LATERAL CONTROL FOR AUTONOMOUS AND CONNECTED FOLLOWING VEHICLES WITH LIMITED PREVIEW INFORMATION

A Dissertation

by

MENGKE LIU

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Swaroop Darbha
Co-Chair of Committee,	Sivakumar Rathinam
Committee Members,	Swaminathan Gopalswamy
	Shankar Bhattacharyya
Head of Department,	Andreas A. Polycarpou

December 2020

Major Subject: Mechanical Engineering

ABSTRACT

Lateral control of an autonomous and connected vehicle (ACV), especially in emergency situations, is important from the safety viewpoint. In these situations, the trajectory to be followed by an ACV must either be planned in real-time (e.g., for a possible evasion maneuver if the obstacle to be avoided is detected) or be communicated from its preceding vehicle. Typically, the trajectory information is available to the following ACV in the form of GPS time samples. From the viewpoint of lateral control, the lateral velocity information is not readily available, and the feedback structure must reflect this reality. In this work, we develop a methodology to synthesize a lateral control algorithm for a following ACV in a two-vehicle platoon in two steps: (1) From the limited preview information of the trajectory to be tracked via samples of GPS waypoints, and we estimate the radius of curvature of the trajectory using "least-square" estimation and (2) develop a fixed-structure feedback control scheme for following the predecessor by synthesizing the set of stabilizing gains corresponding to lateral position error, heading error and heading rate error. Numerical simulation and experimental results corroborate the effectiveness of the proposed feedback-feedforward schemes.

Based on this proposed feedback-feedforward controller, we investigated Emergency Lane Change (ELC) control problem for a convoy of autonomous and connected vehicles. Typically, an ELC maneuver is triggered by emergency cues from the front or the end of convoy as a response to either avoiding an obstacle or making way for other vehicles to pass. From a safety viewpoint, connectivity of ACVs is essential as it entails obtaining or exchanging information about other ACVs in the convoy. This work assumes that ACVs have reliable connectivity and that every following ACV has the information about GPS position traces of the lead and immediately preceding vehicles in the convoy. This information provides a "discretized" preview of the trajectory to be tracked. Based on the available information, this work focuses on two schemes for synthesizing lateral control of ACVs based on (a) a single composite ELC trajectory that fuses lead and preceding vehicle's GPS traces and (b) separate ELC trajectories based on preview data of

preceding and lead vehicles. The former case entails the construction of a single composite ELC trajectory, determination of the cross track error, heading and yaw rate errors with respect to this trajectory and synthesis of a lateral control action. The latter case entails the construction of two separate trajectories corresponding to the lead vehicle's and preceding vehicle's data separately and the subsequent computation of two sets of associated errors and lateral control actions and combining them to provide a steering command. Numerical and experimental results corroborate the effectiveness of these two schemes.

For multiple vehicle control in a convoy/platooning, identifying vehicles and objects in the vicinity of platooning vehicles is critical for convoy/platooning safety. A neighboring vehicle may change lanes and enter a lane with platooning vehicles. Any deployable platooning system must be able to first identify the presence of a cut-in vehicle before performing any control actions on the vehicles. In this work, we present a sensor fusion algorithm that combines radar and vision data obtained onboard a moving truck to identify the states of the cut-in vehicle. We then present experimental results to corroborate the performance of the proposed algorithms.

DEDICATION

This dissertation is dedicated to my parents for their eternal and unconditional love, support and encouragement.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professor Swaroop Darbha [advisor], Professor Sivakumar Rathinam [co-advisor], Professor Swaminathan Gopalswamy of the Department of Mechanical Engineering and Professor Shankar Bhattacharyya of the Department of Electrical & Computer Engineering.

Funding Sources

Graduate study was supported by a grant from U.S. Department of Transportation, University Transportation Centers Program to the Safety through Disruption University Transportation Center (451453-19C36).

NOMENCLATURE

ACV	Autonomous and Connected Vehicle
CD	Center of Gravity
EOM	Equation of Motion
ESP	Electric Power Steering
ELC	Emergency Lane Change
GNSS	Global Navigation Satellite System
IMU	Inertia Measurement Unit
LIDAR	Light Detection and Ranging
NN	Nearest Neighbour
MHT	Multiple Hypothesis Tracking
PDA	Probabilistic Data Association
I, J, K	Unit vectors of the global frame.
i, j, k	Unit vectors of the vehicle body frame.
e_t, e_n	Tangential and normal vectors to the trajectory
m	Vehicle mass
I_z	Moment of inertia about vehicle center of mass
C_f	Cornering stiffness of the front tires
C_r	Cornering stiffness of the rear tires
α_f	Sideslip angle of the front tires
α_r	Sideslip angle of the rear tires
F_{cf}	Cornering forces at the front wheels
F_{cr}	Cornering forces at the rear wheels

δ_c, δ_f	Steering input and outputs
J_w	Steering wheel inertia
b_w	Torsional viscous damping coefficient of the steering column
K_r	Torsional stiffness of the steering column
ζ, w_n	Damping ratio and natural frequency of steering actuation system
v_x, v_y	Longitudinal and lateral components of vehicle's velocity
θ	Vehicle heading with respect to a frame fixed to the ground
e_{lat}	Lateral error of the vehicle from the trajectory
$\tilde{\theta}$	Heading error of the vehicle
$\dot{\tilde{\theta}}$	Heading rate error of the vehicle
β	Sideslip angles at CG of vehicle
$\mathcal{L}^1$	Space of absolutely Lebesgue integrable functions; i.e., $\int \lvert f \rvert < \infty$
$\mathcal{L}_2$	Space of square-Lebesgue-integrable functions, i.e., $\int \lvert f \rvert^2 < \infty$
K_{sg}	Steer gradient of vehicle
W_f, W_r	Front/rear axle loads of vehicle
k_e, k_θ, k_ω	Gain for lateral error, yaw error and yaw rate error
$\mathbf{K}$	Gain set with parameter k_e, k_θ, k_ω
$\Delta(s; \mathbf{K})$	Characteristic polynomial of system with EPS dynamics

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION, LITERATURE REVIEW, NOVELTY AND THESIS OUTLINE

1.1 Introduction and literature review

One of the primary functions of driving is lane-keeping, which is a central task in autonomous vehicle control research. This task involves having a vehicle follow the center of the lane, while moving forward. Thus the vehicle stays within the boundaries of the lane at all times. However, manual driving accounts for major economic loss annually. In 2018, the Texas Department of Transportation reported 540,561 motor vehicle accidents on major Texas roadways. 18% of these accidents were caused by distracted driving due to factors such as cell phone use or carelessness. However, with the advent of the Advanced Driver Assistant Systems (ADAS), accidents can be reduced. These intelligent systems alert drivers or correct vehicle behavior during dangerous maneuvers, thereby saving lives. Thus, it is of paramount interest to improve autonomous vehicle lane-keeping capabilities.

In this work, a lateral controller is developed for lane-keeping purposes. Two applications of this controller are also explored further, which involve:

1. **The lateral control of a single following Autonomous and Connected vehicle (ACV or simply a vehicle) in two-vehicle platooning, during an emergency lane change scenario.**

   A cue for an emergency lane change can arise from the front of the two-vehicle convoy as an evasive maneuver for obstacle avoidance, or from behind as a response to an approaching emergency vehicle, such as an ambulance. In such an emergency scenario, a lane changing trajectory must be constructed in real-time by a vehicle, as one of the responses for obstacle avoidance, and tracked closely by the vehicle's controller. In a similar scenario, it is possible that there can be multiple vehicles following a leading one, where the leading vehicle constructs the trajectory in real-time (as it has the best view of the obstacle.) The trajectory information is conveyed to the followers in real-time. Usually, this trajectory information may only provide a limited preview for the following vehicle(s); *controlling the lateral dy-*

namics of a following ACV with such limited preview information is the first focus of this work.

2. **The lateral control of a/an convoy of ACV(s), in two or more platooning, during an emergency lane change scenario.**

   Emergency Lane Change (ELC) maneuvers can be initiated usually by observing cues from the front or rear of a convoy of ACVs. For example, pot-holes or debris may serve as frontal cues, while emergency vehicles such as police or ambulances, serve as rear cues. The distinguishing feature of an ELC maneuver from a nominal lane change maneuver is that (a) it is not pre-planned e.g. it is constructed in real-time and that (b) the trajectory to be tracked is different from the center of a lane. Additionally, Vehicle-to-Infrastructure (V2X) communication facilitates exchange of information with infrastructure and other vehicles in the presence of obstacles, computation and transmission of the corresponding sensing/control actions for ELC maneuvers of ACVs in the convoy. *How one can exploit the connectedness of an ACV, in terms of utilizing information from different vehicles or infrastructure. It is not readily apparent in the case of convoys for executing an ELC maneuver, which is the second focus of this work.*

The lateral controller for a vehicle typically consists of a feedforward and a feedback component [2, 3, 4, 5, 6, 7]. A feedforward controller takes the preview information and constructs a trajectory to be tracked, based on which a feedforward steering angle input is computed. If the lateral vehicle dynamic model were to be known exactly, and the vehicle were to be following the desired trajectory with no lateral, and heading errors, such an input would keep the vehicle on the trajectory when there are no disturbing inputs on the vehicle. Since this is rarely the case, a feedback controller is necessary. The task of a feedback controller is to get the vehicle back on the desired trajectory in the presence of disturbances, model uncertainties and initial errors in heading and lateral positions.

The difficulty of lateral controller design for the following ACV is closely related to the in-

formation available to the controller. To perform a lateral control task, one must first identify a trajectory or a path and then have the means to detect the deviation from the trajectory. Fenton et al. [2], who first proposed the two-component lateral control framework, used a wire for guidance; this served as the desired trajectory to track, and the deviation from the wire was fed back in lateral control. Vision systems and GNSS are independent of the road infrastructure for vehicle guidance and control [8, 9, 10, 11, 5]. Infrastructure dependent lane detection was adopted by California PATH; lane tracking with magnetic markers enabled the sensing of lateral deviation of the vehicle; moreover, magnetic markers embedded the curvature (preview) information of the road which was sensed by the vehicle [6, 4, 12]. An overview of lateral control for lane keeping and lane changing is provided in [13, 14, 15].

The work considered here differs from that in the literature in three *distinct* ways: the following ACV does not have a preplanned trajectory to track; rather it must construct a trajectory in real-time based on the limited GPS measurements communicated by the lead ACV in the platoon; in this way, it differs from the virtual trajectory construction scheme proposed by [8] and necessitates connectivity for the following ACV; it also differs from [16] in the construction of the desired trajectory by virtue of lead vehicle's GPS data being transmitted in real-time, whereas in [16], the data is specified a priori; secondly, lane change maneuvers have thus far focused only on the design of a maneuver for the lead (or a single) vehicle [17] and not the following ACVs! Thirdly, the structure of the feedback controller chosen in this work is different from what has been considered in the literature [2, 5, 6].

For ELC maneuvers involving two or more following vehicles in a platoon, one cannot but deal with the problem considered here. In this case, the trajectory to be tracked must be constructed in real-time along with its curvature information; from the sensed position and orientation of the following ACV, its lateral deviation and heading error from the constructed trajectory must be computed and corrected with the help of feedback. Usually, the lateral component of vehicle velocity is not readily available (or very noise prone) and correspondingly, one may not be able to feed back the rate of change of lateral deviation; lack of such information renders the task of

determining/tuning stabilizing controller gains difficult.

In this work, we use the following two steps to design the lateral controller. The first step is to fit the curve of the desired trajectory and to compute error signals that are needed by the feedback controller. We use a "least-square" approach to construct a trajectory consisting of straight-line segments and circular arcs using the communicated position information of the lead vehicle; the lateral deviation of the vehicle from the circular arc and the heading error and its rate can then be computed by knowing the position, longitudinal velocity and heading of the controlled vehicle using sensory measurements (from GPS). Fitting a trajectory with circular arc splines (i.e., with geometric primitives, namely, circular arcs and straight line segments) has attracted research in diverse engineering applications such as CAD, CNC machining, etc. [18, 19, 20, 16]; it is reasonable for this application as roads in the US are constructed as circular arc splines [21]. The variation in radius of the trajectory of a vehicle on the road is usually much smaller than the mean radius of the lanes. As alluded to earlier, an approximation of curves with these primitives have the following advantages: the resulting curve will be (a) invariant to rotation, (b) have a well-defined tangent vector at every point on the trajectory, (c) have a standard arc representation consisting of two arc endpoints and its center. *The primary motivation for approximating the trajectory with circular arcs and straight line segments in this work is as follows: it makes the task of computing feedback signals such as lateral deviation, heading angle error and heading rate error simple; correspondingly, the experimental implementation of the lateral control is easier.*

The second step in the controller design is to build the feedforward and feedback controller. The feedforward component of control is based on the knowledge of track length, radius of curvature, longitudinal velocity, and the steer gradient of the vehicle. The feedback component of the vehicle is based only on three measurements (which are also its states) computed from the previous step, namely - lateral deviation, heading error, and its rate. Since the rate of lateral deviation is not measurable to the desired accuracy, we do not consider it a signal that can be fed back. Consequently, feedback synthesis can be posed as a fixed structure (three parameters) stabilization problem. We construct the set of stabilizing controllers using methods described in [22] and pick

a controller from this set to corroborate the lateral vehicle controller design proposed here.

When dealing with a multiple vehicle convoy (more than two vehicles in a platoon), a natural question arises as to what information, if at all, should the ACVs communicate and how should each ACV be controlled so that an emergency lane change maneuver can be performed successfully. As can be expected, the flow of information and architecture for control can have a significant influence on how the maneuver is carried out. Hence, a design of the controller, different from a single- vehicle control, is required.

In executing an ELC maneuver, a controlled vehicle must decide the trajectory to track based on the available data from the lead and other vehicles in the convoy and from its onboard sensors and then extract the requisite feedback/feedforward information from the available information to take a control action. As in the longitudinal control case, if a following vehicle tracks its preceding vehicle's trajectory, error amplification can occur; from a safety viewpoint, it may be better to use the information of the lead vehicle's trajectory in conjunction with that of its preceding vehicle. In this work, we define the first vehicle of platoon is "leading vehicle", and the one ahead of i^{th} vehicle is "preceding vehicle". There is an attendant issue of how one can utilize such information from different vehicles and is not readily apparent in the case of lateral control of convoys for executing an ELC maneuver. This work deals with this issue in Chapter 5.

Another contribution of this work is to demonstrate the mobility and safety benefits of connectivity for Emergency Lane Change Maneuvers. In an ELC maneuver, guidance (in terms of trajectory to be tracked) from infrastructure sensing cannot be directly obtained; it must rather be computed in real-time. To elaborate further, consider the figure 1.1. Consider a convoy of vehicles, each of length L_c, traveling in a two-lane highway with vehicle A (shown) being the $(i-1)^{st}$ ACV and vehicle B following it being the i^{th} ACV in the convoy. Suppose there is an obstacle in the *left half* of the right lane as shown; the convoy makes an ELC as a response to evading the obstacle. The following distance of vehicle B (and every ACV) is d_0; clearly, if d_0 is small, the obstacle will be occluded by the preceding vehicles in the convoy. Let Δ_i be the distance at which the i^{th} following ACV can clearly see the obstacle *for the first time* without any occlusions in front of it,

i.e., this can happen only if the preceding vehicle has moved out of the line of sight between the i^{th} vehicle and the obstacle. If we make the assumption that the lateral displacement of an ACV is proportional to the longitudinal distance it has traveled, and that an ACV will not occlude the view of its following vehicle unless it travels a distance $\frac{2W}{3}$ laterally, where W is the width of the lane (and $\frac{2W}{3}$ is the car width), we can derive a simple recursive relationship between Δ_i and Δ_{i-1} as follows: The longitudinal distance traveled by the preceding (or $(i-1)^{st}$ ACV) when it moves by a distance $\frac{3W}{2}$ for a fully executed ELC is Δ_{i-1}; however, to not occlude the i^{th} ACV, if we assume that its right edge must move laterally by a distance $\frac{2W}{3}$ (equal to the car width), the distance moved by i^{th} and $(i-1)^{st}$ ACVs is $\frac{4}{9}\Delta_{i-1}$. Consequently,

$$\Delta_i = \Delta_{i-1} + d_0 + L_c - \frac{4}{9}\Delta_{i-1} = \frac{5}{9}\Delta_{i-1} + d_0 + L_c.$$

Without V2X communication, there is a danger of the i^{th} ACV seeing the obstacle closer than its

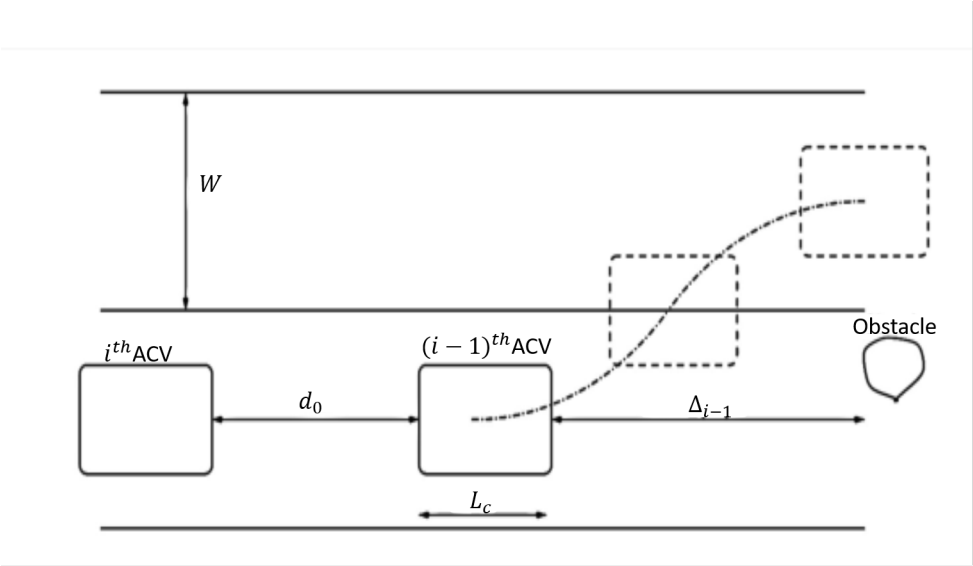

Figure 1.1: An Emergency Lane Change (ELC) Scenario - d_o denotes the desired following distance; Δ_{i-1} is the distance at which the $(i-1)^{th}$ following ACV can clearly see the obstacle.

predecessor, i.e., $\Delta_i < \Delta_{i-1}$, especially when there is no V2X communication and when $d_0 + L_c < \frac{4}{9}\Delta_1$. From the view of safety and convenience, Shui [23] indicated that

$$\Delta_1 \geq 2.4v\sqrt{\frac{W}{A_{max}}},$$

where v is the velocity of the ACV while making a lane change, A_{max} is the maximum allowable lateral acceleration. Correspondingly, one can get a lower bound on how close ACVs can follow in the absence of V2X communications in order to execute the ELC while meeting basic safety and comfort requirements:

$$d_0 \geq \frac{4}{9}\Delta_1 - L_c = \frac{16}{15}v\sqrt{\frac{W}{A_{max}}} - L_c.$$

In other words, every following ACV must maintain a minimum time headway of $\frac{16}{15}\sqrt{\frac{W}{A_{max}}}$ seconds; for a naturalistic lane change maneuvers [24] $A_{max} < 1\ m/s^2$; with a typical lane width of $W \approx 3\ m$, the minimum time headway must at least be $1.85\ sec$; however, with V2X communication, this gap be significantly reduced and mobility can be correspondingly improved.

1.2 Novelty

The *novelty* of this work is as follows: (a) the problem of tracking a vehicle based on its communicated trajectory information (a la bread crumbs dropped by the previous vehicle) has not been considered in the literature before, (b) the use of "least-squares" technique to fit a trajectory consisting of a circular arc and straight line segments using the preview (bread crumb) information of the preceding vehicle for lateral control of following ACVs is novel; it is required for the computation of feedforward control, and in the computation of feedback signals, and (c) the use of fixed structure controller synthesis techniques to account for the lack of sensor measuring the rate of change of error in lateral vehicle position. (d) to provide a direct method to fuse the information of the lead and preceding vehicles to synthesize target trajectories that aid lateral string stability by suppressing the amplification of crosstrack errors, and (e) to corroborate numerically and experimentally the performance of the lateral controller using the feedback controller synthesized in the earlier work of the authors [25, 26]. An additional novel contribution of this paper is to demon-

strate that the frozen parameter controller design approach guarantees closed loop stability if (a) the longitudinal speed of ACV exceeds a certain non-zero threshold and is upper bounded, (b) its longitudinal acceleration is square integrable. This is reasonable given the finite duration of ELC maneuvers.

1.3 Outline of Dissertation

This paper is organized as follows: in Chapter 2, we present the scheme for computing the optimal "least-square" fit circular arc from the limited preview information communicated to the controlled vehicle. Details of computation of signals for feedback and feedforward controller design can be found in this chapter. In Chapter 3, we present a model for lateral vehicle dynamics that we use for controller design. The model is represented by the error signal states, which is helpful for controller design; we also supply the necessary experimental details of finding the parameters required for lateral control. In Chapter 4 , we outline the procedure for constructing the feedback gains for enabling an autonomous and connected following vehicle to track the desired trajectory. Numerical and experimental results presented at the end of the chapter corroborate the effectiveness of designed controllers. In Chapter 5, we investigated Emergency Lane Change (ELC) control for a convoy of autonomous and connected vehicles. We present two possible architectures for constructing an ELC trajectory for a convoy of autonomous and connected vehicles lateral control with Limited Preview Information. Numerical/Simulation results presented at the end of the chpater corroborate the effectiveness of lateral controllers for a convoy of autonomous and connected vehicles. In Chapter 6, we investigated the problem of identification vehicles and objects in the vicinity of platooning vehicles. An Extended Kalman Filter(EKF) is design to estimated the cut-in vehicle' status based on radar and camera fused data. Experimental results support the effectiveness of this design.

2. FITTING CIRCULAR ARCS AND STRAIGHT LINE SEGMENTS THROUGH THE GIVEN DATA

In this section, we present the definition of the preview section (or preview data to be more accurate) first, then demonstrate how to classify the given preview section data corresponds to a circular arc or a straight line segment. Subsequently, we provide a procedure for computing the radius of curvature for feedforward controller and feedback error signals(lateral error) for feedback controller design. Figure 2.1 is the flow chart of this chapter.

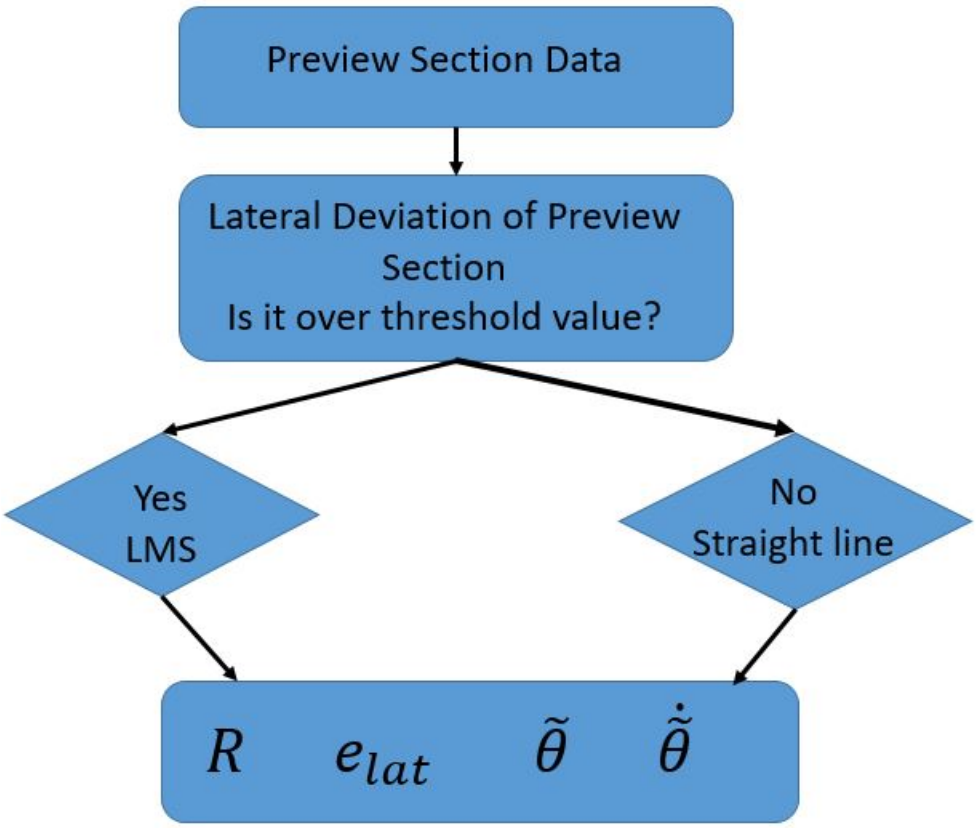

Figure 2.1: Flow chart for curve fitting and error signals derivation

Let the preview information/data be represented by $(x_1, y_1),\ (x_2, y_2),\ \ldots, (x_N, y_N)$, i.e., the

N preview/recent samples of the preceding vehicle's trajectory. Preview data will be based on the current location of the vehicle. The number N could be picked based on the time between samples; for example, if the following vehicle were to maintain a one second of time headway[1], and the GPS/communication sample frequency is 20 Hz, then we can pick $N = 20$; in this case, any data within $1\ sec$ ahead of the current time-stamped data of preceding vehicle's trajectory will constitute preview information. Note that the data $(x_1, y_1), \ldots, (x_N, y_N)$ could be contaminated with noise.

The goal of this section is as follows:

- Based on given preview data: $(x_1, y_1),\ (x_2, y_2),\ \cdots, (x_N, y_N)$, we must decide if it the trajectory of the preceding vehicle corresponds to a straight line or a curve.
- If it has been determined that the data corresponds to a curved trajectory, one must then fit a center (X_c, Y_c), radius R of the "least-square fit" circular arc through these points.
- Once the trajectory is calculated, one must compute the feedback signals: the lateral error of the vehicle from the trajectory, e_{lat}, the heading error of the vehicle, $\tilde{\theta}$, and the heading rate error, $\dot{\tilde{\theta}}$.

Clearly, the above-listed tasks are important for computing feedforward and feedback control inputs – the first two tasks are crucial for the computation of feedforward control computation depending on whether the road is straight or is curving.

The more general problem of fitting a circular arc spline has been considered in the literature [16, 20, 18, 27, 19]; the problem considered here is a simplified version but is based on data that is constantly arriving. In some sense, it is more like a filtering problem that we have here as against a smoothing problem considered in [16, 20, 18, 27, 19].

