyond 0.1 m from its starting position. If the width of the car is taken into account along with the lane width of approximately 3.6 m , then the vehicle is still well within the limits of the lane. Once again, a small sudden change in steering angle is noticed as the steering angle settles to 0. This is again because first the controller starts to correct for the error in relative heading. However, as the vehicle continues to move further away from the lane, the controller steers even more. This happens as there is also a build up of lateral position error. At higher speeds, the vehicle will travel further and therefore the controller will want to correct even more. This explains why at higher speeds, the steering angle is at a higher initial value. However this is still within bounds and is acceptable as there is no oscillation in the lateral position plot so the vehicle continues on its trajectory towards the centre of the lane.

The lateral acceleration plot, shown in the bottom plot of figure D.4, shows that the maximum lateral acceleration reached is about 0.25 m/s^2 for the simulation at 20 m/s . This is approximately the same as the lateral acceleration from the previous simulation. It is relatively high as the vehicle moves in the lateral direction. This is expected as the vehicle begins facing one of the lanes. Therefore as soon as the simulation starts, the vehicle is moving further away from the centre of the lane and the controller has a build up of lateral position and relative heading error and needs to immediately correct for both. This causes the high steering angles and the lateral acceleration values. This case

also shows the problems with using proportional and derivative gains as the controller because even a small build up of error (like 0.05 m for lateral position in this scenario) will cause the vehicle to steer to correct back to the centre of the lane. Again, it is considered acceptable as the vehicle meets the simulation criteria presented in Table 4.1 above.