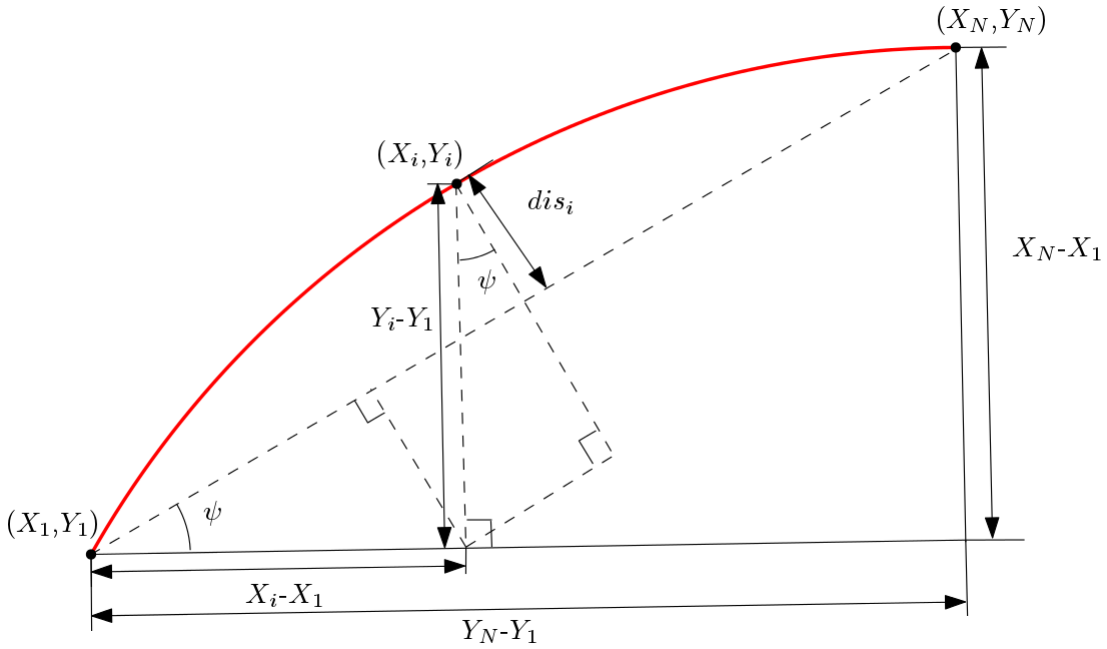

Figure 2.2: Lateral deviation of preview data with a straight line segment connecting the first and last data points

2.1 Determining If Preview Information Represents a Straight Line Segment or a Curved Trajectory

We use a simple heuristic: Since the data is time-stamped and can be sequenced, we connect the first sample (x_1, y_1) and the last sample (x_N, y_N) with a straight line segment, $\mathcal{L}$, and compute, for each i between 1 and N, the distance (dis_i) of the samples (x_i, y_i) from $\mathcal{L}$ using the formula:

$$\begin{aligned} dis_i &= |(Y_i - Y_1)\cos\psi - (X_i - X_1)\sin\psi| \\ &= |\frac{(Y_i - Y_1)(Y_N - Y_1) - (X_i - X_1)(X_N - X_1)}{\sqrt{(X_N - X_1)^2 + (Y_N - Y_1)^2}}| \end{aligned}$$

If $\max_i dis_i < 0.1\ m$, we consider the preview data to represent a straight line segment, as shown as Fig. 2.2; else we fit a circular arc through the given data as in the next subsection.

[1]Time Headway: If $\Delta(v)$ is the desired following distance of a vehicle as an affine function of its velocity, v, i.e., $\Delta(v) = \Delta(0) + hv$, we refer to h as the time headway; note $\Delta(0)$ is its desired standstill following distance.

2.2 Fitting a Circular Arc Through Preview Information

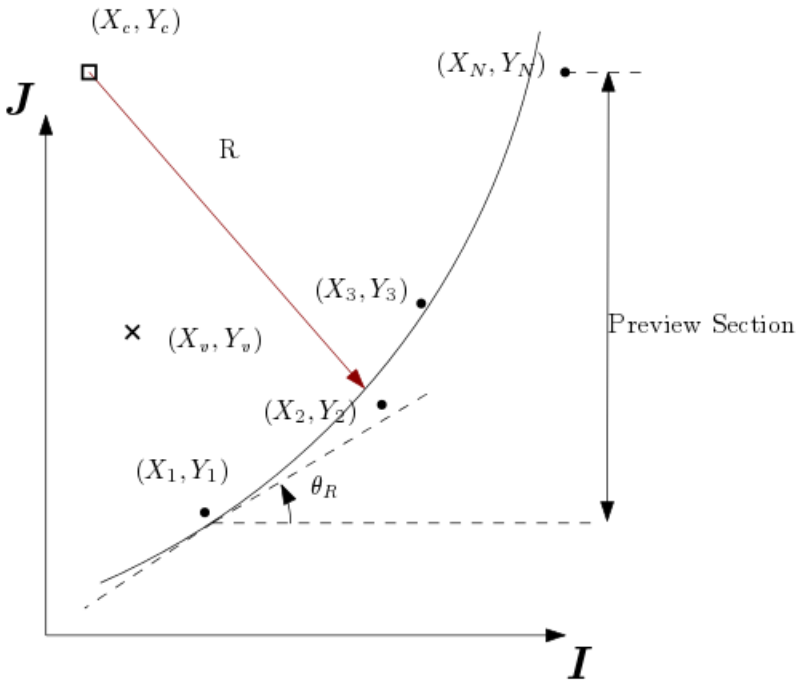

Figure 2.3: Preview information with predicted center position and radius for curve

In this case, the problem can be posed as follows: Given $(x_1, y_1),\ (x_2, y_2),\ \cdots, (x_N, y_N)$, find a "least square fit" of a circular arc, i.e., find the center (X_c, Y_c) and radius R so that the error

$$J = \sum_{i=1}^{N} [R^2 - (x_i - X_c)^2 - (y_i - Y_c)^2]^2$$

is minimized. With each data (x_i, y_i), this objective function assigns an area error – the difference between the area of a circle of radius R and the area of the circle that passes through (x_i, y_i) while being centered at (X_c, Y_c). The total sum of the square of the area errors is π^2 times the objective function J.

There is a plethora of ways to fit a circular arc through a given data [27, 28]; this choice is made to exploit the simplicity of computing the center and radius of the arc with at most three

linear algebraic equations: the first step is to compute the center (x_C, y_C) using the following set of equations:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{bmatrix} \begin{bmatrix} x_C \\ y_C \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix},$$

where

$$\begin{aligned}
a_{11} &:= (\sum_{i=1}^{N} x_i)^2 - N(\sum_{i=1}^{N} x_i^2), \\
a_{12} &:= (\sum_{i=1}^{N} x_i)(\sum_{i=1}^{N} y_i) - N(\sum_{i=1}^{N} x_i y_i) \\
a_{22} &:= (\sum_{i=1}^{N} y_i)^2 - N(\sum_{i=1}^{N} y_i^2), \\
b_1 &:= \frac{1}{2}[(\sum_{i=1}^{N} x_i)(\sum_{i=1}^{N} x_i^2 + y_i^2) - (\sum_{i=1}^{N} x_i(x_i^2 + y_i^2))], \\
b_2 &:= \frac{1}{2}[(\sum_{i=1}^{N} y_i)(\sum_{i=1}^{N} x_i^2 + y_i^2) - (\sum_{i=1}^{N} y_i(x_i^2 + y_i^2))]
\end{aligned}$$

One can explicitly solve for x_C, y_C as:

$$x_C = \frac{a_{22}b_1 - a_{12}b_2}{a_{11}a_{22} - a_{12}^2}, \qquad y_C = \frac{a_{11}b_2 - a_{12}b_1}{a_{11}a_{22} - a_{12}^2}.$$

The radius of curvature, R, can be calculated using

$$R = \sqrt{\frac{1}{N}\sum_{i=1}^{N}[(x_i - x_C)^2 + (y_i - y_C)^2]}.$$

2.3 Computation of Feedback Error Signals

In Fig. 2.4 and 2.5, I, J represent the unit vectors of the global/ground frame. Let (X_v, Y_v) denote the position of ego vehicle in the global frame (as $Z_v = 0$ because the motion of the vehicle is assumed to be in the plane $Z_v = 0$). Computation of feedback error signals depends on whether the trajectory represented by preview data is a straight line segment or a circular arc. In the former case, the desired heading rate is zero; as can been seen from the illustration below, the lateral error

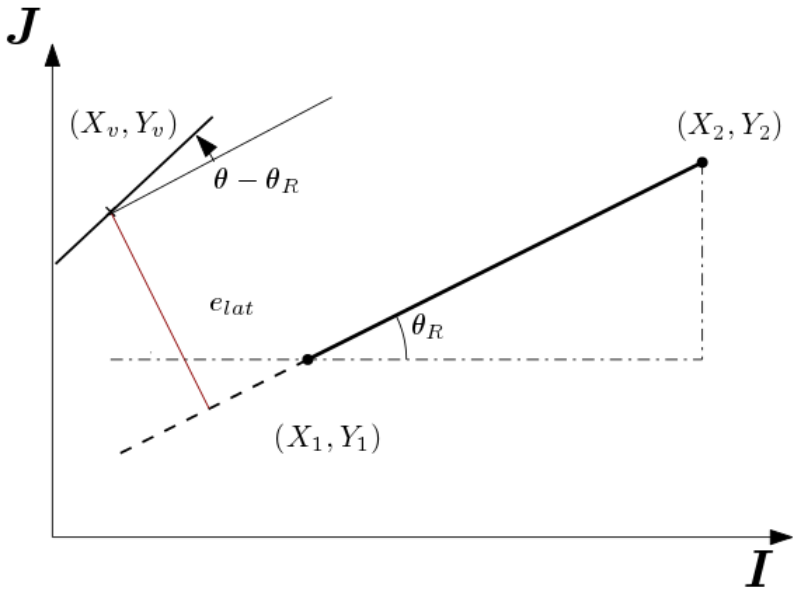

Figure 2.4: Heading and lateral errors for a straight line segment

can be computed to be:

$$e_{lat} = \frac{Y_v - mX_v - c}{\sqrt{1 + m^2}}.$$

where

$$y = mx + c$$

is the straight-line equation based on closest point (X_1, Y_1) and second closest point (X_2, Y_2). Desired yaw angle θ_R is also based on these two points. The yaw error is given by

$$\tilde{\theta} := \theta - \theta_R,$$

and the yaw rate error is given by:

$$\dot{\tilde{\theta}} := \dot{\theta}.$$

If the trajectory is a circular arc, then (X_0, Y_0) is the projection of (X_v, Y_v) onto the circular arc, i.e., it is the point on arc closest to vehicle's current position (X_v, Y_v)) and can be constructed by drawing a line joining the center of the circle (X_c, Y_c) to the current position of the vehicle (X_v, Y_v) and extending, if necessary, to meet the circular arc at (X_0, Y_0). The angle made by the tangent at (X_0, Y_0) with the positive I axis of the ground will be represented by θ_R. As shown, unit vectors i, j are attached to the vehicle along the longitudinal and lateral axes respectively, and the angle θ made by i with I represents the heading θ in the governing equations of motion.

The lateral error can be defined as:

$$e_{lat} := R - \sqrt{(X_v - X_c)^2 + (Y_v - Y_c)^2},$$

The yaw error can be defined as:

$$\tilde{\theta} := \theta - \theta_R$$

where θ_R is defined as based on the closest point and second closest point on the preview section, correspondingly, the yaw rate error is:

$$\dot{\tilde{\theta}} := \dot{\theta} - \frac{v_x}{R}.$$

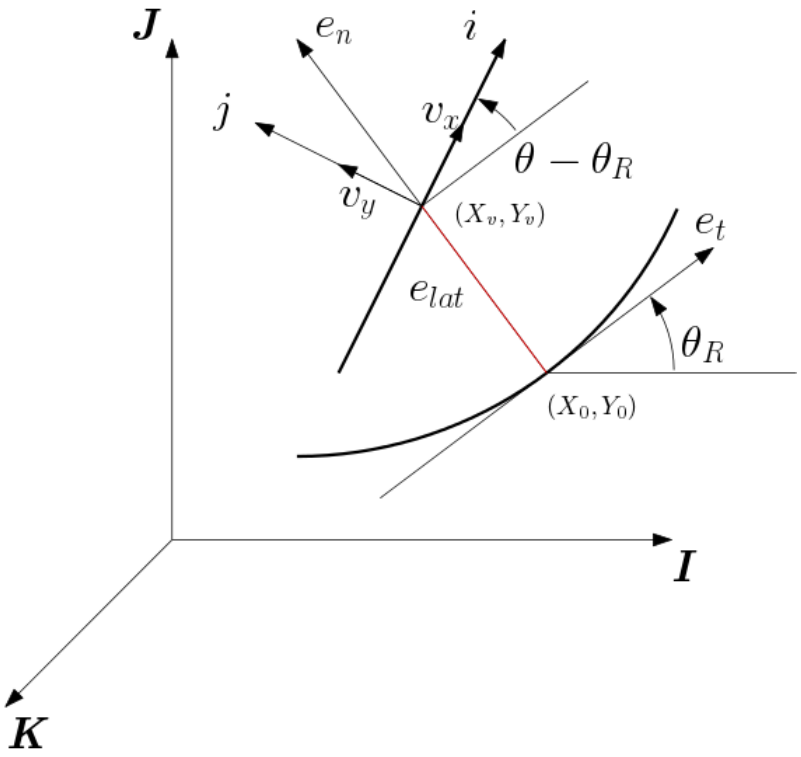

Figure 2.5: Heading and lateral errors for a circular arc

2.4 Concluding Remarks

In this section, we presented how to classify the preview section into a circular arc or a straight line segment. The radius of curvature for feedforward control command can be computed as well as feedback error signals. In the next chapter, we are going to use this error signals to represent our lateral dynamic model, which is essential for the lateral controller design for autonomous and connected following vehicles.

3. LATERAL DYNAMIC MODEL FOR CONTROL*

In this chapter, we will present the lateral dynamic vehicle model for controller design firstly. The lateral dynamic model will be expressed using the error signals of the feedback controller; doing so will aid controller design. In the second section, some key parameters in this modified model will be estimated, such as vehicle weight, moment of inertia, and wheel length. Based on these parameters, we present the determination of front and rear cornering stiffness for feedback controller design and steer gradient for feedforward controller design. Furthermore, we consider a "more" realistic (singularly perturbed) actuation model; we combine a second-order linear steering model with the modified lateral dynamic model. The determination of steering actuation system parameters will also be present in this chapter.

3.1 Original Lateral Dynamic Model for Control

As in [6, 12, 29, 30, 31, 32, 7], a bicycle model is adopted here for lateral control. In this work, we focus on the problem of vehicle lateral motion control with constant longitudinal velocity and/or the variation of it is small and not critical. In such a case, the bicycle model is a simple but very useful vehicle model [33].

Let v_x, v_y denote the longitudinal and lateral components of vehicle's velocity and θ be the vehicle heading with respect to a frame fixed to the ground (as shown in Fig. 3.1). Let I, J represent the unit vectors of the global frame and i, j be the unit vectors attached to the vehicle in the longitudinal and lateral directions respectively. Let m, I_z be the vehicle mass and moment of inertia about its center of mass, and a, b be the distance of the center of mass to the front and rear axles respectively. Let C_f, C_r be the cornering stiffness of the front and rear tires; α_f,α_r be the sideslip angle of the front and rear tires; and F_{cf}, F_{cr} be the cornering forces at the front and rear wheels respectively. For a front-steered vehicle, we define steering input as δ_f. Before we apply Newton-Euler's equations, we list the assumptions for the validity of the model [30], [34]:

*Parts of this chapter are reprinted with permission from [1].

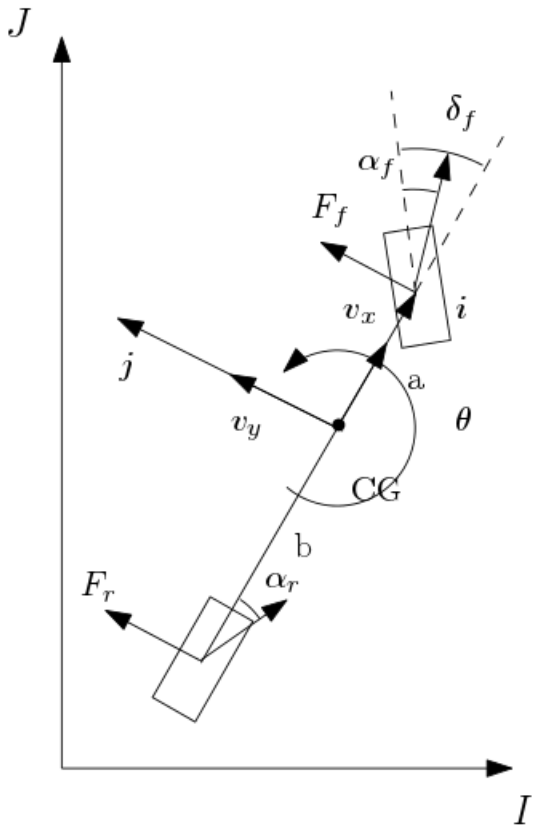

Figure 3.1: Illustration of bicycle model

1. The radius of turn, R, is far larger than wheelbase L.

2. The left and right steer angle must be approximately the same, and the value is approximately equal to steering input (δ_f).

3. The side slip angles of front wheels α_f are equal, as are the slip angles of rear wheel side slip angle α_r.

4. Side slip angles at CG β is small, so that

$$\beta = \arctan \frac{v_y}{v_x} \approx \frac{v_y}{v_x}$$

5. The radius of turn R, the vehicle side slip angle β, and the yaw rate of vehicle $\dot{\theta}$ are fixed during a steady-state turn. The lateral velocity in a steady-state turn $v_y = \dot{\theta} R$,and the lateral

acceleration in steady-turn $a_y = \frac{v_y^2}{R}$.

6. Linear Model for Cornering Forces: $F_r = C_r\alpha_r$, $F_f = C_f\alpha_f$

From Newton-Euler's equations, we obtain lateral vehicle dynamic equation of motion:

$$m(\frac{dv_y}{dt} + v_x\dot{\theta}) = F_{cf} + F_{cr}, \tag{3.1}$$

$$I\ddot{\theta} = aF_{cf} - bF_{cf}. \tag{3.2}$$

In this work, we only focus on lateral motion control and assume that longitudinal velocity is nearly a constant; furthermore, we assume that the slip angles are small. Hence, we use linear tire force model, the tire force can be represented as:

$$F_f = C_f\alpha_f \tag{3.3}$$

$$F_r = C_r\alpha_r \tag{3.4}$$

The tire slip angle (such as α_f or α_r) is defined as the angle from tire velocity to the direction of its orientation; from Fig 3.1, the geometric relationship can be determined to be:

$$\alpha_r = -(\frac{v_y - b\dot{\theta}}{v_x}) \tag{3.5}$$

$$\alpha_f = \delta_f - \frac{v_y + a\dot{\theta}}{v_x} \tag{3.6}$$

Furthermore, we can find out the relationship between steering angle (δ_f) and tire slip angles (α_r, α_f):

$$\alpha_f - \alpha_r = \delta_f - \frac{L}{R} \tag{3.7}$$

Combinin the Eqns (3.1) - (3.6), we obtain:

$$m(\frac{dv_y}{dt} + v_x\dot{\theta}) = C_f\delta_f - \frac{C_f + C_r}{v_x}v_y - \frac{aC_f - bC_r}{v_x}\dot{\theta}, \tag{3.8}$$

$$I\ddot{\theta} = aC_f\delta - \frac{aC_f - bC_r}{v_x}v_y - \frac{a^2C_f + b^2C_r}{v_x}\dot{\theta}. \tag{3.9}$$

These Equations of Motion (EOM) correspond to lateral dynamic model. We notice that it is a set of equations based on the states of a vehicle. However, for the feedback controller design, it is desired to cast the equations of motion using errors in position and heading with respect to the desired trajectory. The EOMs above need to be simplified and modified.

3.2 Simplifying Original Lateral Dynamic Model Based on Error Signal States

The relative lateral error (e_{lat}), yaw error ($\tilde{\theta}$) and yaw error rate ($\dot{\tilde{\theta}}$) can be computed using the results in the previous section based on the computed trajectory.

From the Eqns 3.8, we notice it is a set of nonlinear, vehicle states based equations. In order to express the EOMs using feedback error signals, we need to establish the kinematic relationship between states of a vehicle and the errors in position and heading concerning the desired trajectory.

Consider a vehicle (modeled as a rigid stick, bold line in Fig. 3.2) trying to follow a trajectory (bold curve in Fig. 3.2) shown in Fig. 3.2. Let (X_v, Y_v) denote the position of vehicle's CG in an inertial frame and (X_0, Y_0) is its projection onto a trajectory that it must track. Let e_{lat}(maroon line in Fig. 3.2) be the perpendicular (shortest) distance of (X_v, Y_v) to the trajectory (or equivalently (X_0, Y_0)). Let e_t, e_n be the tangential and normal vectors to the trajectory at (X_0, Y_0) respectively and i, j be the unit vectors attached to the vehicle in the longitudinal and lateral directions respectively. Let I, J be the inertial reference frame. Let θ, θ_R be the angles made by i, e_t with the x-axis of the inertial reference frame.

Define the following position vectors for the vehicle CG and its projection:

$$R_v = X_vI + Y_vJ, \quad R_0 = X_0I + Y_0J \tag{3.10}$$

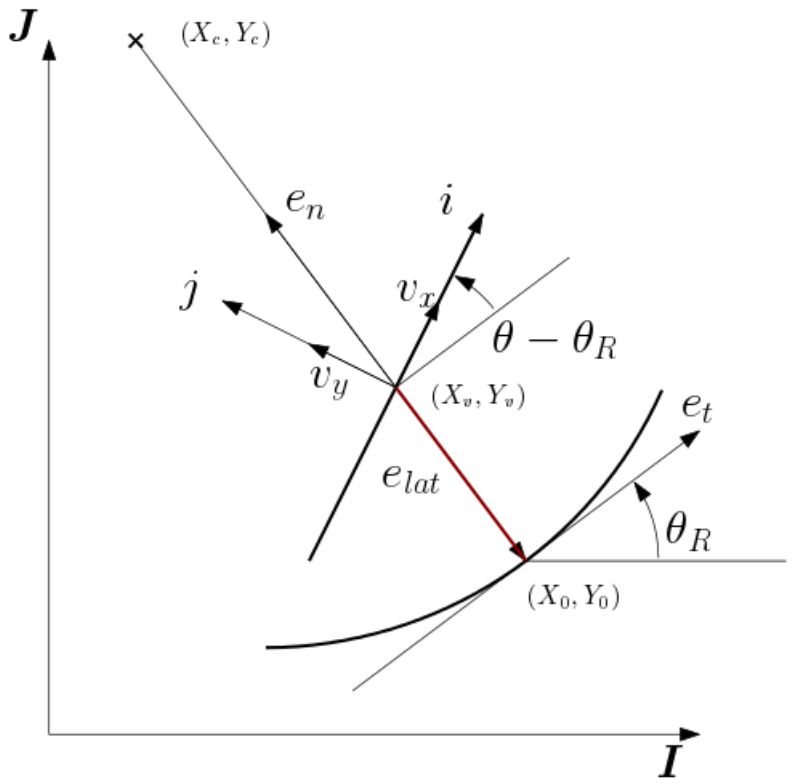

Figure 3.2: Illustration of Heading and position errors in lateral direction.

and

$$R_v = R_0 + e_{lat}e_n \tag{3.11}$$

Note that

$$\frac{de_t}{dt} = \dot{\theta}_R e_n, \quad \frac{de_n}{dt} = -\dot{\theta}_R e_t \tag{3.12}$$

Hence:

$$\frac{dR_v}{dt} = \frac{dR_o}{dt} + \dot{e}_{lat}e_n - e_{lat}\dot{\theta}_R e_t \tag{3.13}$$

Note that

$$\frac{dR_0}{dt} = R\dot{\theta}_R \tag{3.14}$$

where R is the radius of curvature of the track at (X_0, Y_0), it follows that:

$$\frac{dR_v}{dt} = (R - e_{lat})\dot{\theta}_R e_t + \dot{e}_{lat}e_n = v_x i + v_y j \tag{3.15}$$

where v_x and v_y are the components of velocity along the longitudinal and lateral direction. Using coordinate transformation, we get:

$$\begin{bmatrix} (R - e_{lat})\dot{\theta}_R \\ \dot{e}_{lat} \end{bmatrix} = \begin{bmatrix} \cos(\theta - \theta_R) & -\sin(\theta - \theta_R) \\ \sin(\theta - \theta_R) & \cos(\theta - \theta_R) \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix} \tag{3.16}$$

Similarity, the acceleration of vehicle can be represent as:

$$\frac{d^2 R_v}{dt^2} = (\dot{R} - \dot{e}_{lat})\dot{\theta}_R e_t + (R - e_{lat})\ddot{\theta}_R e_t + (R - e_{lat})\dot{\theta}_R^2 e_n + \ddot{e}_{lat} e_n - \dot{e}_{lat}\dot{\theta}_R e_t, \tag{3.17}$$

$$= [(\dot{R} - 2\dot{e}_{lat})\dot{\theta}_R + (R - e_{lat})\ddot{\theta}_R]e_t + [(R - e_{lat})\dot{\theta}_R^2 + \ddot{e}_{lat}]e_n \tag{3.18}$$

$$= (\frac{dv_x}{dt} - v_y\dot{\theta})i + (\frac{dv_y}{dt} + v_x\dot{\theta})j \tag{3.19}$$

Acceleration can be similarly transformed by transform matrix as:

$$\begin{bmatrix} (\dot{R} - 2\dot{e}_{lat})\dot{\theta}_R + (R - e_{lat})\ddot{\theta}_R \\ (R - e_{lat})\dot{\theta}_R^2 + \ddot{e}_{lat} \end{bmatrix} = \begin{bmatrix} \cos(\theta - \theta_R) & -\sin(\theta - \theta_R) \\ \sin(\theta - \theta_R) & \cos(\theta - \theta_R) \end{bmatrix} \begin{bmatrix} \frac{dv_x}{dt} - v_y\dot{\theta} \\ \frac{dv_y}{dt} + v_x\dot{\theta} \end{bmatrix} \tag{3.20}$$

We make the following additional simplifying assumptions:

- Vehicle's longitudinal speed is a constant, say $v_x = V_0$.
- The difference between vehicle's heading angle, θ and road's heading angle, θ_R is relatively small, i.e. $|\theta - \theta_R| << 1$.
- The radius of curvature of the road is a piece-wise constant, i.e. $\frac{dR}{dt} \approx 0$.
- The radius of curvature of the road is significantly larger than lateral deviation, i.e. $R >> e_{lat}$.
- Ignore quadratic or higher order terms.

One can decompose the acceleration and velocity of the vehicle in road and vehicle's frame and relate them through a coordinate transformation; the above assumptions can be used to simplify

velocity (Eqn 3.16) and acceleration (Eqn 3.20) resulting in the following:

$$R\dot{\theta}_R \approx V_0, , \tag{3.21}$$

$$\dot{e}_{lat} \approx V_0(\theta - \theta_R) + v_y, \tag{3.22}$$

$$R\ddot{\theta}_R \approx 0, \tag{3.23}$$

$$R\dot{\theta}_R^2 + \ddot{e}_{lat} \approx \frac{dv_y}{dt} + V_0\dot{\theta}. \tag{3.24}$$

Note that equation (3.24) can be combined with equation (3.21):

$$\ddot{e}_{lat} \approx \frac{dv_y}{dt} + V_0(\dot{\theta} - \dot{\theta}_R). \tag{3.25}$$

Using equations (3.21), (3.22), (3.23) and (3.25), we can express vehicle dynamic equations (3.8) and (3.9) in terms of the states, $e_{lat}, \dot{e}_{lat}, \tilde{\theta}, \frac{d\tilde{\theta}}{dt}$ using the following matrices:

$$\mathbf{M} := \begin{bmatrix} m & 0 \\ 0 & I_z \end{bmatrix}, \tag{3.26}$$

$$\mathbf{C} := \begin{bmatrix} \frac{C_f + C_r}{V_0} & \frac{aC_f - bC_r}{V_0} \\ \frac{aC_f - bC_r}{V_0} & \frac{a^2C_f + b^2C_r}{V_0} \end{bmatrix}, \tag{3.27}$$

$$\mathbf{L} := \begin{bmatrix} 0 & -(C_f + C_r) \\ 0 & -(aC_f - bC_r) \end{bmatrix}, \tag{3.28}$$

$$\mathbf{B} := \begin{bmatrix} 1 \\ a \end{bmatrix}, \tag{3.29}$$

$$\mathbf{F} := \begin{bmatrix} mV_0^2 + (aC_f - bC_r) \\ a^2C_f + b^2C_r \end{bmatrix}, \tag{3.30}$$

$$\mathbf{x} := \begin{bmatrix} e_{lat} \\ \tilde{\theta} \end{bmatrix}. \tag{3.31}$$

Governing equations thus can be written as:

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{L}\mathbf{x} = \mathbf{B}C_f\delta_f - \mathbf{F}(\frac{1}{R}). \tag{3.32}$$

The above model assumes that the actuation is instantaneous, and sometimes, it may be worthwhile to consider a "more" realistic (singularly perturbed) actuation model, which limits feedback gains that can be chosen. The singularly perturbed model will include the dynamics of steering wheel, steering column, steering motor, and rack-pinion, as shown in Fig.3.3. In general, one can lump and associate a transfer function which takes the commanded steering angle, δ_c, as the input and δ_f as the output:

$$\delta_f(s) = H_a(s)\delta_c(s),$$

where $H_a(s)$ is the actuation transfer function. In this work, we use a second order model for steering actuation dynamics; the transfer function

$$H_a(s) = \frac{K_r}{J_w s^2 + b_w s + K_r},$$

where J_w is the steering wheel inertia, b_w is the torsional viscous damping coefficient ,and K_r is the torsional stiffness of the steering column. A natural question arises on the determination of parameters used in the vehicle model as they will be required for synthesizing both feedforward and feedback components. In the following subsections, we will discuss how we have obtained these parameters experimentally for our autonomous car (Lincoln MKZ platform from AutonomouStuff).

We first note that when δ and V_0 are constants, the resulting trajectory of the vehicle is a circle; its radius R is related to δ and V_0 through geometric parameter $L = a + b$ (distance between axles) and the steer gradient, K_{sg} as:

$$\delta = \frac{L}{R} + K_{sg}\frac{V_0^2}{R}. \tag{3.33}$$

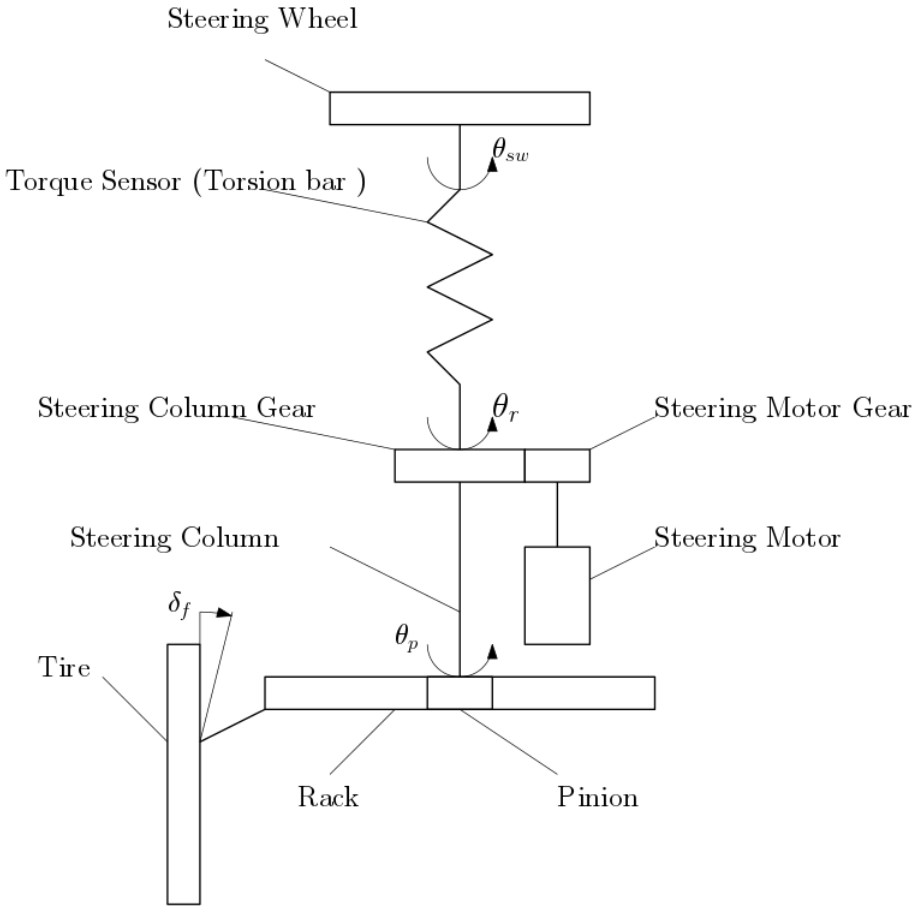

Figure 3.3: Illustration of Electrical Power Steering System

This relation can be determined by setting $\ddot{\mathbf{x}} = 0$ and $\dot{\mathbf{x}} = 0$; we exploit this relationship to estimate key vehicle parameters as in [35].

3.2.1 Estimation of Key Vehicle Parameters

The key vehicle parameters that are needed for vehicle control are: (a) Loads on the front and rear axles - this is necessary for the computation of steer gradient used for determining the feedforward steering input, (b) distance of vehicle's center of mass from the front and rear axles and (c) determination of vehicle's yaw inertia.

The vehicle was placed in a weighing station in different configurations:

1. the rear wheels were on the ground and the front wheels were placed on the scale to get the load on the front axle.

2. the rear wheels were placed on the scale while the front wheels were on the ground to get the rear axle load.

3. all the wheels were placed on the scale to get the total weight and check if the sum of the front and rear axle loads is close to the total weight, W.

Once the front/rear axle loads (W_f, W_r respectively) have been determined experimentally, the distances (a, b respectively) of the center of mass to the front and rear axles can be computed using the following equations representing the moment and force balance:

$$\begin{aligned} W_f \times a - W_r \times b &= 0, \quad W_f + W_r = W, \\ \Rightarrow a &= \frac{W_r}{W}, \quad b = \frac{W_f}{W}. \end{aligned}$$

Vehicle's yaw inertia about the center of mass can then be computed as:

$$I = a^2 \frac{W_f}{g} + b^2 \frac{W_r}{g}.$$

Below is the table of parameters we have determined experimentally. In this table, $m = \frac{W}{g}$.

Table 3.1: Parameters of Vehicle (Reprinted with permission from [1])

Parameter	**Description**	**Value**	**Unit**
m	vehicle total mass	1896	$[kg]$
W_f	vehicle front axle load	1052.32g	N
W_r	vehicle rear axial mass	843.68g	N
I_z	vehicle inertia	3803	$[kg.m^2]$
a	distance of c. g. to the front axle	1.2682	$[m]$
b	distance of c. g. to the rear axle	1.5818	$[m]$

3.2.2 Determination of Front and Rear Cornering Stiffness and Steer Gradient

The parameter K_{sg} can be estimated by carrying out different runs, where in each run corresponds to a specified V_0 and δ. Having collected the data from multiple runs, one can use linear regression to find K_{sg}. The general procedure can be illustrated as Fig.3.4.

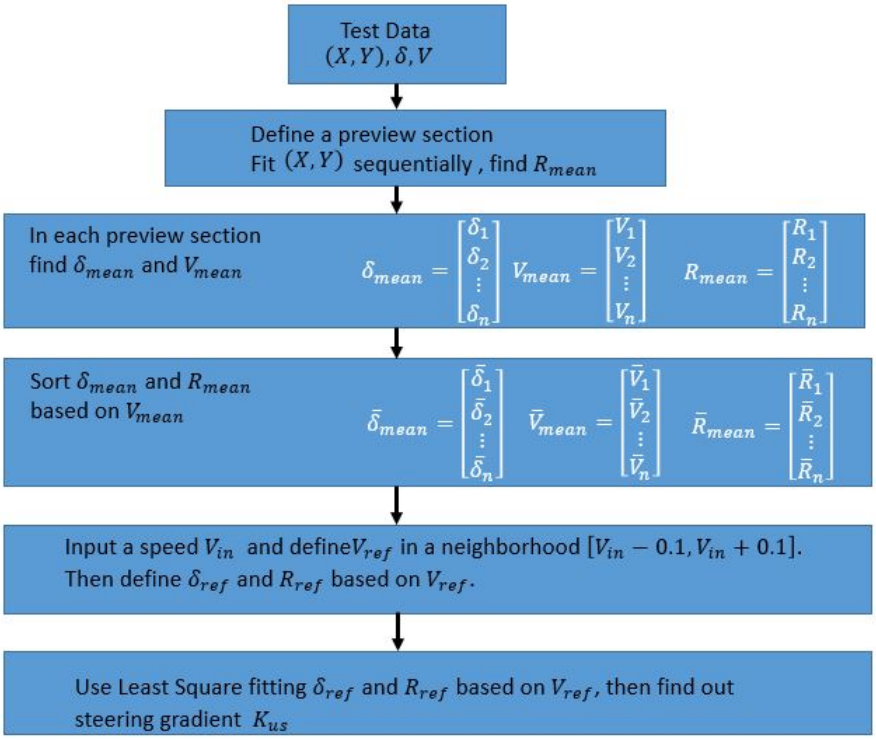

Figure 3.4: Flow chart for K_{us} determination

The collected test data for each run includes: the GPS coordinates of ego vehicle's trajectory, δ and V_0. The values of V_0 ranged from $5\ m/sec$ to $13.5\ m/sec$ (i.e., $11.25\ mph$ to $30\ mph$) with $0.5\ m/sec$ increments. The value of δ was set at $0.1061\ rad$. We did not perform tests with multiple steering angles because of safety and space considerations.

From each run, we constructed multiple data points, noting that the vehicle speed, V_0, or the steering angle, δ, may not remain constant throughout the run. For each run and for each point on the trajectory in the run, we associate a subset of the data from the run (much like the preview

data) corresponding to its future locations in the next 1 second until we come to 1 second before the run. For each data set, we find a mean speed V_{mean} and mean steering angle, δ_{mean} and find the corresponding radius of curvature, R_{mean} using the algorithm described in the earlier section. We sort the data sets in increasing order of speed, and they are plotted in the figure below:

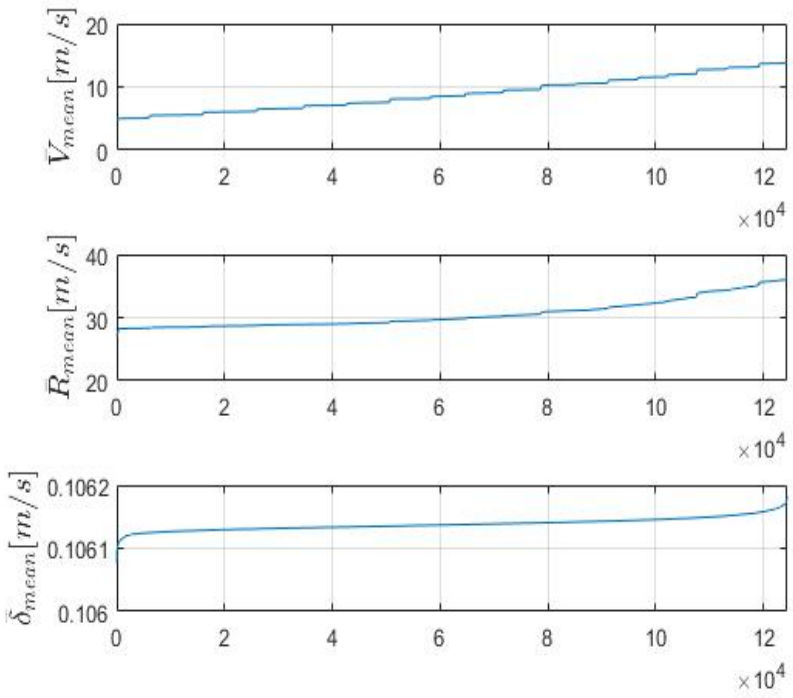

Figure 3.5: Experimental data for computing cornering stiffness and steer gradient (Reprinted with permission from [1])

In the plots, the x-axis represents the index of the data set. The data from the experiments can then be used to set up an overdetermined system of equations in K_{sg} using equation (3.33).

$$AK_{sg} = B.$$

We discretize the vehicle speed ,and for each data set with vehicle speed in the given discretiza-

tion, we then get the best estimate of K_{sg} as:

$$K_{sg} = (A^T A)^{-1} A^T B.$$

Doing so, would give us the variation of steer gradient, K_{sg}, with the speed of the vehicle, V_0, as shown in Fig. 3.6.

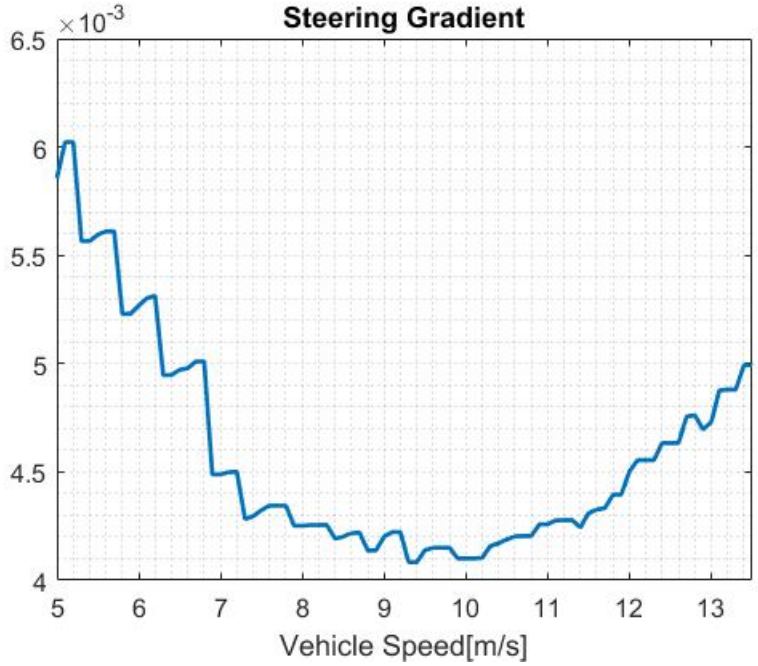

Figure 3.6: Steering gradient as a function of vehicle speed. (Reprinted with permission from [1])

When $V_0 < 5m/s$, we assume the steer gradient is negligible, as Ackerman steering angle component, $\frac{L}{R}$, of δ is dominating. For speeds $V_0 > 13.4m/s$, we take the value of K_{sg} that corresponds to $V_0 = 13.4\ m/sec$; due to limitation of space of RELLIS test track, this is the highest speed that we could achieve in a run within the limitations of space.

Fig. 3.6 was converted to a look-up table and implemented in the vehicle, $i.e.$, for a measured vehicle speed, steering gradient was determined from this look-up table for the computation of feedforward steering input.

For the synthesis of feedback stabilizing controllers, we need to determine the cornering stiffness coefficients C_f and C_r respectively. Once K_{sg} has been determined as a function of speed, we note the relationship between K_{sg} and C_f, C_r as:

$$K_{sg} = \frac{W_f}{C_f} - \frac{W_r}{C_r}.$$

For each discretization of the velocity, we can only get one equation in two variables, C_f and C_r; we can arrange the overdetermined system of equations as:

$$A_c \Psi = K_{sg},$$

where

$$\Psi = \begin{bmatrix} \frac{1}{C_f} \\ \frac{1}{C_r} \end{bmatrix},$$

and

$$A_c = \begin{bmatrix} W_f & -W_r \end{bmatrix}.$$

Correspondingly, we can determine the "least-norm" solution as:

$$\Psi = (A_c^T A_c)^{-1} A_c^T K_{sg}.$$

Plots of C_f and C_r as they vary with V_0 are shown below: We fit a fourth-order polynomial of the cornering stiffness coefficients for synthesizing feedback control; the order of fit seems to be adequate for approximating the plots of C_f and C_r with speed V as shown in the Fig. 3.7.

3.2.3 Identification of Steering Actuation Dynamics

Actuation system is assumed to be a linear, second order system, as in [36]:

$$H_a(s) = \frac{K_r}{J_w s^2 + b_w s + K_r} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}. \tag{3.34}$$

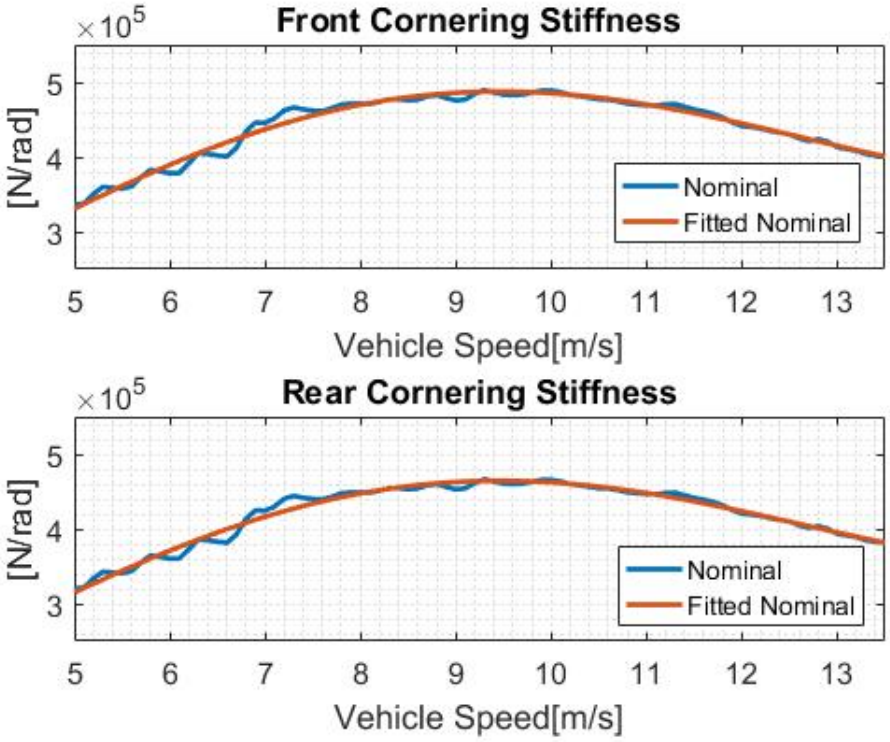

Figure 3.7: Cornering stiffness variation with speed (Reprinted with permission from [1])

The idea of determining the parameters w_n, ζ is to provide a unit step steering command and observe the response. From the response, one can determine ζ and w_n using "log-dec" method [37].

The response of the steering system to a steering command of $0.106\ rad$ is shown in figure 3.8. The first peak (X_1, T_1) and second peak(X_2, T_2) in this figure are used for computing ζ and w_n. The "log-dec", $\eta = ln(\frac{X_1 - SC}{X_2 - SC})$. The time elapsed between the peaks is $T_d = T_2 - T_1$; correspondingly, the damped natural frequency is $\omega_d = \frac{2\pi}{T_d}$, and the damping ratio ζ and natural frequency can be calculated as:

$$\zeta = \frac{\eta}{\sqrt{(2\pi)^2 + \eta^2}}, \quad \omega_n = \frac{\omega_d}{\sqrt{1 - \zeta^2}}. \tag{3.35}$$

We found that $\zeta = 0.4056$, $\omega_n = 21.4813\ rad/s$ from the data and we use these values for design

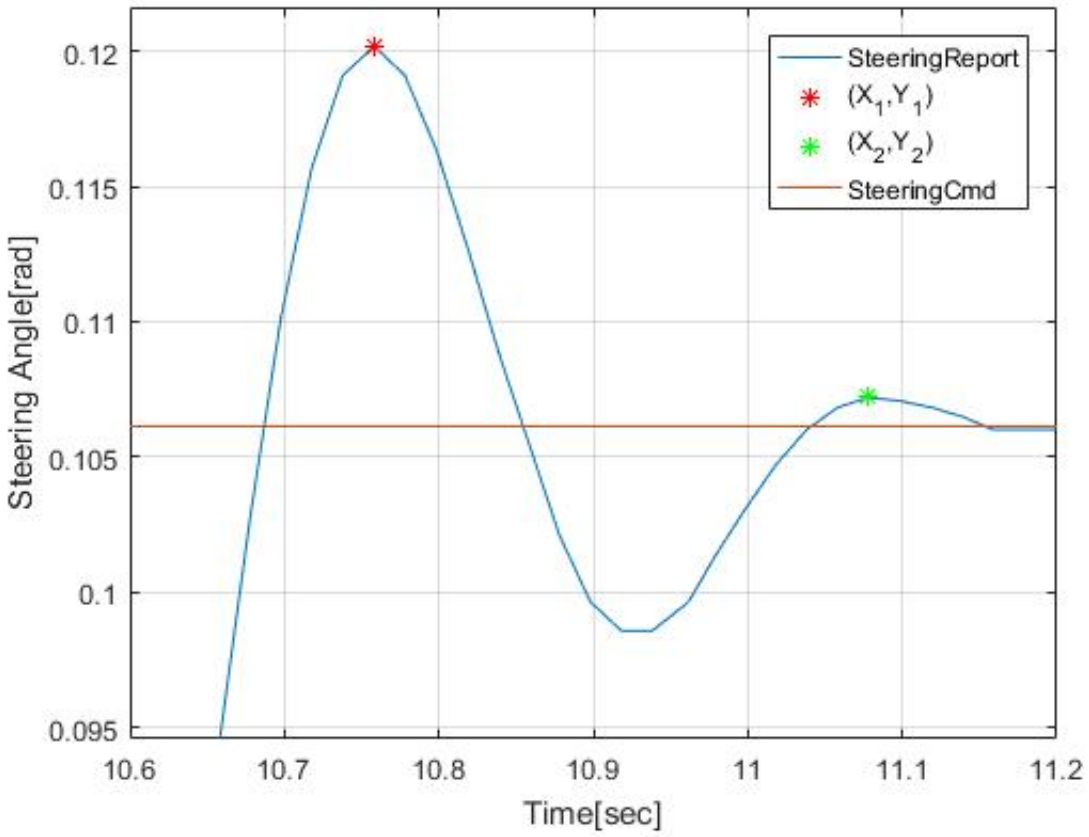

Figure 3.8: Step response of steering actuation system (Reprinted with permission from [1])

and implementation of lateral controller.

4. LATERAL CONTROLLER SYNTHESIS FOR AN AUTONOMOUS VEHICLE CONTROL*

In this chapter, we present the lateral controller for an autonomous vehicle. This controller is designed based on the simplified lateral dynamic model, which has been discussed in Chapter 3. The stability issue will also be discussed in the latter sections of this chapter. Furthermore, a series of experiments will be designed to corroborate the effectiveness of designed lateral controllers.

A Lateral controller for tracking a trajectory can be decomposed into two parts: a feedforward part and a feedback part. A feed forward part is aimed at estimating the steering control command while a feedback part is designed to ensure the decay of feedback error signals. The final desired steering command(δ_c) is the summation of feedforward(δ_{ff}) and feedback(δ_{fb}),i.e.,

$$\delta_c = \delta_{ff} + \delta_{fb}.$$

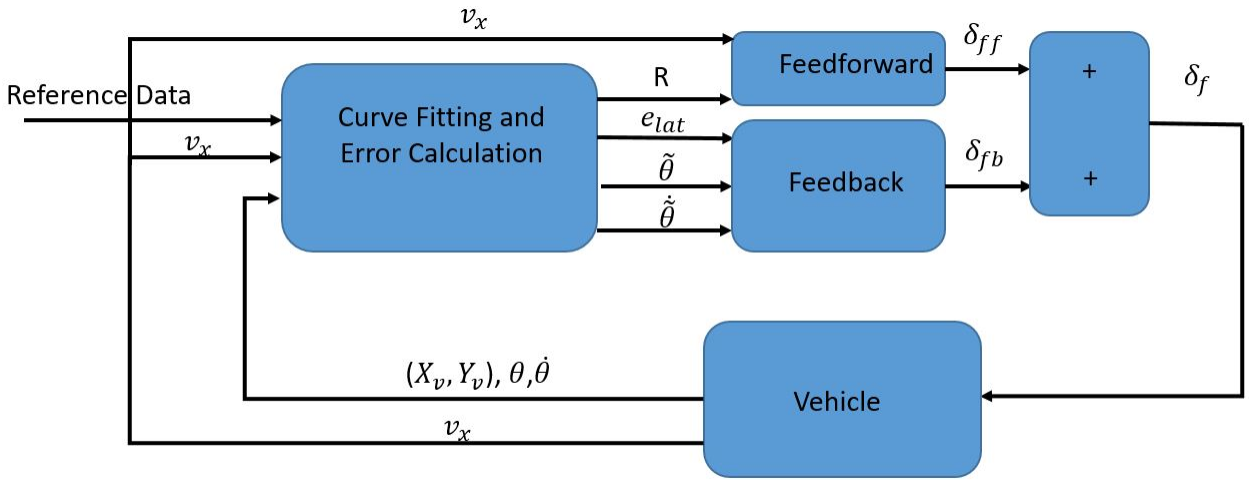

Figure 4.1: Structure of Controller

*Parts of this chapter are reprinted with permission from [1].

As shown in Fig.4.1, the feedforward controller requires road curvature (R), and vehicle's longitudinal velocity (v_x). The feedback controller requires lateral error(e_{lat}), yaw error ($\tilde{\theta}$) and yaw rate error ($\dot{\tilde{\theta}}$). The signal v_x can be measured while the computation of all other signals from available measurements and computed trajectory has been discussed in Chapter 2. Clearly, v_x can be obtained from vehicle directly. In the following sections, we will outline how these controllers are designed:

4.1 Feedforward Controller

The feedforward controller essentially provides the steering input that would keep the vehicle on the desired circular trajectory without any feedback if the initial conditions were to be appropriate and if the disturbance inputs on the vehicle are absent. The feedforward steering controller predicts steering inputs and improves the system performance because it will send out steering command before the vehicle deviates from its path[38]. Moreover, the feedforward steering controller may also sustain a reasonable passenger comfort[39]. One may view the feedforward steering input, δ_{ff}, as that steering input, which results in the vehicle's trajectory being a circle of radius R when it is traveling at a longitudinal speed of V_0. The derivation of feedforward steering input is as shown as follows: In a steady-state turn, the vehicle equation of motion Eqn.3.8 becomes:

$$m\frac{v_y^2}{R} = F_{cf} + F_{cr}, \tag{4.1}$$

$$0 = aF_{cf} - bF_{cf}. \tag{4.2}$$

Then cornering forces on front and rear tires can be represented as:

$$F_f = m\frac{v_x^2}{R}\frac{b}{L} \tag{4.3}$$

$$F_r = m\frac{v_x^2}{R}\frac{a}{L} \tag{4.4}$$

Substituting Eqn.4.4 and Eqn.4.4 into Eqn.3.5 and Eqn.3.6, we got:

$$\alpha_f = m\frac{v_x^2}{R}\frac{b}{L}\frac{1}{C_f} \tag{4.5}$$

$$\alpha_r = m\frac{v_x^2}{R}\frac{a}{L}\frac{1}{C_r} \tag{4.6}$$

Recall the relationship between steering angle(δ_f) and tire slip angles(α_r,α_f) in Eqn. 3.7. Clearly, the corresponding feedforward steering input is:

$$\delta_{ff} = \frac{a+b}{R} + (\frac{mb}{LC_f} + \frac{ma}{LC_r})\frac{V_0^2}{R} = \frac{a+b}{R} + K_{sg}\frac{V_0^2}{R}. \tag{4.7}$$

Note that for understeered vehicles, $K_{sg} > 0$ and the open-loop handling dynamics will be unconditionally stable [40, 41].

4.2 Feedback Controller Design

Feedback control is based on available information, namely, $e_{lat}, \tilde{\theta}$ and $\dot{\tilde{\theta}}$; the lateral velocity information is not readily available; hence, we seek a "fixed structured" control law of the form:

$$\delta_{fb} = -k_e e_{lat} - k_\theta \tilde{\theta} - k_w \dot{\tilde{\theta}},$$

where the gains k_e, k_θ and k_w need to be determined. Higher order controllers can suffer from the problem of fragility [42] and often require more parameters to tune.

From the governing equations, the transfer functions relating $e_{lat}(s)$ and $\tilde{\theta}(s)$ to $\delta_f(s)$ can be

related through the open-loop characteristic polynomial, $\Delta_o(s)$ as:

$$\begin{aligned}
\Delta_o(s) &= s^2(mIs^2 + \frac{(I+ma^2)C_f + (I+mb^2)C_r}{V_0}s \\
&\quad + \frac{(a+b)^2 C_f C_r}{V_0^2} - m(aC_f - bC_r)), \qquad (4.8) \\
\frac{e_{lat}(s)}{\delta_f(s)} &= \frac{C_f}{\Delta_o(s)}(Is^2 + \frac{b(a+b)C_r}{V_0}s + (a+b)C_r), \qquad (4.9) \\
\frac{\tilde{\theta}(s)}{\delta_f(s)} &= \frac{C_f}{\Delta_o(s)}(mas^2 + \frac{(a+b)C_r}{V_0}s). \qquad (4.10)
\end{aligned}$$

Since

$$\begin{aligned}
\delta_f &= H_a(s)\delta_c(s) \\
&= -H_a(s)(k_e e_{lat}(s) + (k_\theta + k_w s)\tilde{\theta}(s)), \qquad (4.11)
\end{aligned}$$

it follows that the closed-loop characteristic equation can be expressed as:

$$\begin{aligned}
&\Delta_o(s) + H_a(s)k_e C_f(Is^2 + \frac{b(a+b)C_r}{V_0}s + (a+b)C_r) \\
&\quad + H_a(s)(k_\theta + k_w s)C_f(mas^2 + \frac{(a+b)C_r}{V_0}s) = 0. \qquad (4.12)
\end{aligned}$$

For the second-order steering actuation model, it follows that the characteristic polynomial may be expressed in terms of the control parameter vector $\mathbf{K} = (k_e, k_\theta, k_w)$ as:

$$\begin{aligned}
\Delta(s;\mathbf{K}) &= (s^2 + 2\zeta\omega_n s + \omega_n^2)\Delta_o(s) + k_e\omega_n^2 C_f(Is^2 + \frac{b(a+b)C_r}{V_0}s + (a+b)C_r) \\
&\quad + (k_\theta + k_w s)\omega_n^2 C_f(mas^2 + \frac{(a+b)C_r}{V_0}s). \qquad (4.13)
\end{aligned}$$

Note that $\Delta(s, \mathbf{K}) = A_6 s^6 + A_5 s^5 + A_4 s^4 + A_3 s^3 + A_2 s^2 + A_1 s + A_0$, where

$$
\begin{aligned}
A_6 &= \frac{Im}{\omega_n^2}, \\
A_5 &= \frac{2Im\zeta}{\omega_n} + \frac{C_f(I + a^2 m) + C_r(I + b^2 m)}{V_0 \omega_n^2}, \\
A_4 &= \frac{2\zeta(C_f(I + a^2 m) + C_r(I + b^2 m))}{V_0 \omega_n} \\
&+ (\frac{(a+b)^2 C_f C_r}{V_0^2 \omega_n^2} - m(aC_f - bC_r)) + mI, \\
A_3 &= (\frac{2\zeta((a+b)^2 C_f C_r)}{V_0^2 \omega_n} - m(aC_f - bC_r)) \\
&+ \frac{C_f(I + a^2 m) + C_r(I + b^2 m)}{V_0} + C_f m a k_\omega, \\
A_2 &= (\frac{(a+b)^2 C_f C_r}{V_0^2} - m(aC_f - bC_r)) \\
&+ (C_f I k_e + C_f C_r \frac{(a+b)}{V_0} k_\omega + maC_f k_\theta), \\
A_1 &= C_f C_r \frac{(a+b)}{V_0}(bk_e + k_\theta), \\
A_0 &= C_f C_r (a+b) k_e,
\end{aligned}
$$

where ζ, ω_n are the damping ratio and natural frequency of steering actuation. They can be found out in our previous work[1] as well as previous Chapter 2. The detailed derivation of $\Delta(s; \mathbf{K})$ can be seen in Appendix A.

4.2.1 Stabilizing Gain Set Generation

A natural problem regarding the stability of tracking is to determine the set of control gains, $\mathbf{K}$, so that the closed-loop characteristic polynomial (4.13) is Hurwitz. An advantage of determining the entire set is that the set can be pruned to accommodate additional performance criteria.

There are many ways to construct the set of controllers [43, 22, 44, 45] in the parameter space. We adopt the D-decomposition approach of [22, 45] here. This method has been used as early as the 1970s by Siljak in his work [46] and subsequently by Bhattacharyya and his collaborators [22],[43], in the construction of the set of stabilizing PID and fixed structure controllers. Here we

notice the error signals for feedback controller design are lateral error(e_{lat}), yaw error($\tilde{\theta}$), and yaw rate error($\dot{\tilde{\theta}}$). However, there is no lateral error rate($\dot{e}_{lat}$) available. One may adopt an observer-based controller design that estimates the lateral error rate; however, we have chosen a more direct approach of synthesizing a static output feedback controller with available information. This will ensure that the number of parameters to tune is small (three). There are also some popular estimation techniques such as KF, or EKF to estimate the non-observable states. However, the drawbacks are obvious: First, from the aspect of EKF itself, it estimation can be degraded by the accumulation rounding errors in addition to the non-Gaussian noise[47]. Also, this kind of Dynamic Output Feedback scheme has more control parameters to tune; in KF or EKF, we need to specify extra gains: because of separation theorem, we must have both controller and observer gains; moreover, with dynamic output feedback, the closed-loop can be fragile [42]. If a static output feedback controller can be designed, it is simpler to implement and tune as we only need three gains. Moreover, based on this D-decomposition method, one can construct the set of stabilizing gains and visualize the set!

Primarily, this approach relies on the continuous dependence of roots of a polynomial on its coefficients for every regular perturbation. The approach then involves decomposing the parameter space into disjoint signature-invariant regions by identifying their boundaries. The boundaries can be obtained by

- Determining the set of $\mathbf{K}$ for which $\Delta(0, \mathbf{K}) = 0$
- Determining the set of $\mathbf{K}$ for which $\Delta(jw, \mathbf{K}) = 0$ for some w.

Once the parameter space is partitioned, one can then sample every partition to determine the partition corresponding to $\Delta(s; \mathbf{K})$ being Hurwitz. Here we notice that there are three parameters, k_e, k_θ and k_ω. For the synthesis of a stabilizing set, we fix the k_e for some constant value and find out the signature-invariant regions in k_e-k_ω plane.

The k_θ and k_ω can be represented as a function of ω by D-decomposition as:

$$k_\theta = \frac{(D_{4o}(j\omega)\,(-D_{1e}(j\omega) - keD_{2e}(j\omega)) + D_{4e}(j\omega)\,(D_{1o}(j\omega) + k_eD_{2o}(j\omega)))}{|A_{\theta\omega}|} \quad (4.14)$$

$$k_\omega = \frac{(D_{3o}(j\omega)(D_{1e}(j\omega) + keD_{2e}(j\omega)) + D_{3e}(j\omega)(-D_{1o}(j\omega) - k_eD_{2o}(j\omega)))}{|A_{\theta\omega}|} \quad (4.15)$$

where:

$$\begin{aligned}
D_{1e}(j\omega) &= -\frac{\alpha_5}{\omega_n^2}w^6 + (\frac{\alpha_1}{\omega_n^2} + \frac{\alpha_4}{\omega_n} + \alpha_5)w^4 - (\alpha_1)w^2, \\
D_{1o}(j\omega) &= (\frac{\alpha_4}{\omega} + \frac{2\zeta\alpha_5}{\omega_n})w^4 - (\frac{2\zeta\alpha_1}{\omega} + \alpha_4)w^2, \\
D_{2e}(j\omega) &= -(\alpha_2)w^2 + \alpha_0, D_{2o}(j\omega) = (\frac{\alpha_0}{v_0}b) \\
D_{3e}(j\omega) &= -(\alpha_3)w^2, D_{3o} = \frac{\alpha_0}{V_0} \\
D_{4e}(j\omega) &= -\frac{\alpha_0}{V_0}w^2, D_{4o} = -(\alpha_1)w^2
\end{aligned}$$

From equations above, k_θ and k_θ can be plotted. An example is displayed when $k_e = 0.06$, at $V_0 = 15m/s$, the relevant C_f and C_r have been gathered from Fig. 3.7 and the controller set is illustrated in Fig. 4.2. The whole $k_\theta - k_\omega$ plane is partitioned by boundaries conditions above. The whole plane is decomposed by 3 disjoint signature invariant regions. By checking the signature of the characteristic polynomial for a sampled controller in each region, only gain set (k_θ, k_ω) from inner contour of curve, i.e. region 2 stabilized system. The detailed procedure can be found out in Appendix B.

We can increase k_e and similarly get related stabilizing regions. Then several layers of stabilizing regions can be determined, and a stabling set can be constructed. The controller gains can then be chosen from this stabilizing set.

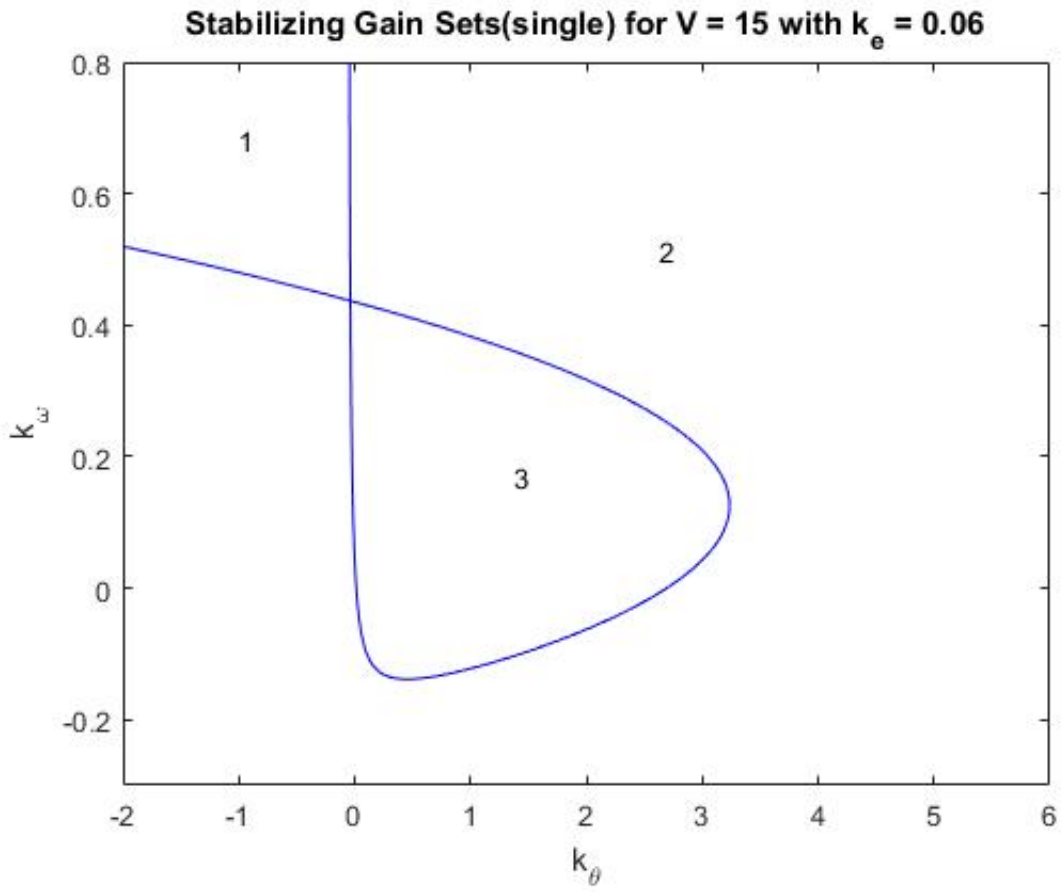

Figure 4.2: Single layer of Stabilizing Gain Set for $V = 15m/s$ with $k_e = 0.06$.

4.2.2 Closed-loop Stability with varying longitudinal velocity

Let $\mathbf{Q}, \mathbf{G}, \mathbf{H}$ be appropriate matrices (in controllable canonical form) so that the the transfer function, $H_a(s)$, from steering command, δ_c, to steering angle, δ_f, has the following minimal realization:

$$\dot{\eta} = \mathbf{Q}\eta + \mathbf{G}\delta_c, \quad \delta_f = \mathbf{H}\eta.$$

Note that the dynamics of the vehicle can be expressed as

$$\begin{pmatrix} \dot{\mathbf{x}} \\ \ddot{\mathbf{x}} \end{pmatrix} = \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{L} & -\gamma\mathbf{C}_0 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \dot{\mathbf{x}} \end{pmatrix} + \begin{pmatrix} 0 \\ \mathbf{B} \end{pmatrix} C_f\delta_f - \begin{pmatrix} 0 \\ \mathbf{F} \end{pmatrix} \frac{1}{R}. \tag{4.16}$$

Let $\mathbf{K}_p := [k_e \;\; k_\theta]$ and $\mathbf{K}_v := [0 \;\; k_\omega]$. Since

$$\delta_c = -\mathbf{K}_p\mathbf{x} - \mathbf{K}_v\dot{\mathbf{x}}, \tag{4.17}$$

the closed loop dynamics is then given by:

$$\underbrace{\begin{pmatrix}\dot{\mathbf{x}}\\ \ddot{\mathbf{x}}\\ \dot{\eta}\end{pmatrix}}_{\dot{\mathbf{z}}} = \underbrace{\begin{pmatrix}\mathbf{0} & \mathbf{I} & \mathbf{0}\\ -\mathbf{L} & -\gamma\mathbf{C}_0 & \mathbf{B}\mathbf{H}C_f\\ -\mathbf{G}\mathbf{K}_p & -\mathbf{G}\mathbf{K}_v & \mathbf{Q}\end{pmatrix}}_{\mathbf{A}(\gamma)} \underbrace{\begin{pmatrix}\mathbf{x}\\ \dot{\mathbf{x}}\\ \eta\end{pmatrix}}_{\mathbf{z}} + \underbrace{\begin{pmatrix}0\\ -\mathbf{F}\\ 0\end{pmatrix}}_{\tilde{\mathbf{F}}} \frac{1}{R}. \tag{4.18}$$

Once the controller gains are fixed, i.e., $\mathbf{K}_p, \mathbf{K}_v$ are known, closed loop stability can be examined by formally considering a linear system

$$\dot{\mathbf{z}} = \mathbf{A}(t)\mathbf{z}(t). \tag{4.19}$$

Problem (9.31) of [48] can be employed to establish closed loop stability; it is stated here for completeness:

Lemma 1: Suppose the linear system:

$$\mathbf{z} = \mathbf{A}(t)\mathbf{z}(t) \tag{4.20}$$

satisfies the following conditions:

- For some $k > 0$, $\|A(t)\| \leq k$ for all $t \geq 0$,
- For some $\sigma > 0$, every eigenvalue of $\mathbf{A}(t)$ has a real part less than $-\sigma$ for every $t \geq 0$, and
- $\|\dot{\mathbf{A}}(t)\|$ is square integrable, i.e., for some $\rho > 0$,

$$\int_0^\infty \|\dot{\mathbf{A}}(t)\|^2 dt \leq \rho. \tag{4.21}$$

Then, the solution $\mathbf{z} = 0$ is exponentially stable.

Proof: It can be found in the solutions manual [49].

Closed loop stability with varying longitudinal speed has not been considered in the literature; the following result provides the connection between "frozen parameter" control synthesis and stability of linear time varying system at hand with varying longitudinal speed:

Proposition 1: Consider the linear time varying system

$$\dot{\mathbf{z}} = \mathbf{A}(\gamma(t))\mathbf{z}$$

given above. Suppose K_p, K_v have been chosen to satisfy the following conditions:

(a) The real part of eigenvalues of $\mathbf{A}(\gamma)$ are less than $-\sigma$ for every $\gamma \in [\frac{1}{V_{max}}, \frac{1}{V_{min}}]$.

(b) The longitudinal acceleration of the vehicle is square integrable, i.e, $\int_0^\infty \dot{v}_x^2(\tau)d\tau < \infty$.

Then, the equilibrium $\mathbf{z} = 0$ is exponentially stable.

Proof: We will use the proof of Problem (9.31) of [48] to arrive at the result. Essentially, it requires that three conditions be satisfied:

- Note that $\gamma(t) = \frac{1}{v_x(t)}$. Hence, $\|\mathbf{A}(\gamma(\mathbf{t}))\|$ is bounded for all t or equivalently, it is bounded for all $\gamma \in [\frac{1}{V_{max}}, \frac{1}{V_{min}}]$; in this case, it is easy to verify that this condition holds.

- The real part of the eigenvalues of $\mathbf{A}(\gamma(t))$ are less than $-\sigma$ for some $\sigma > 0$; again this condition holds because the gains $\mathbf{K}_p, \mathbf{K}_v$ have been chosen to be in the set of stabilizing controllers that render $\mathbf{A}(\gamma(t))$ Hurwitz for every $\gamma \in [\frac{1}{V_{max}}, \frac{1}{V_{min}}]$.

- The third condition requires $\int_0^\infty \|\dot{A}(\gamma(\tau))\|^2 d\tau < \infty$; in our case, it is easy to see that $\mathbf{A}(\gamma)$ is linear in γ and for some appropriate *constant* matrices $\mathbf{A}_0$ and $\mathbf{A}_1$, we can express $\mathbf{A}(\gamma) = \mathbf{A}_0 + \gamma\mathbf{A}_1$. Consequently,

$$\dot{\mathbf{A}}(\gamma) = \dot{\gamma}\mathbf{A}_1, \tag{4.22}$$

implying that $\dot{\mathbf{A}}(\gamma)$ is square integrable if $\dot{\gamma}$ is square integrable, i.e., $\dot{\gamma} \in \mathcal{L}_2$ as γ is bounded. However,

$$\begin{aligned} \dot{\gamma} &= -\frac{1}{v_x^2}\dot{v}_x \\ \implies \int_0^\infty |\dot{\gamma}(\tau)|^2 d\tau &= \int_0^\infty \frac{1}{v_x^4(\tau)} |\dot{v}_x(\tau)|^2 d\tau \\ &\leq \frac{1}{V_{min}^4} \int_0^\infty |\dot{v}_x(\tau)|^2 d\tau. \end{aligned} \tag{4.23}$$

By virtue of $\dot{v}_x$ being square integrable, the third condition is also satisfied. Hence, the closed loop time varying system is exponentially stable by Lemma 1.

The assumption that the longitudinal acceleration is square integrable is very reasonable because often acceleration/deceleration maneuvers are associated with a finite change in longitudinal speed; the finite time duration of acceleration maneuvers renders this condition readily satisfied.

4.3 Numerical and Experimental Results

4.3.1 Fitting a Circular Arc

The effectiveness of estimating the trajectory using the "least-square" approach in Section 2 depends on the noise in the sampled values of GPS. To examine this issue, we consider a preview for a vehicle is traversing a circular trajectory of radius $R \in \{100,\ 200\}$ meters at a speed of $25m/s$; we sample it at $20\ Hz$ and provide a preview of 20 samples of the trajectory to the vehicle (A preview time of 1 second translates to a preview distance of $25\ m$). We induce noise in the samples and want to determine the noise level at which the error in reconstructing the trajectory exceeds $25\ cm$; the error in reconstructing the trajectory can be calculated via the Euclidean distance between the estimated value of the sample and its true value. For each level of noise, we conduct 100 numerical simulations, each corresponding to one realization of the random noise sequence of twenty samples. Figures shown below show demonstrate trajectory reconstruction with a noise in GPS of $15\ cm$ corresponding to two trajectories of radii $100\ m$ and $200\ m$,respectively. While the cross-track error in both cases is acceptable (<20 cm), the longitudinal error in the estimation of

position seems to increase with the noise level (from about 25 cm to 5 m depending on the noise level). It appears that the longitudinal error in estimation can be corrected with other sensing and communication mechanisms.

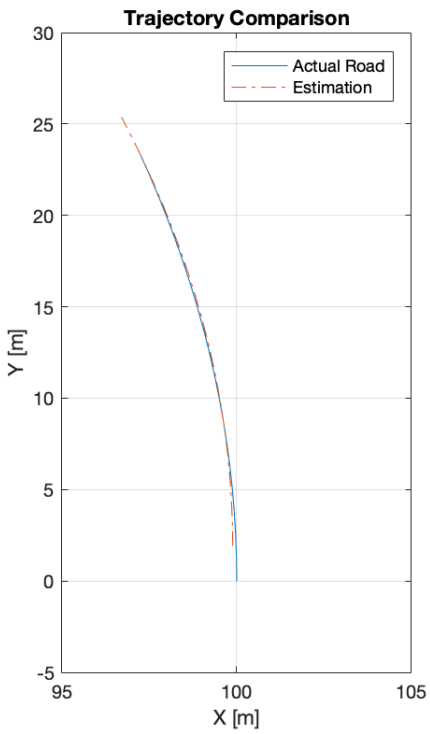

Figure 4.3: Trajectory Comparison at $R = 100\ m$

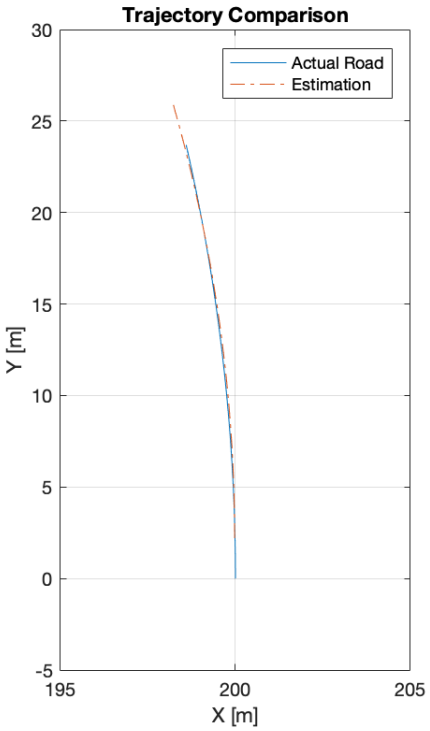

Figure 4.4: Trajectory Comparison at $R = 200\ m$

4.3.2 Construction of Stabilizing Set of Feedback Gains

In order to construct the set of stabilizing feedback gains, we used the parameters from Table 3.1. Since the stabilizing set (which is non-convex) depends on V_0, we find the sets of all the stabilizing controllers for a range of speeds in $\{10, 20, 30, 40, 50, 60, 67\}\ mph$ and choose a controller in the intersection of all these stabilizing sets. The actual gains using this procedure are $(k_e, k_\theta, k_\omega) = (0.06, 0.96, 0.08)$ and is an interior point in all the stabilizing sets. The stabilizing set corresponding to $V_0 \in \{10, 20, 30, 40, 50, 60, 70\}\ mph$ is shown in Fig. 4.5, 4.6,4.7,4.8,4.9,4.10

and 4.11.

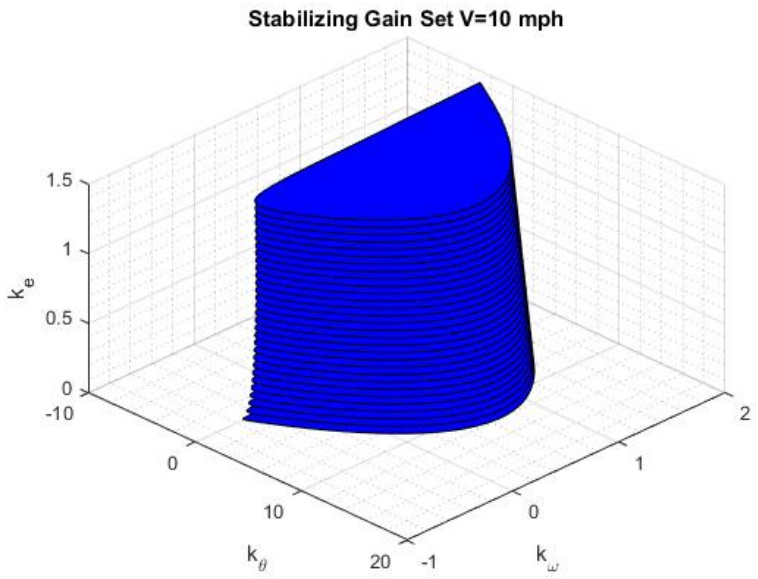

Figure 4.5: Stabilizing set of controllers for 10 mph

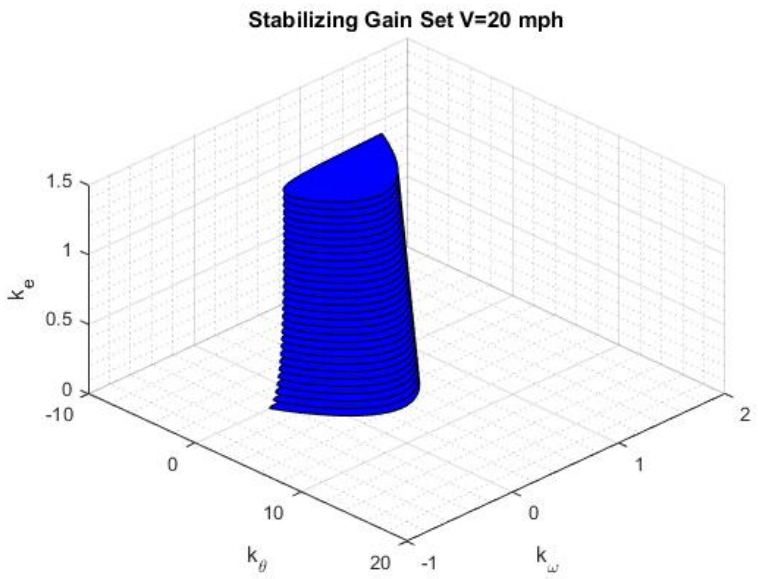

Figure 4.6: Stabilizing set of controllers for 20 mph

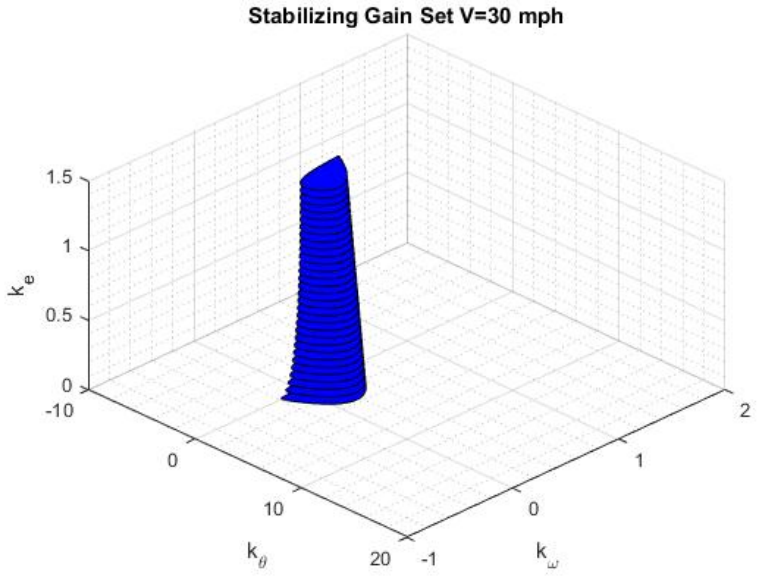

Figure 4.7: Stabilizing set of controllers for 30 mph

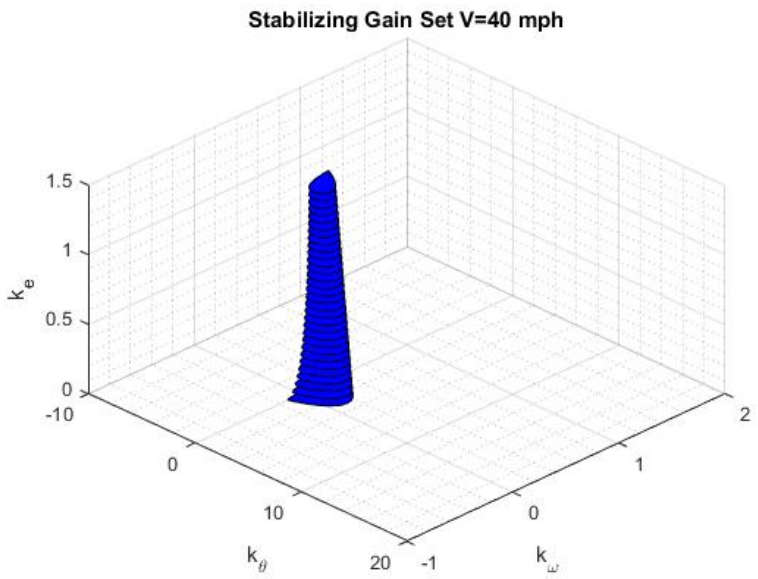

Figure 4.8: Stabilizing set of controllers for 40 mph

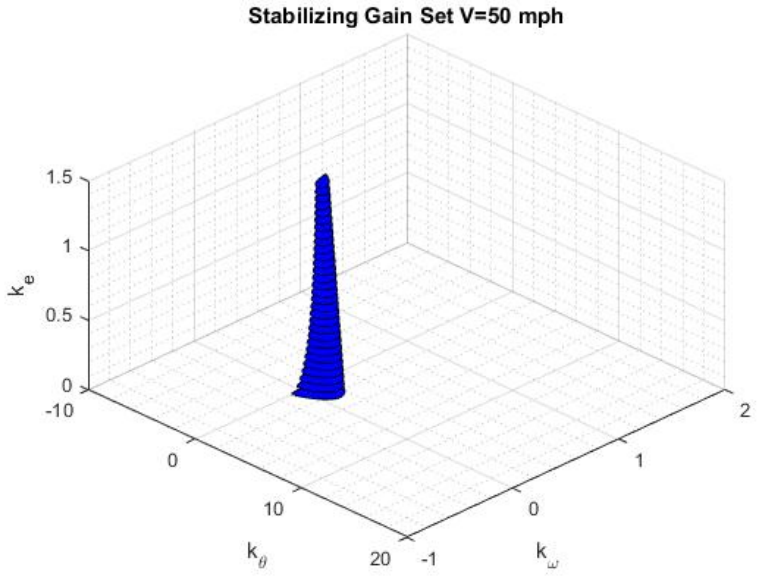

Figure 4.9: Stabilizing set of controllers for 50 mph

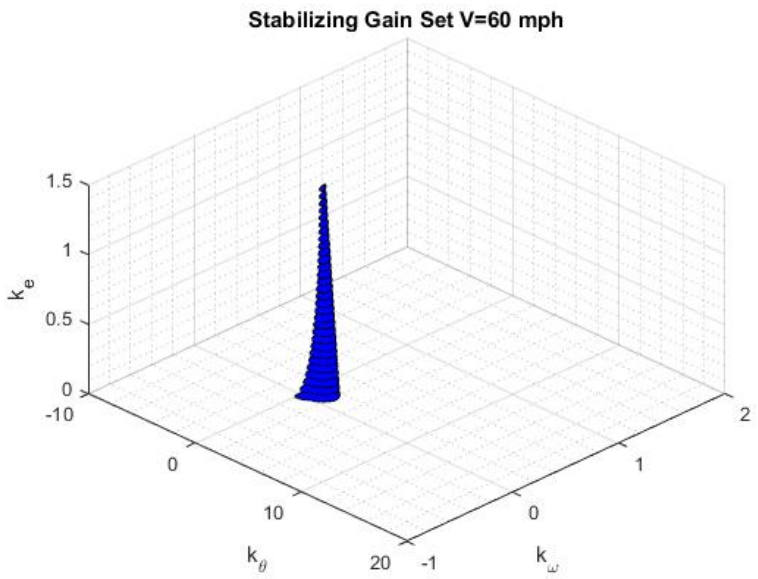

Figure 4.10: Stabilizing set of controllers for 60 mph

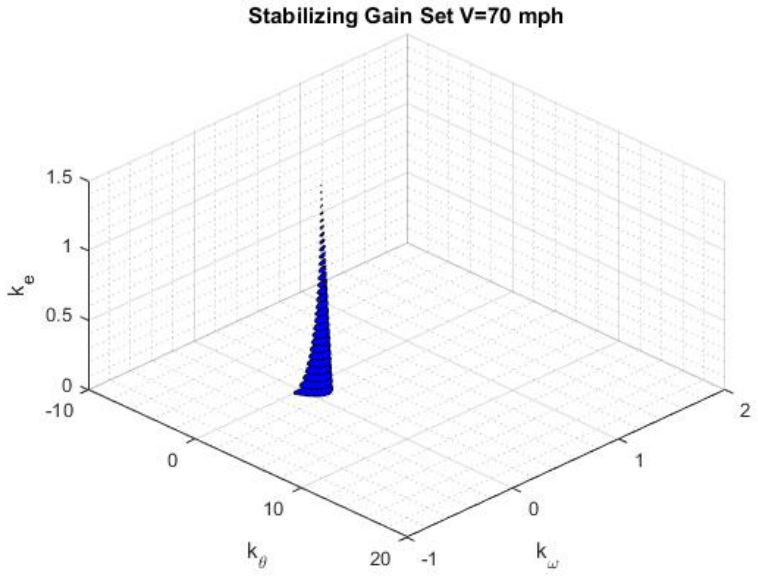

Figure 4.11: Stabilizing set of controllers for 67 mph

From the figures above, the stabilizing gain set space volume is shrinking: with higher operation speed, the system is more natural to be unstable. Also, the cross-section area is shrinking with k_e value increasing, which indicates that the controller is very sensitive to k_e. Hence, there are two rules that need to be followed:

- Pick gain sets at stabilizing space highest operating speed.
- Pick k_e as low as possible, which provides a larger gain sets for (k_θ and k_ω)

The actual gains using this procedure are $(k_e, k_\theta, k_\omega) = (0.06, 0.96, 0.08)$ and is an interior point in stabilizing set for $V = 70\ mph$. It is also inside the stabilizing set of controllers for all speeds until $V = 70\ \ mph$.

4.3.3 Experimental Setup

The algorithms were implemented on a Lincoln MKZ car. The vehicle was equipped with a drive-by-wire system for autonomous steering, throttle and brake control. Data concerning the vehicle states (position, yaw angle, and yaw rate) were obtained from an onboard integrated GPS system and IMU unit. The controller operates at 50 Hz.

4.3.3.1 Lane Keeping Test

In this test, the vehicle follows a 2-mile long loop at the TAMU's RELLIS campus with its pre-recorded GPS coordinates (see Fig. 4.12); there are four turns, and the vehicle followed this path in the clockwise direction from the starting point (shown using a small circle in the figure). The turning radii at the four corners along the path are approximately: 42m, 81m, 76m ,and 114m. The target speed is 13.4 m/s. For driver safety, we limited the vehicle speed so that the lateral acceleration does not exceed $1.5m/s^2$ during the turns; this implies that the turning speed can at most be $v_{turn} = \sqrt{a_{lim}R}$.

With the designed controller, the lateral (cross-track) error is shown in Fig. 4.13. There are four pairs of red vertical bars; each pair denotes the time interval when the vehicle was taking a turn around the corner. The maximum lateral error obtained through this entire experiment was

less than $0.45\ m$. The maximum error occurs during the first turn, which is the sharpest one of all the four turns. On an average, the cross-track error settled to around 0.1m after each of the turns. The corresponding vehicle speed, yaw error, yaw rate error ,and the feedback (FB) and the feedforward (FF) control commands can be seen in Figs. 4.14, 4.15, 4.16 and 4.17. Please note that feedforward command is triggered only when it can be determined that the preview section data corresponds to a curved segment. As shown in Fig. 4.17, the feedforward control input is only supplied while turning.

4.3.3.2 Lane Change Test

We conducted multiple lane change tests to corroborate the controller's performance. The procedure used in each of the tests is as follows: In the first run, the vehicle was run in the manual mode. It starts from rest, accelerates to $30\ m/sec$, then performs a double lane change maneuver (simulating a maneuver where the vehicle turns left to go to the next lane and then right to get back onto the original lane). The corresponding reference path is shown in Fig. 4.18. We then collected the GPS data corresponding to this run for simulating the data that a following connected and autonomous vehicle would receive via communication. In the next run, the vehicle was run in autonomous mode, with the pre-recorded data from the previous run providing limited preview information. To simulate this, the autonomous vehicle could only use the GPS waypoint information that is at most $30\ ms$ ahead. In the lane-change tests, a maximum lateral error of about $0.3\ m$ was obtained and occurred at the end of each lane change maneuver. The corresponding vehicle speed, yaw error, yaw rate error ,and feedback and feedforward control commands for one of the tests can be seen in Figs. 4.20, 4.21, 4.22 and 4.23. Similar to lane-keeping tests, the feedforward command is triggered only when it can be determined that the preview section data corresponds to a curved segment (when lane change actually occurs).

To corroborate the performance of the controller for non-zero initial errors, we conduct seven test runs with different initial errors. The lateral error corresponding to multiple runs is shown in Fig. 4.24. In these runs, the autonomous vehicle was tracking the same track repeatedly during each run. The maximum lateral error observed was still less than $0.35\ m$ during each of the multiple

runs. From table 4.1, we can notice that the maximum of lateral error is 0.3358 m. The maximum of the average lateral error is 9 cm, and the minimum of the the average lateral error is 6.89 cm. Considering that the initial position of the vehicle for each run is not identical, the performance of the controller is stable, and consistent.

Table 4.1: Maximum and Average Lateral Error for Multiple-runs of Lane Change test (Reprinted with permission from [1])

Test Number	**Maximum**	**Average**	**Unit**
1	0.3262	0.0729	$[m]$
2	0.2850	0.0689	$[m]$
3	0.3358	0.0861	$[m]$
4	0.2841	0.0756	$[m]$
5	0.2976	0.0832	$[m]$
6	0.3003	0.0759	$[m]$
7	0.3192	0.09	$[m]$

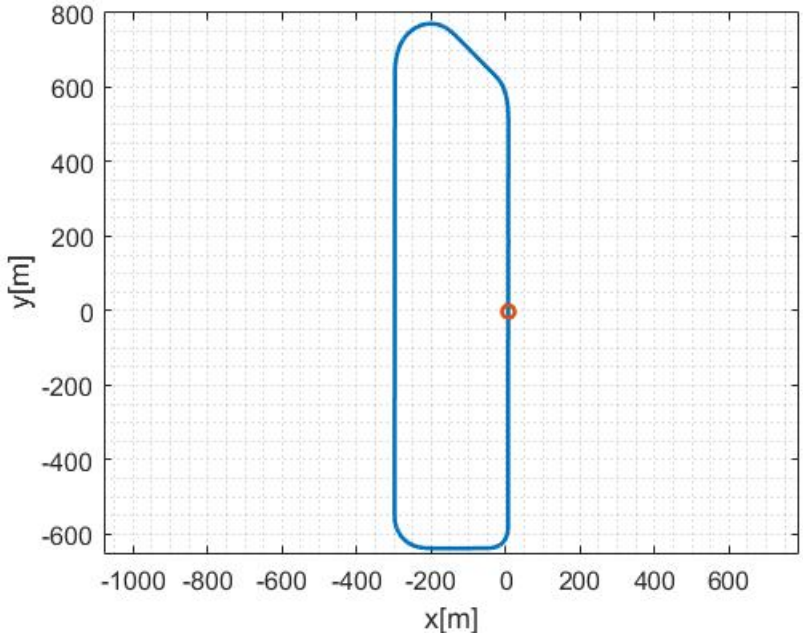

Figure 4.12: Lane keeping test: Vehicle reference path (Reprinted with permission from [1])

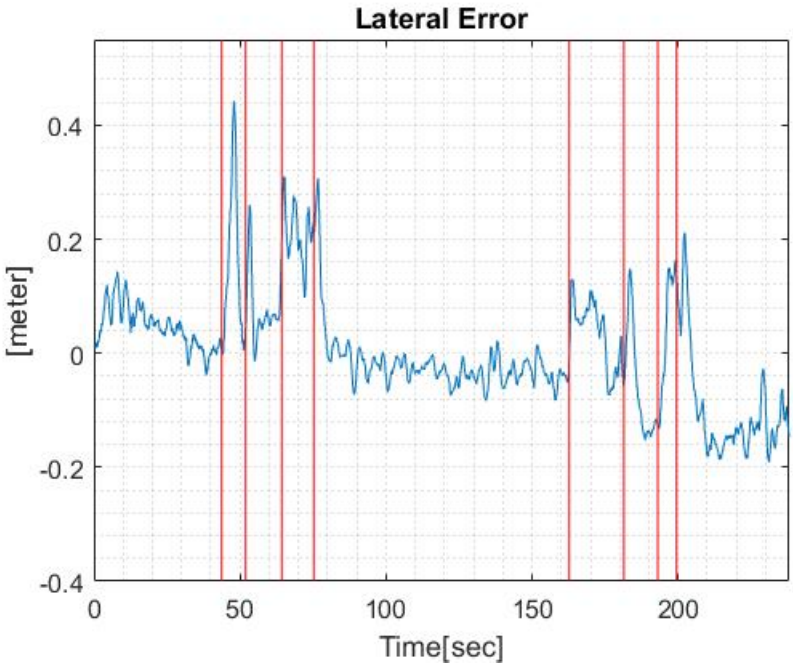

Figure 4.13: Lane keeping test: Lateral error (Reprinted with permission from [1])

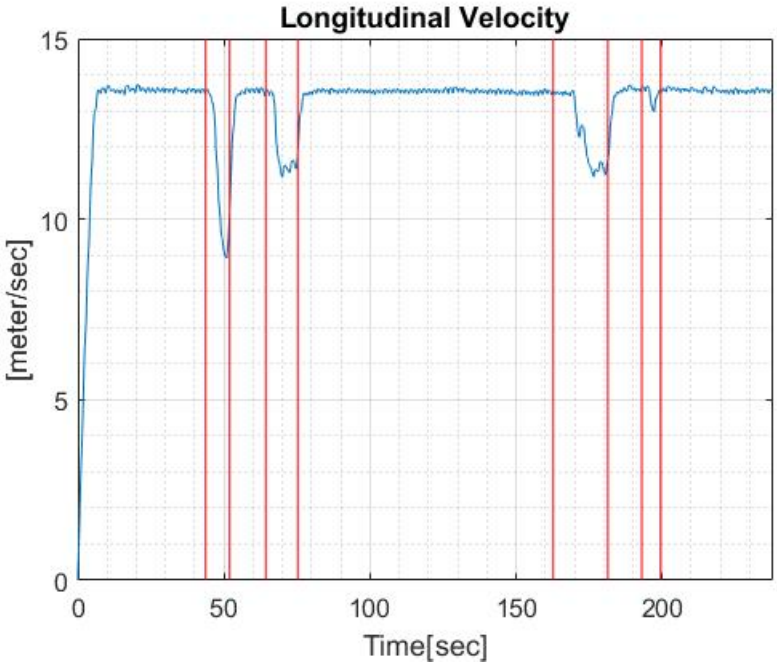

Figure 4.14: Lane keeping test: Longitudinal speed (Reprinted with permission from [1])

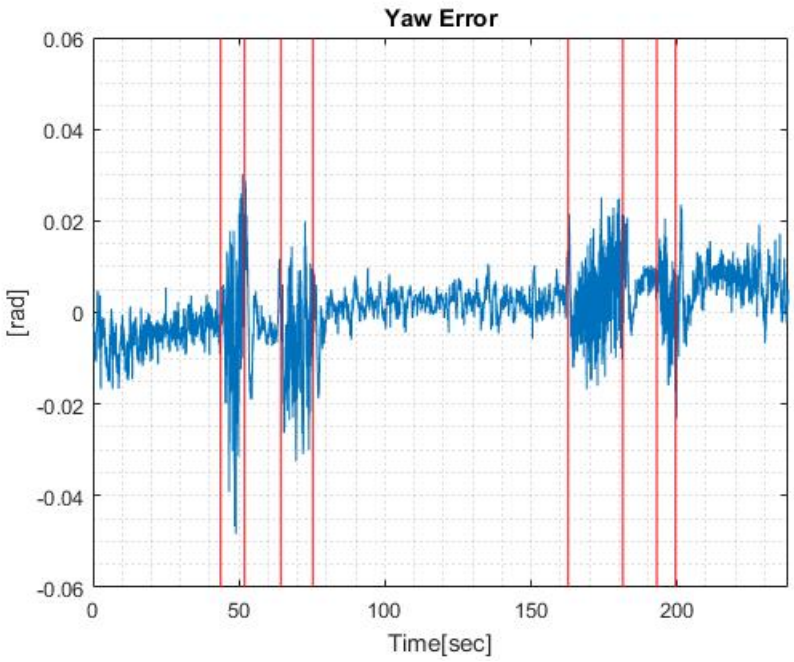

Figure 4.15: Lane keeping test: Yaw error (Reprinted with permission from [1])

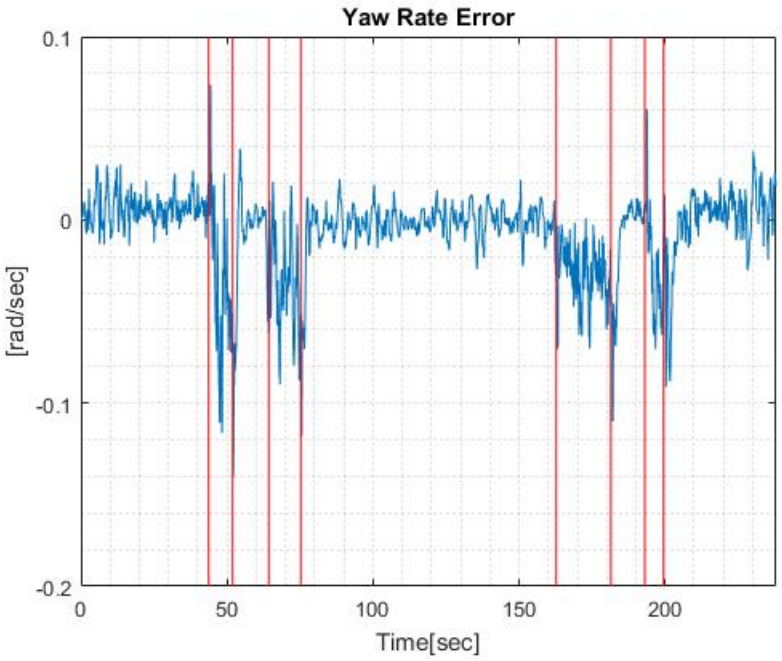

Figure 4.16: Lane keeping test: Yaw rate error (Reprinted with permission from [1])

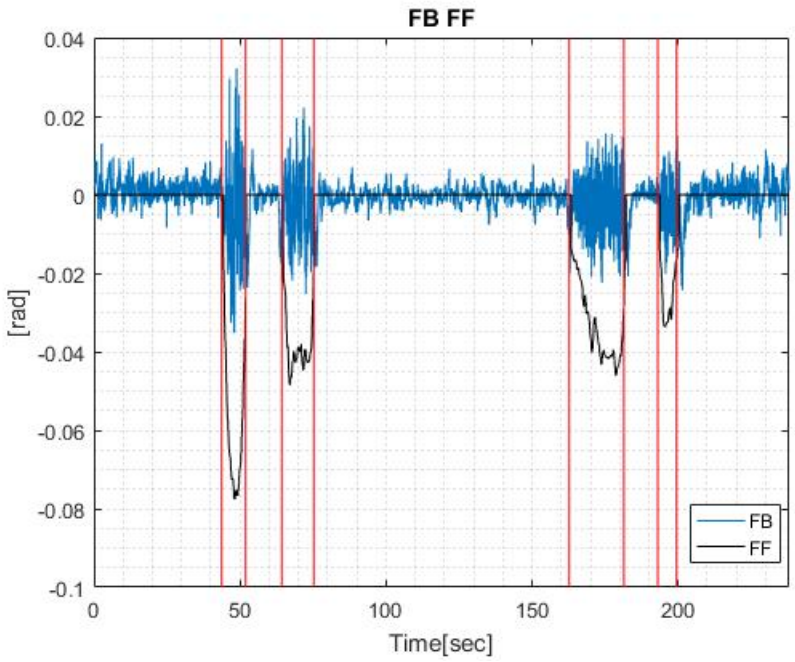

Figure 4.17: Lane keeping test: Feedback/Feedforward command (Reprinted with permission from [1])

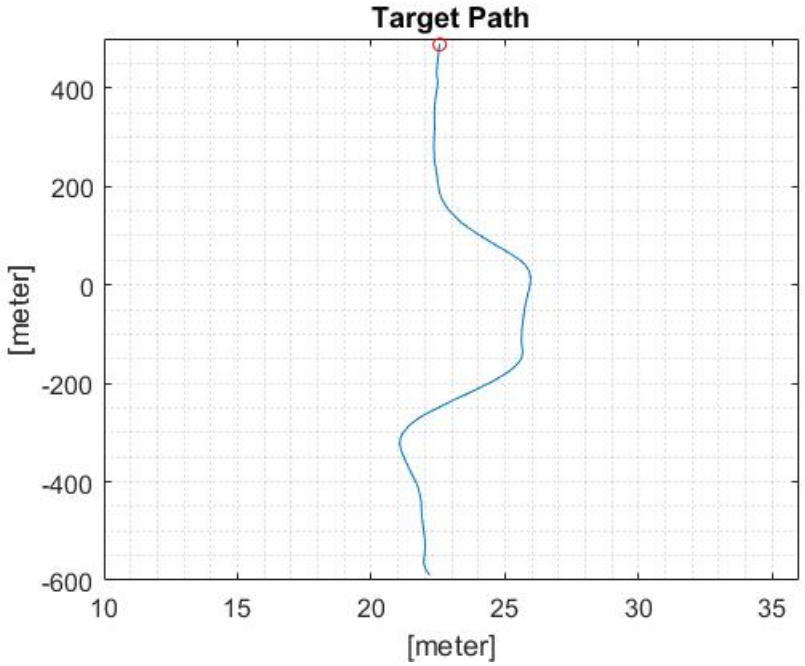

Figure 4.18: Lane change test: Vehicle reference path (Reprinted with permission from [1])

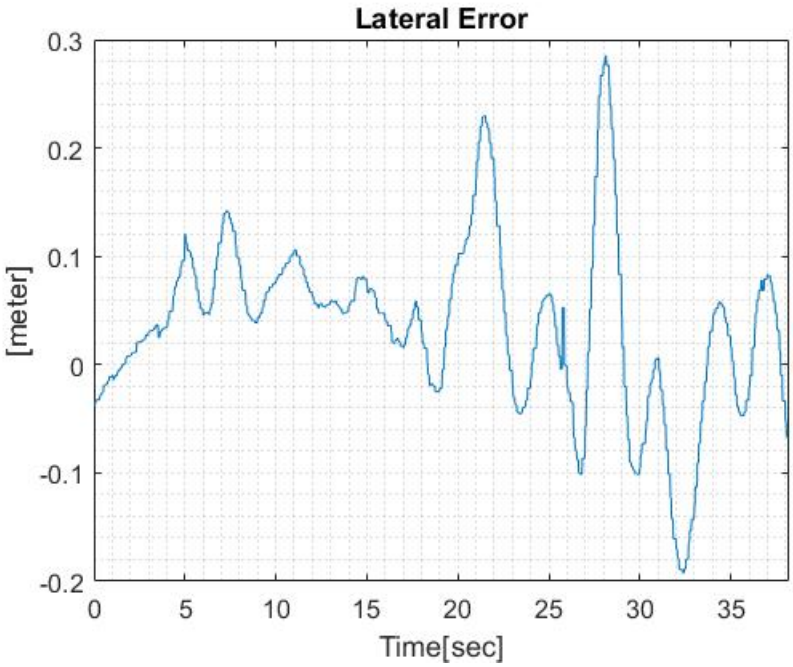

Figure 4.19: Lane change test: Lateral error (Reprinted with permission from [1])

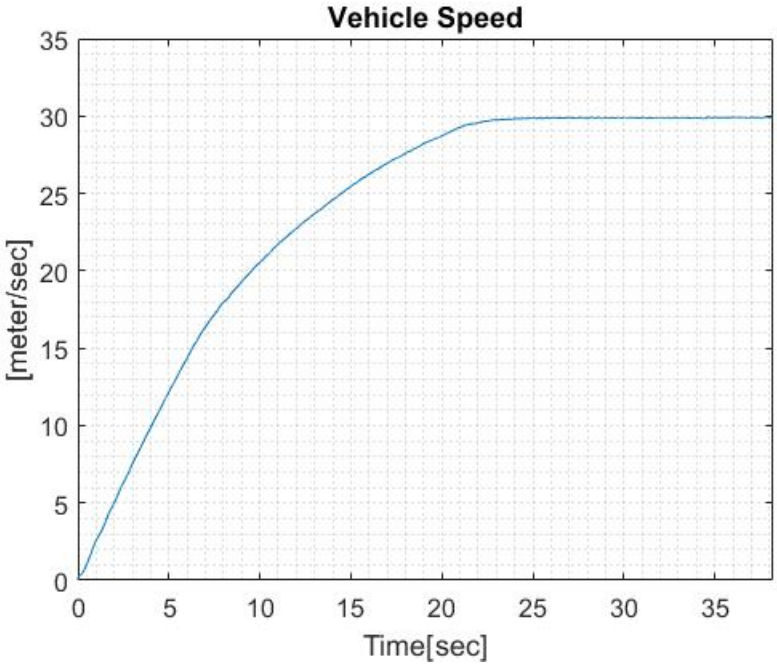

Figure 4.20: Lane change test: Longitudinal speed (Reprinted with permission from [1])

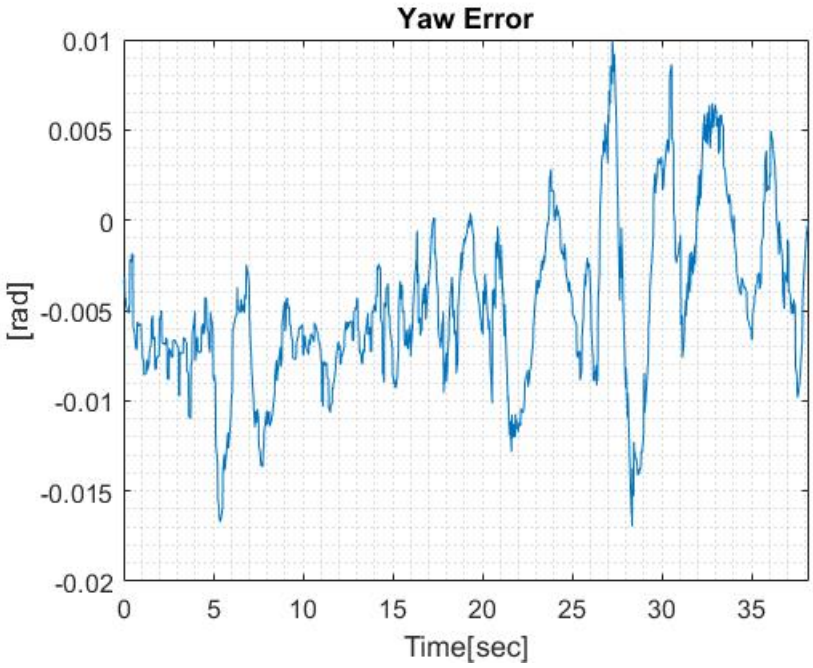

Figure 4.21: Lane change test: Yaw error (Reprinted with permission from [1])

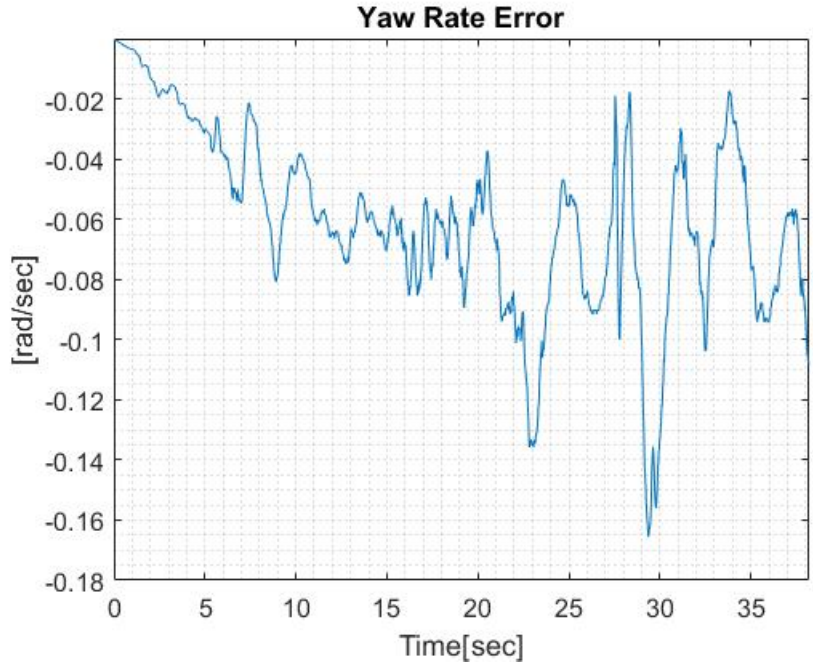

Figure 4.22: Lane change test: Yaw rate error (Reprinted with permission from [1])

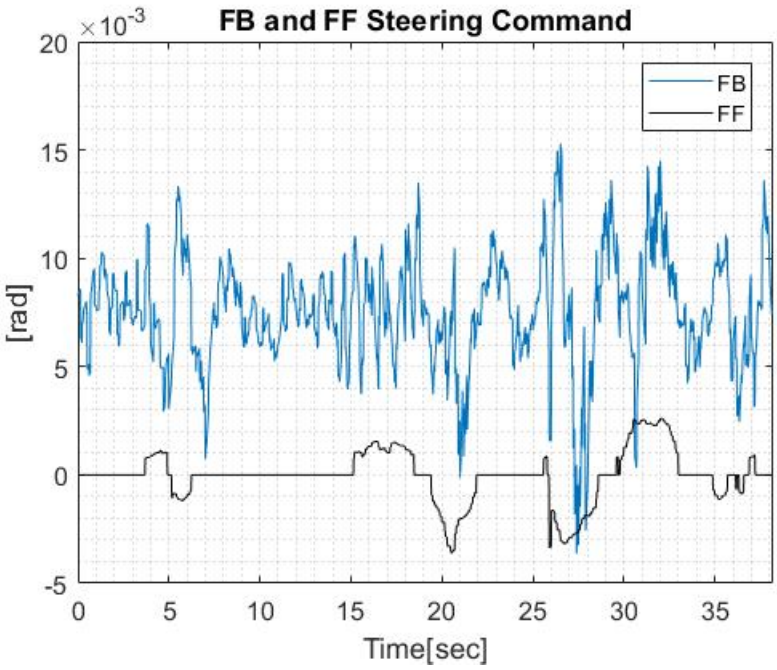

Figure 4.23: Lane change test: Feedback/Feedforward command (Reprinted with permission from [1])

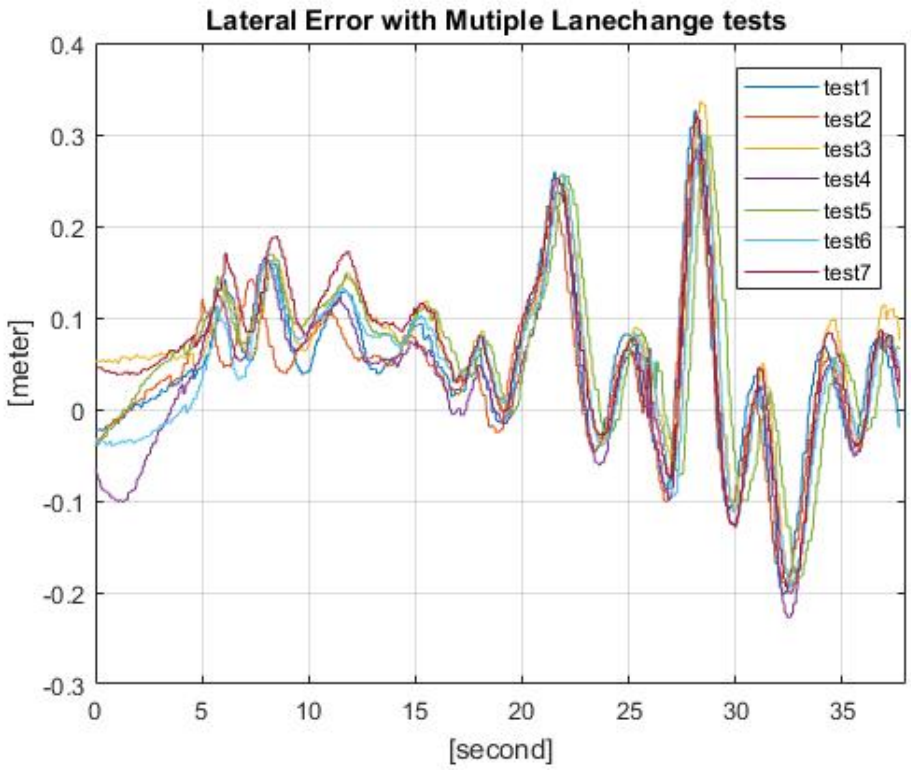

Figure 4.24: Lateral error of the vehicle corresponding to multiple lane changing tests (Reprinted with permission from [1])

5. LATERAL CONTROLLER SYNTHESIS FOR A CONVOY OF AUTONOMOUS AND CONNECTED VEHICLES

In this chapter, we consider an architecture where the lead vehicle in a convoy constructs a trajectory in real-time for an ELC maneuver based on the information that it has, and communicates its position information to its following vehicles. Since vehicles need not be identical, communicating vehicle-specific information such as throttle angle, steering angle, or brake pedal position information may not be suitable; hence, it is simpler from the implementation viewpoint to focus on communicating information about vehicle kinematics , such as vehicle's position, yaw angle, yaw angle rate etc.. Note that the first vehicle of platoon is "leading vehicle", and the one ahead of i^{th} vehicle is "preceding vehicle". Every following vehicle could additionally get the position information of its preceding vehicle (which are GPS coordinates referring to the same frame as lead vehicle), in synthesizing the controller. This architecture is similar in spirit to the longitudinal vehicle following architecture employed by California PATH [50]. However, the details of how the lead vehicle information is used are markedly different, as can be seen in the following sections.

To reiterate [51, 26], consideration of GPS information of lead and preceding vehicles arises from the need to address string instability in the lateral direction - certainly, crosstrack errors should not amplify for ensuring safety. Fig. 5.1 illustrates a convoy, where the ego ACV (shown in purple color) has access to sampled position data of the ACVs ahead. The red and blue colored dots correspond respectively to the locations at which their position data was transmitted to other ACVs including the ego ACV. The ego ACV receives this data, stores it and uses only the data that is within a distance $L_{preview}$ ahead of it; this is shown in Fig. 5.1 as ego preview data; this data can be sorted in increasing order of distance from the ego ACV. let $(x_1^l, y_1^l), (x_2^l, y_2^l), \ldots, (x_N^l, y_N^l)$ (corresponding to red dots in the ego preview data) denote the GPS data of lead vehicle; similarly, let $(x_1^p, y_1^p), (x_2^p, y_2^p), \ldots, (x_M^p, y_M^p)$ denote the data corresponding to the preceding ACV. The problem is to construct a target trajectory as a circular arc spline approximation of the data at hand.

Circular arc spline[1] is especially useful here because (1) a majority of US roads are built as circular arc splines [21], (2) computation of curvature and error signals (namely, crosstrack error, heading and heading rate errors) becomes easy.

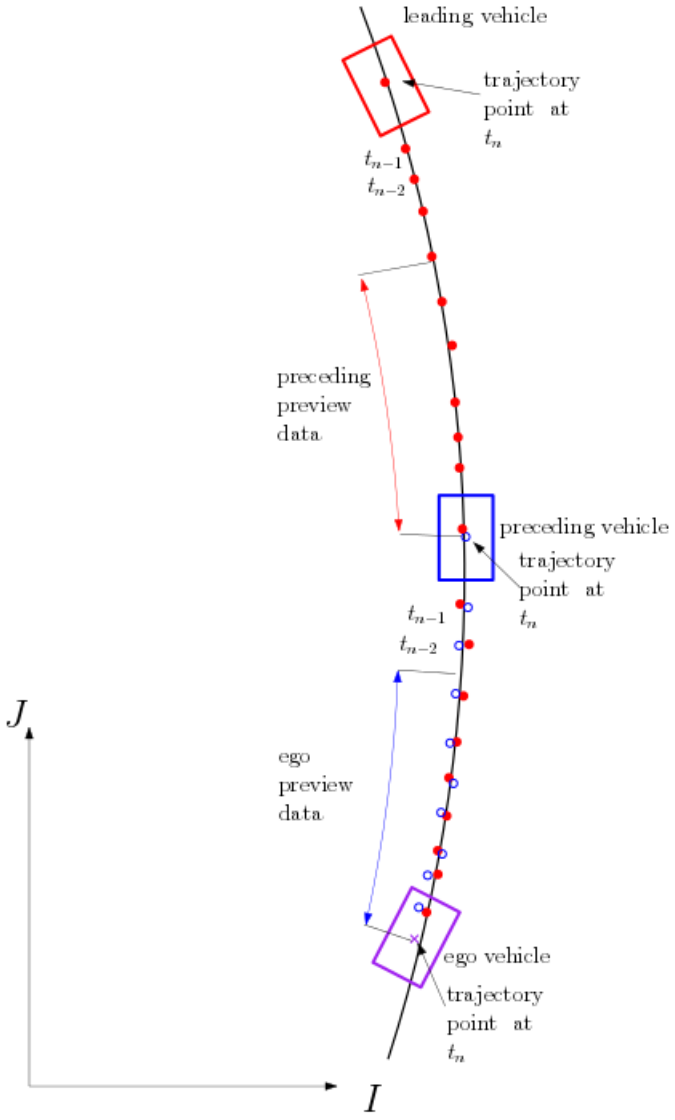

Figure 5.1: Illustration of the sampled trajectories of ACVs in the convoy and the information available to the ego ACV

[1] A circular arc spline is a union of straight line segments and circular arcs so that the resulting curve is continuous with tangent defined everywhere but allows for piecewise constant curvature

It has been mentioned in [52, 53] that human drivers require a preview/lookahead information equivalent to about $0.8\ sec$ of time headway; in other words, if v is the ACV's longitudinal speed, all the preview data within $L_{preview} = 0.8 \times v \approx 30\ m$ should suffice for highway speeds. Typically the maximum frequency of GPS data update is $20\ Hz$ and hence, $M, N < 20$. The duration for a lane change is typically $5 - 6\ sec$ [54]; clearly, for this reason, it is reasonable to assume that there can be atmost one change in curvature of the target trajectory in $1\ sec$. We currently use GPS-RTK [55] that has an accuracy of approximately $2\ cm$ in position; correspondingly, the transmitted data is assumed to be of the same order of accuracy. Since the radius of curvature of the data is typically of the order of $100's$ of meters, and the distance between successive samples of GPS data transmitted by the ACVs is of the order of $1\ m$, it suffices to only consider circular arc splines where circular arcs and straight lines alternate in succession. Since the data is sorted and the GPS-RTK provides position data with a cm accuracy with the distance between successive data samples from ACV being of the order of m, we can exploit this situation further. For example, consider the case where the ego ACV that is going straight needs to decide whether to turn based on the available preview data. Since the data is sorted by distance, consider the data from the preceding vehicle (the same can be done for the lead vehicle also): Find the largest $k \geq 3$ such that points $(x_2^p, y_2^p), \ldots, (x_{k-1}^p, y_{k-1}^p)$ are no farther than a threshold, say $\epsilon\ cm$, from the line connecting (x_1^p, y_1^p) and (x_k^p, y_k^p); clearly, we can associate a straight line segment with the data $(x_1^p, y_1^p), \ldots, (x_k^p, y_k^p)$ and associate a circular arc with the rest of the data. We choose the threshold to be $10\ cm$ (about 5 times the accuracy specified by GPS-RTK). This would allow us to focus on circular arc approximation of the rest of data.

There can be various architectures possible for constructing an ELC trajectory; for example, as in California PATH longitudinal control architecture, one could have the sampled GPS information of the lead and preceding vehicles via V2X communication. Such V2X communication is also be adopted by previous works[56, 57, 58]. Consideration of GPS information of lead and preceding vehicles arises from the need to address string instability in the lateral direction - certainly, lateral tracking errors should not amplify for ensuring the safety of vehicles in the neighboring

lanes. Without lead and preceding vehicles' information, the lateral string stability can not be guaranteed[56]. In this case, there are two possible architectures:

- Synthesize a composite ELC trajectory that fuses the GPS information of both lead and preceding vehicles of every following vehicle in the convoy.
- Synthesize many ELC trajectories with each trajectory corresponding to the information from a preceding/lead vehicle. One must then correspondingly synthesize feedback information to regulate the tracking error.

A general formulation for fitting a circular arc that would suffice for these two possible architectures will be considered in the next subsection.

5.1 Synthesizing a Composite ELC Trajectory

5.1.1 Fitting a circular arc based on preview data

There are two sources for the preview data – lead ACV and the preceding ACV; one may want to weigh the data differently. For this case, consider a weighing parameter $\alpha \in [0, 1]$ that will be used in defining a convex combination of two different error functions.

Let (X_c, Y_c) be the coordinates of the center of the circle sought and R be its radius.

A circle is defined by an equation of the form:

$$e(x, y) = 0,$$

where

$$e(x, y) := (x - X_c)^2 + (y - Y_c)^2 - R^2.$$

One can think of πe as the difference between the areas of a circle passing through a point (x, y) with center at (X_c, Y_c) and another circle of radius R. Associated with the given data,

$(x_1^p, y_1^p), \ldots, (x_M^p, y_M^p)$, $(x_1^l, y_1^l), \ldots, (x_N^l, y_N^l)$, one can define a composite error

$$J = \alpha \sum_{i=1}^{M} (e(x_i^p, y_i^p))^2 + (1-\alpha) \sum_{j=1}^{N} (e(x_j^l, y_j^l))^2.$$

Then $(X_c, Y_c), R$ can be determined as the arguments minimizing J and can be determined from three linear equations in $X_c, Y_c, (R^2 - X_c^2 - Y_c^2)$.

In the first architecture, we pick $\alpha = 0.5$, i.e., we provide equal weight to the data from lead and preceding ACVs and then determine the composite ELC trajectory. The choice of $\alpha = 0.5$ simplifies implementation as we can treat the data from lead and preceding ACVs as coming from the same source ACV.

In the second architecture, we can set $\alpha = 1$ to compute a target trajectory based on the data associated with preceding ACV and $\alpha = 0$ to compute another target trajectory based on the data associated with lead ACV.

Once an ELC trajectory is computed, one must now calculate the feedback signals: the lateral error of the vehicle from the trajectory, e_{lat}, the heading error of the vehicle, $\tilde{\theta}$, and the heading rate error, $\dot{\tilde{\theta}}$. The radius of curvature allows us to compute the feedforward control with respect to an ELC trajectory and the feedback signals help vehicle track the trajectory!

5.2 Lateral Controller Synthesis

Lateral controller for tracking a trajectory can be decomposed into two parts: a feedforward part and a feedback part, i.e.,

$$\delta_c = \delta_{ff} + \delta_{fb} \tag{5.1}$$

In the following subsections, we will outline how these controllers are designed:

5.2.1 Feedforward Controller

Feedforward controller structure depends on the architecture employed.

5.2.1.1 With a Composite ELC Trajectory:

In this case, there is only a single trajectory. The feedforward controller essentially provides the steering input that would keep the vehicle on the circular part of trajectory without any feedback if the initial conditions were to be appropriate and if the disturbance inputs on the vehicle are absent. One may view the feedforward steering input, δ_{ff}, as that steering ,input which results in the vehicle's trajectory being a circle of radius R when it is traveling at a longitudinal speed of V_0. Clearly, the corresponding feedforward steering input is:

$$\delta_{ff} = \frac{a+b}{R} + K_{sg}\frac{V_0^2}{R} \tag{5.2}$$

The detailed derivation has already been present in Chapter4 Note that for understeered vehicles, $K_{sg} > 0$ and the open-loop handling dynamics will be unconditionally stable [40, 41].

5.2.1.2 Separate ELC Trajectories Based on Lead and Preceding Vehicles

Let $\alpha \in (0, 1)$ denote the weight we assign to the lead vehicle's trajectory in terms of tracking. Then

$$\delta_{ff} = \alpha(\frac{a+b}{R_l} + K_{sg}\frac{V_0^2}{R_l}) + (1-\alpha)(\frac{a+b}{R_p} + K_{sg}\frac{V_0^2}{R_p}) \tag{5.3}$$

where R_l, R_p denote respectively the radii of trajectory computed based on the preview data of lead and preceding vehicles respectively.

5.2.2 Feedback Controller Design

As with feedforward controller design, feedback controller design also depends on the architecture.

5.2.2.1 *With a Composite ELC Trajectory:*

Feedback control is based on available information, namely, $e_{lat}, \tilde{\theta}$ and $\dot{\tilde{\theta}}$; the lateral velocity information is not readily available; all the error signals can be found in Chapter2. Hence, we seek a control law of the form:

$$\delta_{fb} = -k_e e_{lat} - k_\theta \tilde{\theta} - k_w \dot{\tilde{\theta}} \tag{5.4}$$

where the gains k_e, k_θ and k_w need to be determined.

5.2.2.2 *Separate ELC trajectories corresponding to lead and preceding vehicle's preview data*

Suppose $e_{lat,l}$, $\tilde{\theta}_l$ and $\dot{\tilde{\theta}}_l$ be the lateral spacing error, heading error and yaw rate error corresponding to the ELC trajectory based on lead ACV's data; similarly, $e_{lat,p}$, $\tilde{\theta}_p$ and $\dot{\tilde{\theta}}_p$ be corresponding errors for the ELC trajectory based on preview data from preceding ACV.

The structure of feedback controller we pick in this case is as follows:

$$\delta_{fb} = -(1-\alpha)(k_e e_{lat,l} + k_\theta \tilde{\theta}_l + k_\omega \dot{\tilde{\theta}}_l) - \alpha(k_e e_{lat,p} + k_\theta \tilde{\theta}_p + k_\omega \dot{\tilde{\theta}}_p).$$

5.2.3 Construction of the Set of Stabilizing Structured Feedback Controllers

The gains k_e, k_θ and k_ω must be chosen for implementation; in this subsection, we will adopt the procedure from our earlier work on the construction of structured feedback controllers to arrive at the set for both cases.

5.2.3.1 *With a Composite ELC Trajectory*

Our earlier work in Chapter4 is directly can be applied to synthesizing the set of stabilizing feedback gains. From the governing equations, the transfer functions relating $e_{lat}(s)$ and $\tilde{\theta}(s)$ to

$\delta_f(s)$ can be related through the open-loop characteristic polynomial, $\Delta_o(s)$ as:

$$\begin{aligned}
\Delta_o(s) &= s^2(mIs^2 + \frac{(I+ma^2)C_f + (I+mb^2)C_r}{V_0}s \\
&\quad + \frac{(a+b)^2 C_f C_r}{V_0^2} - m(aC_f - bC_r)), \qquad (5.5) \\
\frac{e_{lat}(s)}{\delta_f(s)} &= \frac{C_f}{\Delta_o(s)}(Is^2 + \frac{b(a+b)C_r}{V_0}s + (a+b)C_r), \qquad (5.6) \\
\frac{\tilde{\theta}(s)}{\delta_f(s)} &= \frac{C_f}{\Delta_o(s)}(mas^2 + \frac{(a+b)C_r}{V_0}s). \qquad (5.7)
\end{aligned}$$

Since

$$\begin{aligned}
\delta_f &= H_a(s)\delta_c(s) \\
&= -H_a(s)(k_e e_{lat}(s) + (k_\theta + k_w s)\tilde{\theta}(s)), \qquad (5.8)
\end{aligned}$$

it follows that the closed-loop characteristic equation can be expressed as:

$$\begin{aligned}
&\Delta_o(s) + H_a(s)k_e C_f(Is^2 + \frac{b(a+b)C_r}{V_0}s + (a+b)C_r) \\
&\quad + H_a(s)(k_\theta + k_w s)C_f(mas^2 + \frac{(a+b)C_r}{V_0}s) = 0. \qquad (5.9)
\end{aligned}$$

For the second-order steering actuation model, it follows that the characteristic polynomial may be expressed in terms of the control parameter vector $\mathbf{K} = (k_e, k_\theta, k_w)$ as:

$$\begin{aligned}
\Delta(s;\mathbf{K}) &= (s^2 + 2\zeta\omega_n s + \omega_n^2)\Delta_o(s) + k_e\omega_n^2 C_f(Is^2 + \frac{b(a+b)C_r}{V_0}s + (a+b)C_r) \\
&\quad + (k_\theta + k_w s)\omega_n^2 C_f(mas^2 + \frac{(a+b)C_r}{V_0}s). \qquad (5.10)
\end{aligned}$$

5.2.3.2 *With Separate ELC Trajectories Based on Preview Data of Preceding and Leading vehicles*

One can examine the evolution of $e_{lat,p}$ and apply the feedforward and feedback control inputs specified in the earlier subsections for this case. We can rewrite the feedback law as:

$$\begin{aligned} \delta_{fb} &= -(k_e e_{lat,p} + k_p \tilde{\theta}_p + k_\omega \dot{\tilde{\theta}}_p) \\ &\quad +\alpha(k_e(e_{lat,p} - e_{lat,l}) + k_\theta(\tilde{\theta}_p - \tilde{\theta}_l) + k_\omega(\dot{\tilde{\theta}}_p - \dot{\tilde{\theta}}_l)). \end{aligned} \tag{5.11}$$

One may neglect the component of state: $e_{lat,p}, \tilde{\theta}_p, \dot{\tilde{\theta}}_p$ in the terms $e_{lat,p} - e_{lat,l}$, $\tilde{\theta}_p - \tilde{\theta}_l$ and $\dot{\tilde{\theta}}_p - \dot{\tilde{\theta}}_l$ as these will be dominated by the difference in the ELC trajectories constructed based on preview data of lead and preceding vehicles. For this reason, the characteristic polynomial will remain the same as before. Hence, a fundamental problem regarding the stability of tracking is to determine the set of control gains, $\mathbf{K}$, so that the closed-loop characteristic polynomial (B.1) is Hurwitz. An advantage of determining the entire set is that the set can be pruned to accommodate additional performance criteria.

5.2.3.3 *Construction of the Set of Stabilizing Fixed Structure Controllers*

In Chapter 4, we present "D-decomposition" for the construction of the set of stabilizing fixed structure controllers. Essentially, this approach relies on the continuous dependence of roots of a polynomial on its coefficients for every regular perturbation. The approach then involves decomposing the parameter space into disjoint signature-invariant regions by identifying their boundaries. The boundaries can be obtained by (a) determining the set of $\mathbf{K}$ for which $\Delta(0, \mathbf{K}) = 0$ and (b) determining the set of $\mathbf{K}$ for which $\Delta(jw, \mathbf{K}) = 0$ for some w. Once the parameter space is partitioned, one can then sample every partition to determine the partition corresponding to $\Delta(s; \mathbf{K})$

being Hurwitz. Recall that $\Delta(s, \mathbf{K}) = A_6 s^6 + A_5 s^5 + A_4 s^4 + A_3 s^3 + A_2 s^2 + A_1 s + A_0$, where

$$
\begin{aligned}
A_6 &= \frac{Im}{\omega_n^2}, \\
A_5 &= \frac{2Im\zeta}{\omega_n} + \frac{C_f(I + a^2 m) + C_r(I + b^2 m)}{V_0 \omega_n^2}, \\
A_4 &= \frac{2\zeta(C_f(I + a^2 m) + C_r(I + b^2 m))}{V_0 \omega_n} \\
&+ (\frac{(a+b)^2 C_f C_r}{V_0^2 \omega_n^2} - m(aC_f - bC_r)) + mI, \\
A_3 &= (\frac{2\zeta((a+b)^2 C_f C_r)}{V_0^2 \omega_n} - m(aC_f - bC_r)) \\
&+ \frac{C_f(I + a^2 m) + C_r(I + b^2 m)}{V_0} + C_f m a k_\omega, \\
A_2 &= (\frac{(a+b)^2 C_f C_r}{V_0^2} - m(aC_f - bC_r)) \\
&+ (C_f I k_e + C_f C_r \frac{(a+b)}{V_0} k_\omega + maC_f k_\theta), \\
A_1 &= C_f C_r \frac{(a+b)}{V_0}(bk_e + k_\theta), \\
A_0 &= C_f C_r (a+b) k_e,
\end{aligned}
$$

where ζ, ω_n are the damping ratio and natural frequency of steering actuation. The rest of procedure is same as Chapter 4.

5.3 Numerical and Experimental Results

5.3.1 Construction of Stabilizing Set of Feedback Gains

In order to construct the set of stabilizing feedback gains, we used the parameters from Table 5.1.In this test, we set the vehicle numerical test speed at $V_0 = 67\ mph$. The vehicle's cornering stiffness can be fixed by the Fig. 3.7. Since the stabilizing set (which is non-convex) depends on V_0, we find the sets of all the stabilizing controllers for a range of speeds in $\{10, 20, 30, 40, 50, 60, 67\}\ mph$ and selected a controller $(k_e, k_\theta, k_\omega) = (0.06, 0.96, 0.08)$ that belongs to the interior of the intersection of all these stabilizing sets. The stabilizing set corresponding to $V_0 = 67\ mph$ is shown in Fig. 4.11.

Table 5.1: Parameters of EPS and vehicle with $V = 67\ mph$

Parameter	Value	Unit
m	1896	$[kg]$
I	3803	$[kgm^2]$
a	1.2682	$[m]$
b	1.5818	$[m]$
C_f	400000	$[N/rad]$
C_r	381900	$[N/rad]$
ζ	0.4056	[-]
ω_n	21.4813	$[rad/s]$

5.3.2 Numerical Simulation Setup

The trajectory in Fig. 5.2 represents a nominal double lane change on a $1\ km$ road section and is the target trajectory for lead ACV. It corresponds to an ACV turning left to go to the adjacent left lane and then right to get back into the original lane. The second ACV (literally, the first following ACV) in the convoy will use the sampled trajectory information of the lead ACV and adopt the control law developed in this paper. All other vehicles in the platoon will have access to the sampled trajectory information of the lead and their preceding ACVs path and adopt the lateral controller designed in the earlier section; we test the two designs - one is based on the composite ELC trajectory, the other one is based on separate trajectory on lead and preceding vehicles. For simulations in this section, we consider a four-vehicle platoon. Every vehicle in the platoon maintains a constant speed of $30m/s$. Initially, all the vehicles have zero lateral value and zero heading error.

5.3.2.1 Lateral control with a composite ELC trajectory

For simplicity of implementation, α was chosen to be 0.5 in the computation of the composite ELC trajectory for each ACV. Based on the composite ELC trajectory, feedforward control input, feedback error signals and feedback control have been determined as described in the controller synthesis section. Fig. 5.3 shows the lateral error response of the control scheme while trying to

track the composite ELC trajectory in Fig. 5.2. In the double lane change maneuver considered, the first (left) lane change begins at around 8 seconds and ends before 15 seconds; the second (right) lane change begins at approximately 24 seconds and ends at 30 seconds. The maximum lateral error was found to be approximately $8\ cm$ and occurred at the beginning and end of curved section of the trajectory; see Fig 5.3; moreover, the maximum lateral errors seem to form a monotonically decreasing sequence affirming lateral string stability.

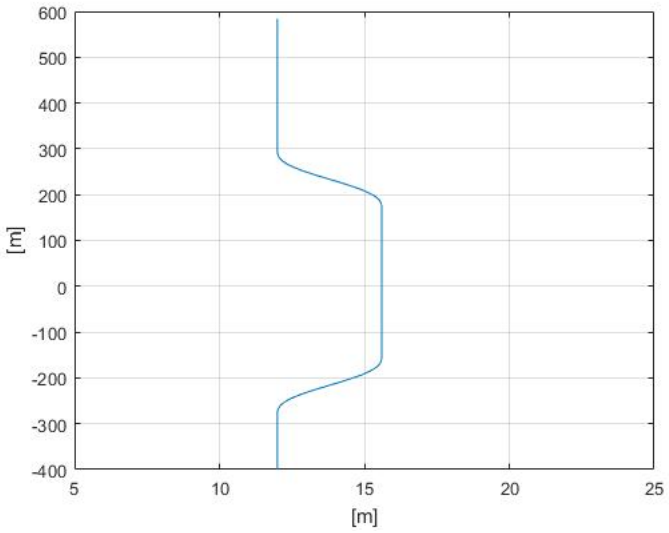

Figure 5.2: Composite ELC Trajectory: Vehicle reference path

5.3.2.2 Lateral control with separate ELC trajectories

In this section, consider the separate ELC trajectories, one based on the lead vehicle's data and the other based on preceding vehicle's data. As described in Section III, we design the feedback and feedforward controllers and weigh them equally, i.e., set $\alpha = 0.5$. The target paths is shown in Fig. 5.4. As shown in Fig. 5.5, the maximum lateral error of about 0.08 meters was obtained and occurred at the beginning and end turning section of lane change. Moreover, string instability was not observed in the simulations.

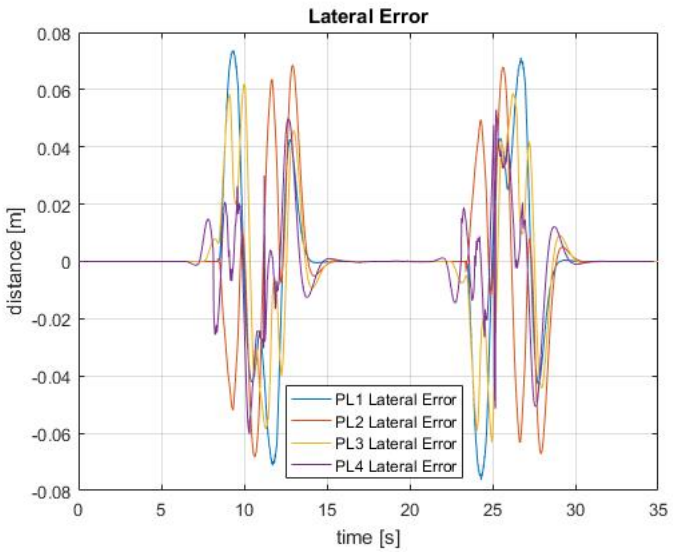

Figure 5.3: Composite ELC Trajectory: Lateral error

5.3.3 Experimental Setup

The developed control algorithms were implemented on a Lincoln MKZ car. The vehicle was equipped with a drive-by wire system for autonomous steering, throttle and brake control. Data concerning the vehicle states (position, yaw angle, and yaw rate) were obtained from an on-board integrated GPS system and IMU unit. The controller operates at 50 Hz. Since we had only one Lincoln MKZ, we emulated a convoy with multiple runs of the same vehicle - this was possible because of the look-ahead nature of the control scheme as every following vehicle needed only the information of its preceding ACV and the lead ACV in the convoy. In the first run, the Lincoln MKZ vehicle would track the trajectory as a lead ACV in the convoy; the closed loop trajectory information (provided via time stamped GPS information) is stored on-board and accessed in subsequent runs. In the second run, it will access the stored GPS information from the first run to simulate the acquisition of perfectly communicated GPS information from the lead ACV; in the i^{th} run, it would access the stored trajectory information from the preceding run and the first run to compute the steering command.

In the experiment conducted on our RELLIS campus, the target speed of every vehicle in the

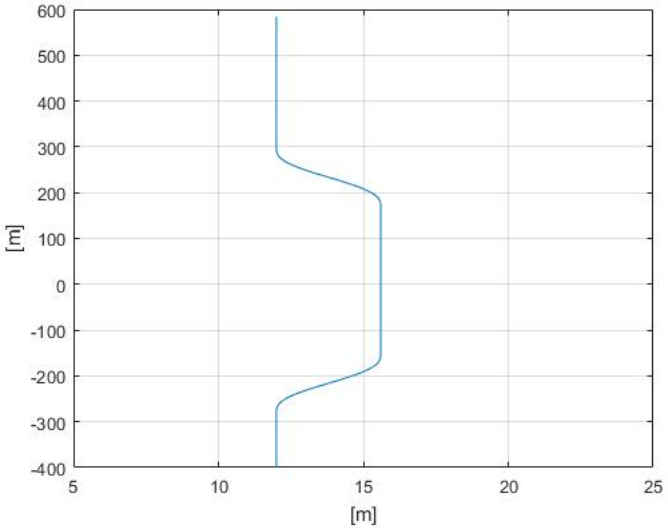

Figure 5.4: Separate Trajectories: Vehicle reference path

platoon is $25m/s$. The target trajectory for the experiment is the same as the one considered in the numerical simulation section. *Due to the time limitation, we computed the steering angle using the first architecture described in this paper - which is based on composite ELC trajectory.* The desired trajectory is shown as Fig. 5.6.

5.3.4 String Instability with lack of lead ACV's preview data

If the target trajectory for a following ACV were to be constructed solely based on the position (preview) data of its preceding vehicle, lateral string instability can set in. In the plot, $PL1$ corresponds to the lead vehicle, $PL2$ and $PL3$ correspond to the first and second following ACVs respectively. As can be seen from Fig. 5.8, crosstrack errors seem to amplify. Animation of these results suggest that the second following ACV crosses the lane boundary `https://youtu.be/UT1OYCb50gU`.

5.3.5 String Stability with lead ACV's preview data

If the target trajectory for the following ACV incorporates the lead vehicle's position information, then lateral string stability was observed in Fig. 5.7. The readers are encouraged to view the accompanying video at `https://youtu.be/UT1OYCb50gU`.

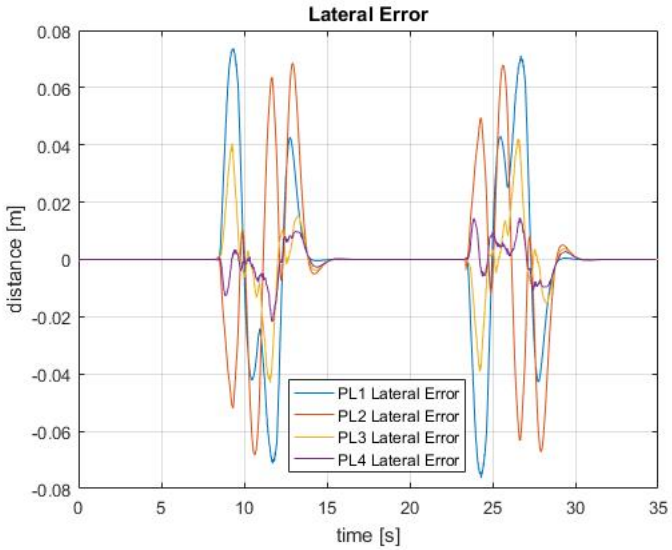

Figure 5.5: Separate Trajectories: Lateral difference between vehicles and their preceding one's track

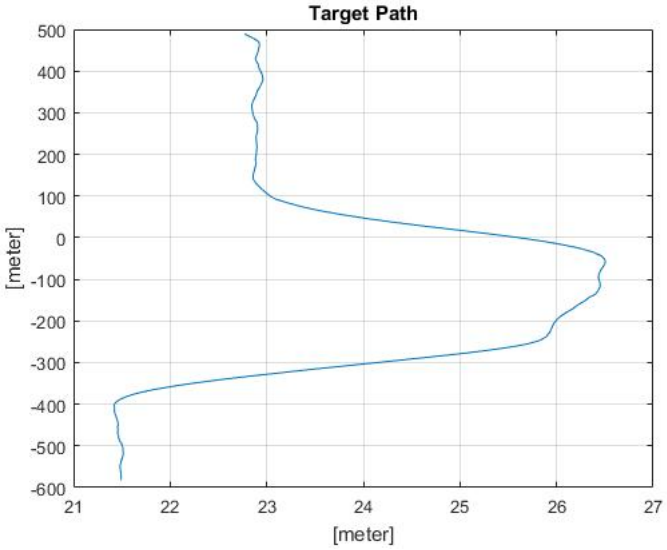

Figure 5.6: Experiment: Vehicle reference path

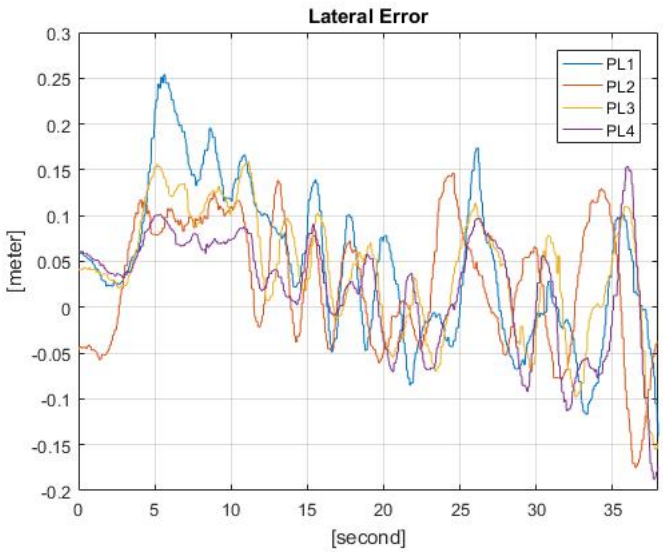

Figure 5.7: Experiment: Lateral String Stability With Lead Vehicle's (Preview) Data

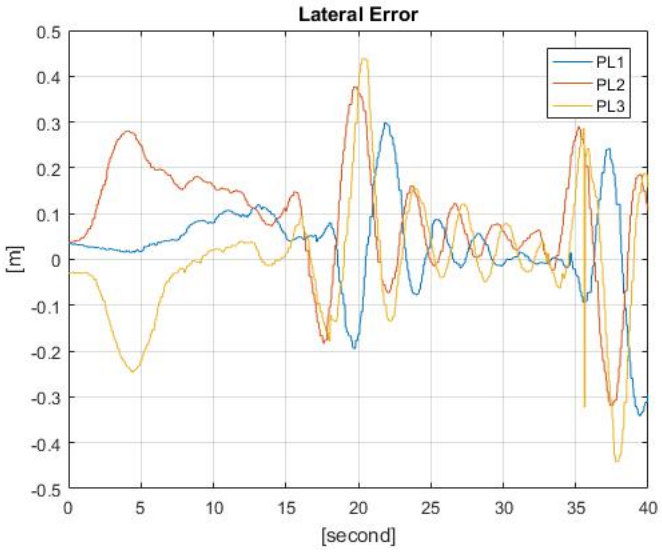

Figure 5.8: Experiment: Lateral String Instability Without Lead Vehicle's (Preview) Data

6. FUSING RADAR AND VISION DATA FOR CUT-IN VEHICLE IDENTIFICATION IN PLATOONING APPLICATIONS

6.1 Introduction

Truck platooning systems (Level 2) extend the radar, camera and vehicle-to-vehicle communications based, cooperative adaptive cruise control to provide precise automated lateral and longitudinal vehicle control in order to maintain a tight formation of vehicles with short following distances. A manually driven truck leads a platoon, while the steering, acceleration and braking of following truck(s) are automatically controlled with the driver(s) engaged, monitoring system performance and the driving environment at all times. Level 1 truck platooning has already demonstrated the potential for significant fuel savings, enhanced mobility, and associated emissions reductions from platooning vehicles. Level 2 automation may increase these benefits while reducing driver workload and increasing safety.

The objective of this work is to address fundamental problems that arise when vehicles present in the adjacent lanes of the platooning vehicles maneuver to change lanes and cut into the gap between the platooning trucks. The sensing systems (figure 6.1) in the platooning trucks must be able to identify and estimate the state of these "cut-in" vehicles in real-time so that suitable control actions can be taken. The current truck platooning system developed at the Texas Transportation Institute uses a forward looking scanning radar (Delphi [59]), a forward looking camera (MobilEye) and Dedicated Short Range Communication (DSRC) information for situation awareness. The specific goal of this article is to develop a sensor fusion algorithm that combines radar, camera and inertial data to provide real-time estimates of any cut-in vehicle present in the field of view of these sensors. Development of these algorithms can significantly increase the situational awareness of the platooning system, and improve the safety and performance of the overall system.

Fusing radar, camera and inertial data sensed from moving platforms is challenging for several reasons. The sensors, radar and camera, on-board the trucks provide data at around 20 Hz that

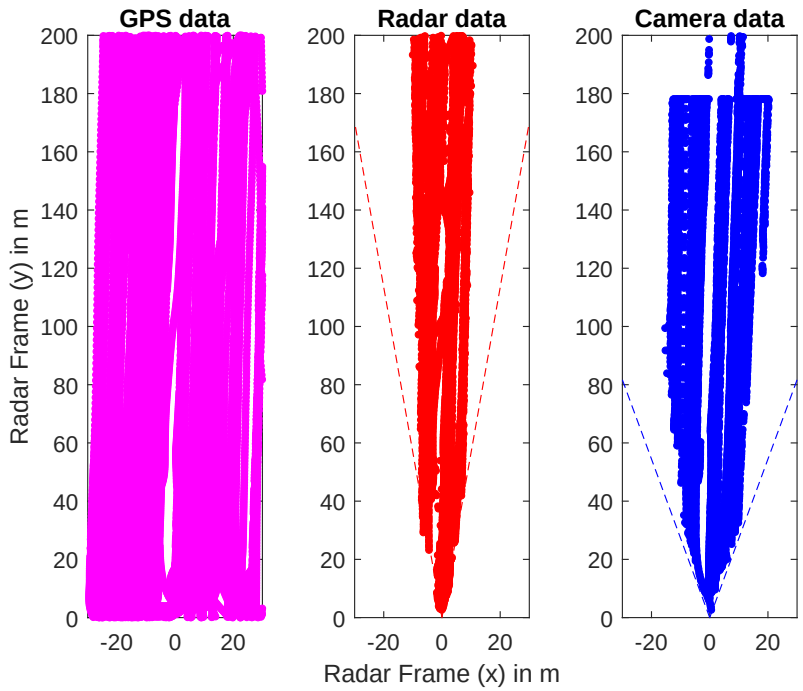

Figure 6.1: Data collected of neighboring vehicles from a stationary truck using the onboard sensors over a period of 40 minutes. The first subplot shows the GPS data of the center of mass of the vehicles. The second subplot shows the radar data and the third subplot shows the corresponding camera data collected on-board the truck. The radar has a field of view of 20 degrees and the camera has a field of view of 40 degrees.

is intermittent and noisy. We obtain this data through the Bendix Wingman Fusion system in the form of ten tracks associated with both the radar and camera. Each radar track returns the following estimates of an object: radar confidence, range, range rate, lateral position, object angle and acceleration of the object. Each camera track returns the following estimates of an object: angle corresponding to the left corner of the object, angle corresponding to the right corner of the object, longitudinal distance and range rate. At any time instant, an object may be present in the radar data and not be present in the camera data and vice-versa. Also, the object-track assignment in both radar and camera data is not known a-priori. For example, an object can appear in track 1 of the radar data at a time instant, and then switch tracks in other time instants. It is also possible that an object appears in multiple tracks of radar at any time instant. The sensor fusion algorithms must be able to handle these challenges in real-time.

Sensor fusion algorithms that track multiple objects in real-time has been of significant interest to the researchers over the last couple of decades owing to its importance in transportation, robotics and autonomous vehicle applications. The algorithms generally differ in the type of sensors used, how the data association problem between different sensors is addressed, the type of object tracking problems posed and any specific constraints with respect to the context/application where algorithms will be finally used. Data collected from sensors on moving platforms and the field of view of the sensors also play a critical role in posing the different subproblems that arise in sensor fusion. As a comprehensive review of multiple object tracking is out of scope of this conference paper, we refer the readers for survey papers in [60, 61]. Here, we mention few papers related to our work.

LIDAR and vision data is fused for pedestrian identification in [61]. A multiple object tracking algorithm for identifying pedestrians and cyclists using radar and vision data is presented in [62]. A particle filter based algorithm for pedestrian tracking using radar and vision data is developed in [63]. Authors in [64] present a perception system built from multiple cameras, LIDARs and radars that can be deployed on cars in realistic environments which aim to track multiple classes of objects including vehicles, pedestrians, bicyclists etc. LIDAR and radar data are fused in [65].

In this Chapter, we fuse data from a forward looking radar and camera from a moving vehicle to estimate the state of the cut-in vehicles in the adjacent lanes. Each of the sensors (radar and camera) already provide us with processed measurements such as range, range-rate, object angle etc. We develop a data association algorithm to first associate a radar measurement with a camera measurement, and then assign each fused measurement with the tracked objects (section 6.3). The state equations for describing the motion of a cut-in vehicle are developed from a moving platform in section 6.4. An Extended Kalman Filter is then used to predict the motion of each of the cut-in vehicles in the field of view of the sensors (section 6.5). Experimental results are provided with multiple cut-in vehicles from a stationary and a moving truck to corroborate the performance of the proposed algorithms in section 6.6. Finally, the article concludes with some suggestions for future work.

6.2 Overall Approach

There are four keys steps in our approach. First, at each time instant, we pre-process the radar data to only include any track with a confidence that is at least equal to a minimum threshold[1]. Also, any radar measurement that is very close to another radar measurement is deleted. In the second step, we fuse the radar and camera measurements using a nearest neighbor algorithm with a ranking on the priority of the measurements (section 6.3). In the third step, we use a simple greedy algorithm to assign a fused measurement to an object. Specifically, a fused measurement is assigned to an object if the Euclidean distance between the measurement and the predicted state of the object is less than a pre-defined limit. Any un-assigned fused measurement is retained for a specific amount of time until we can associate it with another fused measurement at a later time. Any object or a un-assigned fused measurement that cannot be assigned to another fused measurement obtained within a time horizon is removed from the system. At the end of third step, we have a list of objects with (possibly) a fused measurement assigned to each object. In the final step, an extended Kalman filter is used to find the new state of each object (sections 6.4 and 6.5). Note that our approach is applicable to any moving object. In the platooning application, the moving object we are interested in is a cut-in vehicle.

6.3 Fusing Radar and Camera Data

There are several methods available for fusing data from different sensors: Nearest Neighbour (NN) algorithm, Probabilistic Data Association (PDA), Multiple Hypothesis Tracking (MHT) and Joint PDA which generalizes the PDA approach with a limit on the number of hypothesis used in MHT. In this work, we use a NN algorithm with a ranking on the priority of the measurements. We chose the NN algorithm because it is simple to implement and produces quick solutions. This algorithm works in the following way:

- At each time step, a camera measurement is first assigned to a radar measurement if Eulidean distance and the perpendicular distance (figure 6.2) of the radar measurement is within a

[1]This step only applies to radar data as the confidence information is only available for a radar track.

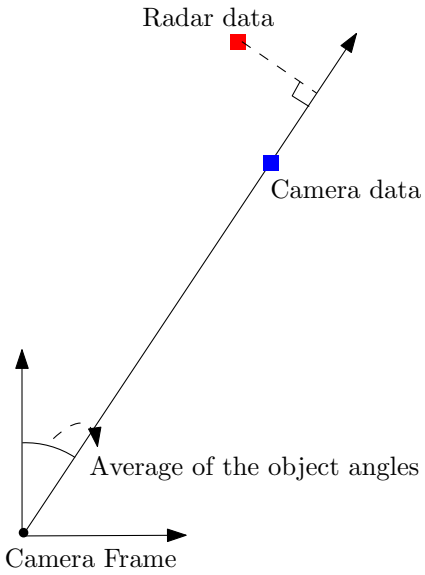

Figure 6.2: Computing the perpendicular distance of a radar measurement to the line corresponding to a camera measurement.

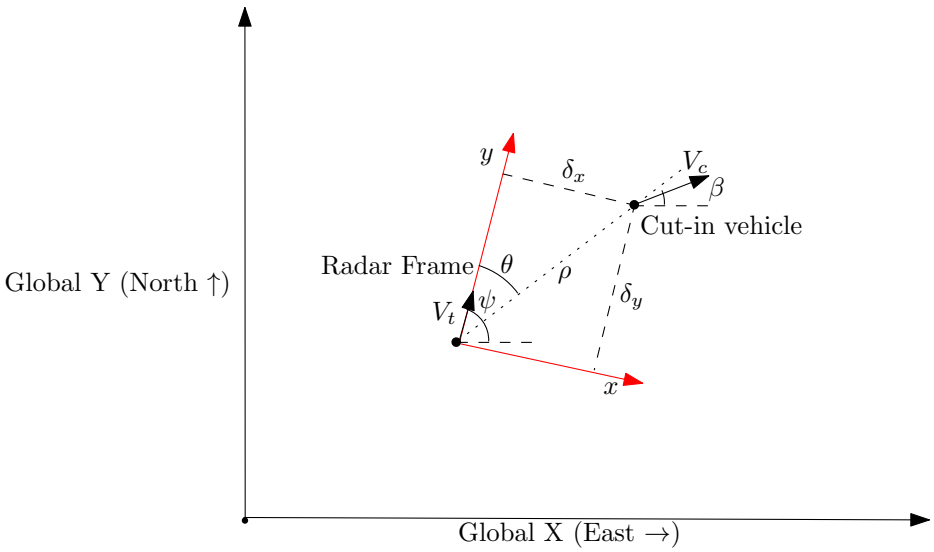

Figure 6.3: An illustration of the cut-in vehicle and the corresponding reference frames.

pre-defined limit from the camera measurement. Initial data processing suggested to us that the object distance and relative speed (or range-rate) estimates from the radar were more accurate while the object angle estimate from the camera was more reliable. Therefore, after a camera measurement is assigned to the radar measurement, a new fused measurement is

added which includes the object distance and object velocity from the radar, and the object angle from the camera. Also, once a fused measurement is added, the corresponding radar and camera measurement is removed from the system.

- After exhausting the possible assignments of the camera with the radar measurements, we may have un-assigned measurements from both radar and camera. Therefore, the measurements are ranked in the following order of descending priorities: The set of fused measurements, the set of un-assigned camera measurements and the set of un-assigned radar measurements. We do not impose any ordering within each of the sets. Un-assigned measurements from camera were given more priority because there was lot more noise in the radar data.

6.4 Equations of motion for the Cut-in Vehicle

Consider a cut-in vehicle traveling in front of a moving truck. Let (X_t, Y_t) denote the global position of the origin of the radar frame attached to the truck. Let the global position of the cut-in vehicle be represented as (X_c, Y_c). There are four states we would like to estimate in our problem: The lateral (δ_x) and the longitudinal (δ_y) position of the cut-in vehicle with respect to the radar frame, the speed (V_c) of the cut-in vehicle in the global frame and the heading angle (β) of the cut-in vehicle in the global frame. Each of the measurements obtained at the end of the algorithm discussed in the previous section has the following information: The lateral position (δ_x) of the cut-in vehicle with respect to the origin of the radar frame, distance of the cut-in vehicle (ρ) with respect to the origin of the radar frame, the relative speed of the cut-in vehicle ($\dot{\rho}$) with respect to the origin of the radar frame.

For simplicity, we only consider the kinematics of the vehicles. We assume the cut-in vehicle is traveling at a constant speed at a fixed heading angle, $i.e.$, $\dot{V}_c = 0$ and $\dot{\beta} = 0$. We assume the radar attached to the truck is moving at a constant velocity and turning with a constant yawrate, $i.e.$, $\dot{V}_t = 0$ and $\ddot{\psi} = 0$ (V_t is the speed and ψ is the heading angle of the truck as shown in figure 6.3).

6.4.1 State Equations

Note that $(X_c - X_t, Y_c - Y_t)$ are the global coordinates of the relative position of the cut-in vehicle with respect to the origin of the radar frame. These coordinates are related to (δ_x, δ_y) in the following way:

$$\begin{bmatrix} \delta_x \\ \delta_y \end{bmatrix} = \begin{bmatrix} \sin\psi & -\cos\psi \\ \cos\psi & \sin\psi \end{bmatrix} \begin{bmatrix} X_c - X_t \\ Y_c - Y_t \end{bmatrix}. \tag{6.1}$$

By differentiating the above equation, we get,

$$\begin{bmatrix} \dot{\delta}_x \\ \dot{\delta}_y \end{bmatrix} = \dot{\psi} \begin{bmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} X_c - X_t \\ Y_c - Y_t \end{bmatrix} + \begin{bmatrix} \sin\psi & -\cos\psi \\ \cos\psi & \sin\psi \end{bmatrix} \begin{bmatrix} \dot{X}_c - \dot{X}_t \\ \dot{Y}_c - \dot{Y}_t \end{bmatrix}. \tag{6.2}$$

Substituting $\dot{X}_c = V_c cos\beta$, $\dot{Y}_c = V_c \sin\beta$, $\dot{X}_t = V_t \cos\psi$, $\dot{Y}_t = V_t \sin\psi$, and further simplifying, we get

$$\begin{bmatrix} \dot{\delta}_x \\ \dot{\delta}_y \end{bmatrix} = \dot{\psi} \begin{bmatrix} \delta_y \\ -\delta_x \end{bmatrix} + \begin{bmatrix} \sin\psi & -\cos\psi \\ \cos\psi & \sin\psi \end{bmatrix} \begin{bmatrix} V_c \cos\beta - V_t \cos\psi \\ V_c \sin\beta - V_t \sin\psi \end{bmatrix}. \tag{6.3}$$

Summarizing all the equations for the state variables $\bar{x} = (\delta_x, \delta_y, V_c, \beta)$, we get,

$$\begin{bmatrix} \dot{\delta}_x \\ \dot{\delta}_y \\ \dot{V}_c \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} V_t + \begin{bmatrix} 0 & \dot{\psi} & \sin(\psi - \beta) & 0 \\ -\dot{\psi} & 0 & 0 & \cos(\psi - \beta) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_x \\ \delta_y \\ V_c \\ \beta \end{bmatrix}. \tag{6.4}$$

Note that in the above equations, V_t and ψ are inputs and are assumed to be known.

6.4.2 Output equations

The observations are represented using the vector $y = (\delta_x, \rho, \dot{\rho})$ where $\rho = \sqrt{\delta_x^2 + \delta_y^2}$. $\dot{\rho}$ can be expressed in the terms of the state variables and the inputs in the following way:

$$\begin{aligned} \dot{\rho} &= \frac{\delta_x}{\rho}\dot{\delta}_x + \frac{\delta_y}{\rho}\dot{\delta}_y \\ &= \frac{\delta_x}{\rho}(\dot{\psi}\delta_y + \sin(\psi - \beta)V_c) + \frac{\delta_y}{\rho}(-\dot{\psi}\delta_x + \cos(\psi - \beta)V_c) \\ &= \frac{\delta_x}{\rho}\sin(\psi - \beta)V_c + \frac{\delta_y}{\rho}\cos(\psi - \beta)V_c. \end{aligned} \tag{6.5}$$

6.5 Extended Kalman Filter Equations

The state equations in 6.4 can be represented as

$$\dot{\bar{x}} = f(\bar{x}, u).$$

Discretizing these equations with respect to time, we get,

$$\bar{x}_k = \underbrace{\bar{x}_{k-1} + \Delta t f(\bar{x}_{k-1}, u_{k-1})}_{\bar{f}(\bar{x}_{k-1}, u_{k-1})} \tag{6.6}$$

where $\bar{x}_k$ denotes the state of the system at time t_k, u_k denotes the input at time t_k and Δt represents the time interval between any two adjacent measurements. Similarly, the output equations for variables $y = (\delta_x, \rho, \dot{\rho})$ can also be discretized and written in the form $y_k = h(\bar{x}_k, u_k)$ where y_k represents the output of the system at time t_k.

As the above equations are non-linear, we linearize the state and output equations to obtain the corresponding Jacobian matrices at each step in the following way:

$$\frac{\partial \bar{f}}{\partial \bar{x}} = \begin{bmatrix} 1 & \Delta t\dot{\psi} & \Delta t\sin(\psi - \beta) & F_{14} \\ -\Delta t\dot{\psi} & 1 & \Delta t\cos(\psi - \beta) & F_{24} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$F_{14} = -\Delta t V_c \cos(\psi - \beta),$$

$$F_{24} = \Delta t V_c \sin(\psi - \beta),$$

and

$$\frac{\partial h}{\partial \bar{x}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{\delta_x}{\sqrt{\delta_x^2+\delta_y^2}} & \frac{\delta_y}{\sqrt{\delta_x^2+\delta_y^2}} & 0 & 0 \\ H_{31} & H_{32} & H_{33} & H_{34} \end{bmatrix},$$

$$\begin{aligned} H_{31} &= V_c \sin(\psi-\beta)\frac{\delta_y^2}{(\delta_x^2+\delta_y^2)^{\frac{3}{2}}} - \frac{\delta_y\delta_x}{(\delta_x^2+\delta_y^2)^{\frac{3}{2}}}(V_c\cos(\psi-\beta)-V_t), \\ H_{32} &= -V_c \sin(\psi-\beta)\frac{\delta_y\delta_x}{(\delta_x^2+\delta_y^2)^{\frac{3}{2}}} + \frac{\delta_x^2}{(\delta_x^2+\delta_y^2)^{\frac{3}{2}}}(V_c\cos(\psi-\beta)-V_t), \\ H_{33} &= \frac{\delta_x}{\sqrt{\delta_x^2+\delta_y^2}}\sin(\psi-\beta) + \frac{\delta_y}{\sqrt{\delta_x^2+\delta_y^2}}\cos(\psi-\beta), \\ H_{34} &= -\frac{\delta_x}{\sqrt{\delta_x^2+\delta_y^2}}V_c\cos(\psi-\beta) + \frac{\delta_y}{\sqrt{\delta_x^2+\delta_y^2}}V_c\sin(\psi-\beta) \end{aligned}$$

The EKF algorithm can be summarized as follows: The states are first propagated through the following equations to obtain an estimate for $\bar{x}_k$ based on $\bar{x}_{k-1}$. This estimate is denoted as $\hat{x}_{k|k-1} = \bar{f}(\hat{x}_{k-1}, u_{k-1})$. Let the state covariance matrix at time t_k be denoted as P_k. As the states are propagated, the covariance matrices also need to be updated suitably. Define

$$F_k = \frac{\partial \bar{f}}{\partial \bar{x}}|\hat{x}_{k-1} \tag{6.7}$$

$$H_k = \frac{\partial h}{\partial \bar{x}}|\hat{x}_{k|k-1}. \tag{6.8}$$

The new covariance matrix for the states obtained based on $\bar{x}_{k-1}$ and u_k is given by

$$P_{k|k-1} = F_{k-1}P_{k-1}F_{k-1}^T + Q_{k-1} \tag{6.9}$$

where Q_k is the covariance matrix associated with the noise in the states. Next, the Kalman Filter

gain matrix, K can be obtained as

$$K_k = P_{k|k-1}H_k^T(H_kP_{k|k-1}H_k^T + R_k)^{-1} \tag{6.10}$$

where R_k is the covariance matrix associated with the noise in the outputs. Once a measurement (y_k) is associated with a cut-in vehicle, the new estimated state and its covariance matrix at time t_k for the cut-in vehicle is given by

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k(y_k - H_k\hat{x}_{k|k-1}) \tag{6.11}$$

and

$$P_k = (I - K_kH_k)P_{k|k-1} \tag{6.12}$$

where I is the identity matrix.

6.6 Experimental Results

We first discuss the parameters used in the key steps of our algorithm for the experiments. In the pre-processing step, the minimum threshold for the confidence for retaining any radar measurement was set to 0.7. In the data association or data fusion step, a radar measurement is associated with a camera measurement if the distance between the measurements is less than a pre-defined limit (=5 meters). Also, any measurement or an object that is not associated with a new measurement for more than 0.5 secs is deleted from the system. An object can be associated with a fused measurement if the Euclidean distance between them is less than 5 meters. Finally, for the EKF implementation, the following are the covariance matrices used in the experiments:

$$Q_k = \frac{\Delta t^2}{0.1^2}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix}$$

and

$$R_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

A cut-in vehicle is said to be present in the field of view of the sensing system of the truck if it is present in the region of interest of the truck as shown in figure 6.4. In the experiments, we chose the *width limit* to be equal to 10 meters and the *distance limit* to be equal to 150 meters.

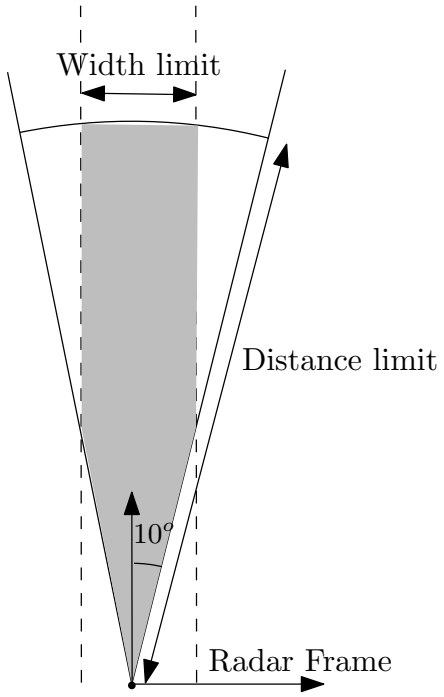

Figure 6.4: Region of interest in front of the truck.

In the field experiments we performed, there was one cut-in vehicle in between two moving trucks. The sensor fusion algorithms presented here were implemented in the following truck to identify the states of the cut-in vehicle as it changed lanes. The exact position of the center of mass (CM) of the cut-in vehicle was measured using a Piksi GPS (which has 2 cm accuracy).

The estimated states of the cut-in vehicle from the moving truck are shown in figures 6.5-6.8. The average error in the estimation of δ_x, δ_y, V_c and β was $0.68\ m$, $1.92\ m$, $0.25\ m/sec$ and 0.11 radians respectively. Note that the radar and the camera measurements we obtain are likely associated with the rear end of the car. This discrepancy could explain the error we mainly observed in δ_y. The heading angle of the cut-in vehicle was the most challenging to estimate as there is no direct measurement of this angle and this difficulty can be observed in figure 6.8.

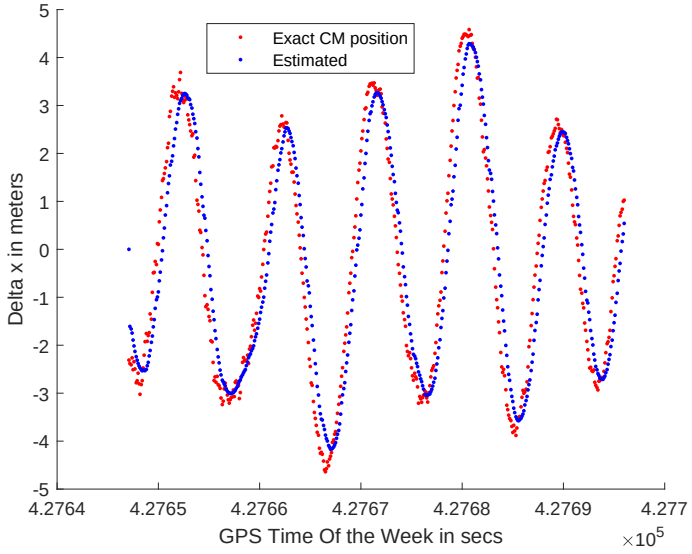

Figure 6.5: Estimates of δ_x compared with the exact x values of the center of mass of the vehicle.

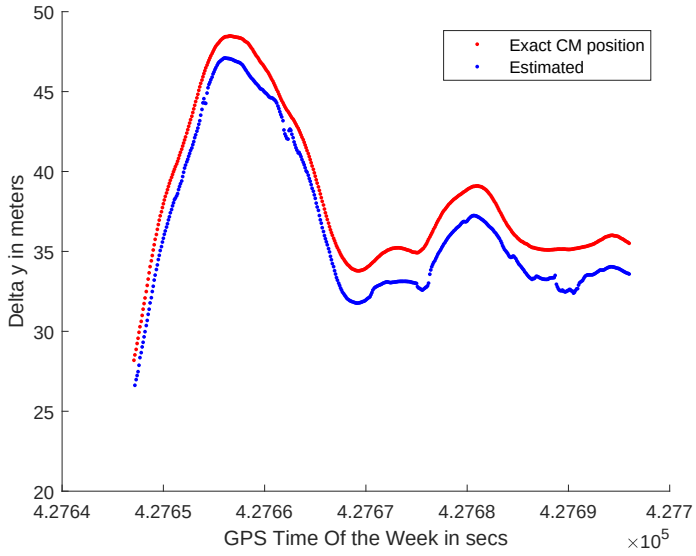

Figure 6.6: Estimates of δ_y compared with the exact y values of the center of mass of the vehicle.

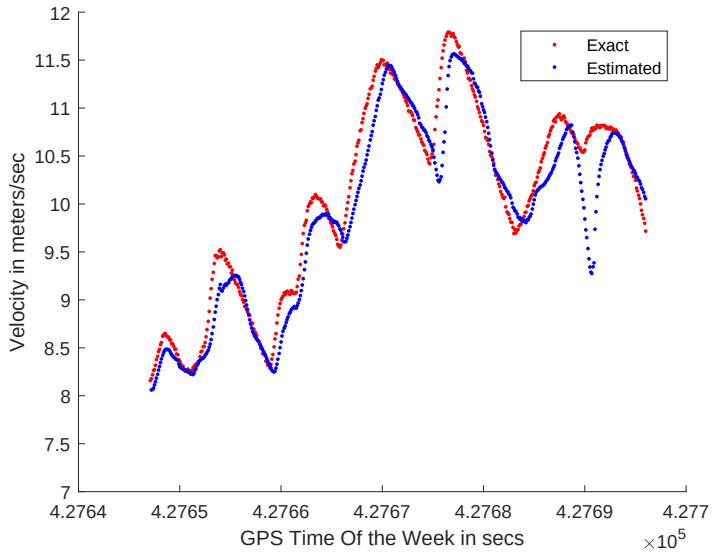

Figure 6.7: Estimates of V_c compared with the exact speed of the vehicle.

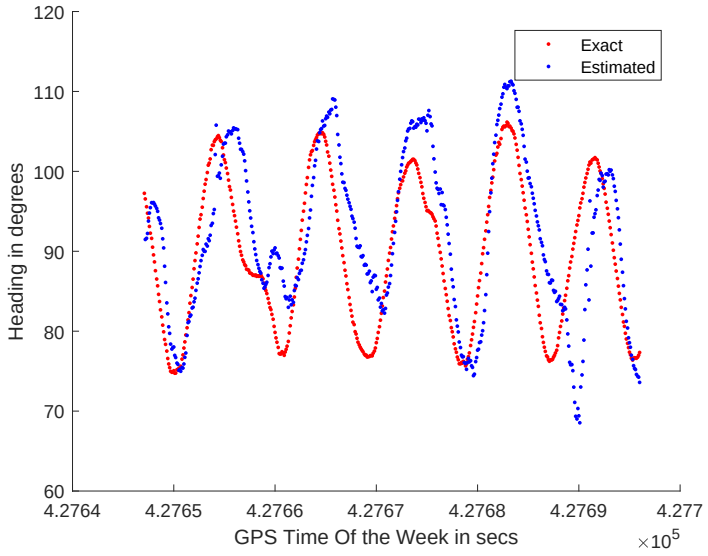

Figure 6.8: Estimates of β compared with the exact yaw angle of the vehicle.

7. CONCLUSION AND FUTURE WORK

7.1 Conclusion

In this work, we first considered the problem of controlling the lateral motion of a vehicle traveling at a constant speed based on the sampled GPS coordinates of the preceding vehicle. We deployed an algorithm in Chapter 2 to compute the curvature of the trajectory based on the limited preview, and used it for constructing feedforward steering input. Using a fixed structure feedback controller synthesis, we constructed the steering feedback input to enable a vehicle to track the desired trajectory. Experimental results corroborate the effectiveness of the design procedure.

In this work, we also considered the problem of controlling the lateral motion of a convoy of autonomous vehicles; lateral control of an autonomous following vehicle is based on the sampled GPS data from its lead and preceding vehicles. We presented two controller architectures to accommodate (and fuse) the communicated data. In either case, we considered data that corresponded to a position of the lead/preceding vehicle within a specified distance of the ego vehicle in the convoy. In the first architecture, a composite ELC trajectory was constructed as a circular arc spline by treating both data streams as if they originate from the same source. This is simpler in terms of implementation as there is only one trajectory to be computed and only one set of feedback errors and corresponding feedback and feedforward actions to be computed. In the second architecture, a trajectory each was constructed for each ego vehicle as a circular spline that corresponded to the data from its preceding and lead vehicles respectively. Two sets of feedback errors and control actions were then combined with appropriate weights to synthesize control actions. Simulation and experimental results corroborate the effectiveness of the control schemes. The second scheme was found to be slightly better in terms of its performance in simulation; experimentation will be conducted in the near future once the prevailing COVID restrictions on experimentation are relaxed. In simulations, and in the experimental tests conducted so far, string instability was not observed thus far.

A sensor fusion algorithm was presented in this work Chapter 6 to fuse radar and camera data obtained from sensors on-board a moving truck to identify the presence of a cut-in vehicle. A nearest neighbor algorithm with a priority on the ordering of fused measurements is used to first fuse radar and vision data, and then a greedy algorithm is used to assign a fused measurement with the cut-in vehicle state. The position and velocity estimates of the cut-in vehicle is estimated using the EKF algorithm ,and the overall proposed approach has been experimentally tested.

The algorithms presented in this article are general and are useful in other applications of platooning also. For example, the hitch angle between the tractor and trailer of a heavy truck (say A) can be estimated by communicating with the truck (B) immediately following the trailer of A in a platoon. Specifically, the fused information of the rear end of the trailer can be computed in B and this information can be sent to the tractor of A through Dedicated Short Range Communication (DSRC). Sensor fusion algorithms similar to the one proposed in this article can be easily implemented to estimate the hitch angle of A. This estimation is quite useful for accurately controlling A and can augment any sensing systems already present in A.

7.2 Future work

In this work, we design an effective lateral controller for an autonomous vehicle firstly. Though we made a "constant" longitudinal velocity assumption and there is no longitudinal considered in our work. A proper longitudinal vehicle velocity controller is necessary and will contribute to maintain longitudinal speed and distance, especially in the multiple vehicle control scenario. Then longitudinal-lateral vehicle coupling issue should be discussed further.

In controller design for a convoy of autonomous vehicles, we only considered vehicles in the convoy. In the future, we should consider vehicles vehicles in adjacent lanes. And other complex scenarios.

In "cut-in" vehicle identification chapter, future work can include removing some of the assumptions used in this part with respect to the dynamics of the moving truck and the cut-in vehicle. Adding more sensors will also provide a larger field of view ,which will enable more accurate cut-in vehicle identification algorithms.

REFERENCES

[1] M. Liu, S. Rathinam, and S. Darbha, "Lateral control of an autonomous and connected vehicle with limited preview information," *IEEE Transactions on Intelligent Vehicles*, 2020. Under review.

[2] R. Fenton, G. Melocik, and K. Olson, "On the steering of automated vehicles: Theory and experiment," *IEEE Transactions on Automatic Control*, vol. 21, no. 3, pp. 306–315, 1976.

[3] S. Malan, M. Milanese, P. Borodani, and A. Gallione, "Lateral control of autonomous electric cars for relocation of public urban mobility fleet," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 590–598, 2007.

[4] Han-Shue Tan, J. Guldner, S. Patwardhan, Chieh Chen, and B. Bougler, "Development of an automated steering vehicle based on roadway magnets-a case study of mechatronic system design," *IEEE/ASME Transactions on Mechatronics*, vol. 4, no. 3, pp. 258–272, 1999.

[5] Han-Shue Tan, B. Bougler, J. A. Farrell, and Yunchun Yang, "Automatic vehicle steering controls: Dgps/ins and magnetic markers," in *Proceedings of the 2003 American Control Conference, 2003.*, vol. 1, pp. 60–65, 2003.

[6] H. Peng and M. Tomizuka, "Preview control for vehicle lateral guidance in highway automation.," *Journal of Dynamic Systems, Measurement, and Control*, vol. 4, no. 115, pp. 679–689, 1993.

[7] N. Kapania and J. Gerdes, "Design of a feedback-feedforward steering controller for accurate path tracking and stability at the limits of handling.," *Vehicle System Dynamics*, vol. 53, no. 12, pp. 1687–1704, 2015.

[8] P. T. Ho, M. L.and Chan and A. B. Rad, "Lane change algorithm for autonomous vehicles via virtual curvature method," *Journal of Advanced Transportation*, vol. 43, no. 1, pp. 47–70, 2007.

[9] Q. N. Li and H. Cheng, "Springrobot: A prototype autonomous vehicle and its algorithms for lane detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 4, pp. 300–308, 2004.

[10] Y. Yim and S. Oh, "Three-feature based automatic lane detection algorithm (tfalda) for autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, pp. 219–225, 2003.

[11] Ching-Yao Chan and Han-Shu Tan, "Application of a robust steering controller in emergency situations," in *Proceedings 199 IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems (Cat. No.99TH8383)*, pp. 94–99, 1999.

[12] S. E. Shladover, C. A. Desoer, J. K. Hedrick, M. Tomizuka, J. Walrand, W. . Zhang, D. H. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown, "Automated vehicle control developments in the path program," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 114–130, 1991.

[13] S. E. Shladover, "Review of the state of development of advanced vehicle control systems (avcs)," *Vehicle System Dynamics*, vol. 24, no. 6-7, pp. 551–595, 1995.

[14] D. Bevly, X. Cao, M. Gordon, G. Ozbilgin, D. Kari, B. Nelson, J. Woodruff, M. Barth, C. Murray, A. Kurt, K. Redmill, and U. Ozguner, "Lane change and merge maneuvers for connected and automated vehicles: A survey," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 105–120, 2016.

[15] H. Zheng, J. Zhou, Q. Shao, and Y. Wang, "Investigation of a longitudinal and lateral lane-changing motion planning model for intelligent vehicles in dynamical driving environments," *IEEE Access*, vol. 7, pp. 44783–44802, 2019.

[16] D. Meek and D. Walton, "Approximation of discrete data by g1 arc splines," *Computer-Aided Design*, pp. 301–306, 1992.

[17] J. Funke and J. Christian Gerdes, "Simple Clothoid Lane Change Trajectories for Automated Vehicles Incorporating Friction Constraints," *Journal of Dynamic Systems, Measurement, and Control*, vol. 138, 12 2015. 021002.

[18] K. Jakubczyk, "Approximation of smooth planar curves by circular arc splines," 2012.

[19] K. Marciniak and B. Putz, "Approximation of spirals by piecewise curves of fewest circular arc segments," *Computer-Aided Design*, vol. 16, no. 2, pp. 87–90, 1984.

[20] D. Meek and D. Walton, "Approximating smooth planar curves by arcs splines," *Journal of Computational and Applied Mathematics*, vol. 59, no. 2, pp. 44783–44802, 1995.

[21] R. L. Cheu, "Highway geometric design," in *The Handbook of Highway Engineering* (T. F. Fwa, ed.), ch. 6, pp. 1–15, CRC Taylor & Francis Group, 1 ed., 2006.

[22] A. Bhattacharyya, S. P.; Datta and L. H. Keel, *Linear control theory: structure, robustness, and optimization*. CRC press, 2018.

[23] Y. Shui, "Strategic trajectory planning of highway lane change maneuver with longitudinal speed control," Master's thesis, The Ohio State University, 1995.

[24] Suzanne E. Lee, Erik C.B. Olsen, and Walter W. Wierwille, "A comprehensive examination of naturalistic lane changes," Tech. Rep. DOT HS 809 702, National Highway Traffic Safety Administration, March 2004.

[25] M. Liu, S. Rathinam, and S. Darbha, "Lateral control of an autonomous car with limited preview information," in *18th European Control Conference (ECC)*, pp. 3192–3197, 2019.

[26] M. Liu, S. Rathinam, and S. Darbha, "Lateral control of a convoy of vehicles with a limited preview information," in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications II* (T. Pham, L. Solomon, and K. Rainey, eds.), vol. 11413, pp. 547 – 558, International Society for Optics and Photonics, SPIE, 2020.

[27] S. Ashok and G. Samuel, "Least square curve fitting technique for processing time sampled high speed spindle data.," *International Journal of Manufacturing Research*, vol. 6, no. 3, p. 256, 2011.

[28] N. Chernov and C. Lesort, "Least squares fitting of circles and lines," 2003.

[29] H. Guldner, J.; Tan and S. Patwardhan, "Analysis of automatic steering control for highway vehicles with look-down lateral reference systems," *Vehicle System Dynamics*, vol. 26, no. 4, pp. 243–269, 1996.

[30] P. Hingwe and M. Tomizuka, "Experimental evaluation of a chatter free sliding mode control for lateral control in ahs," in *Proceedings of the 1997 American Control Conference (Cat. No.97CH36041)*, vol. 5, pp. 3365–3369, 1997.

[31] Wonshik Chee, M. Tomizuka, S. Patwardhan, and Wei-Bin Zhang, "Experimental study of lane change maneuver for ahs applications," in *Proceedings of American Control Conference*, vol. 1, pp. 139–143, 1995.

[32] H. A. Pham and J. K. Hedrick, "A robust optimal lateral vehicle control strategy," in *Proceeding of the IEEE International Conference on Control Applications held together with IEEE International Symposium on Intelligent Control*, vol. 1, pp. 361–366, 1996.

[33] Y. Gao, *Model Predictive Control for Autonomous and Semiautonomous Vehicles*. PhD thesis, University of California, Berkeley, 2014.

[34] A. G. Ulsoy, H. Peng, and M. Çakmakci, *Automotive Control Systems*. Cambridge University Press, 2012.

[35] C. Hatipoglu, "Lateral control for highway automation," Master's thesis, The Ohio State University, 1995.

[36] Meihua Tai, Pushkar Hingwe, and M. Tomizuka, "Modeling and control of steering system of heavy vehicles for automated highway systems," *IEEE/ASME Transactions on Mechatronics*, vol. 9, no. 4, pp. 609–618, 2004.

[37] S. Rao, *Mechanical Vibration*. Prentice Hall, 2011.

[38] K. Kritayakirana, *Autonomous Vehicle Control at the Limits of Handling*. PhD thesis, Stanford University, 2012.

[39] M. Tai and M. Tomizuka, "Feedforward compensation for lateral control of heavy vehicles for automated highway system (ahs)," *IFAC Proceedings Volumes*, vol. 35, no. 1, pp. 373 – 378, 2002. 15th IFAC World Congress.

[40] L. Segel, "An investigation of automobile handling as implemented by a variable-steering automobile.," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 6, no. 4, pp. 333–341, 1964.

[41] D. Crolla and D. Cao, "The impact of hybrid and electric powertrains on vehicle dynamics, control systems and energy regeneration.," *Vehicle System Dynamics*, vol. 50, pp. 95–109, 2012.

[42] L. H. Keel and S. P. Bhattacharyya, "Robust, fragile or optimal?," in *Proceedings of the 1997 American Control Conference (Cat. No.97CH36041)*, vol. 2, pp. 1307–1313 vol.2, 1997.

[43] W. A. Malik, S. Darbha, and S. P. Bhattacharyya, "A linear programming approach to the synthesis of fixed-structure controllers," *IEEE Transactions on Automatic Control*, vol. 53, no. 6, pp. 1341–1352, 2008.

[44] M. Henrion, D.; Sebek and V. Kucera, "Positive polynomials and robust stabilization with fixed-order controllers.," *IEEE Transactions on Automatic Control*, vol. 48, no. 7, pp. 1178–1186, 2003.

[45] D. D. Siljak, "Parameter space methods for robust control design: a guided tour," *IEEE Transactions on Automatic Control*, vol. 34, no. 7, pp. 674–688, 1989.

[46] J. S. Karmarkar and D. D. Siljak, "Stability analysis of systems with time delay," *Proceedings of the Institution of Electrical Engineers*, vol. 117, no. 7, pp. 1421–1424, 1970.

[47] M. A. Akram, P. Liu, M. O. Tahir, W. Ali, and Y. Wang, "A state optimization model based on kalman filtering and robust estimation theory for fusion of multi-source information in highly non-linear systems.," *SENSORS*, vol. 19, no. 7, n.d.

[48] H. K. Khalil, *Nonlinear systems*, vol. 3. Prentice hall Upper Saddle River, NJ, 2002.

[49] H. K. Khalil, *Nonlinear systems: solution manual*, vol. 3. New York: Macmillan., 1992.

[50] D. Swaroop and J. K. Hedrick, "String stability of interconnected systems," *IEEE Transactions on Automatic Control*, vol. 41, no. 3, pp. 349–357, 1996.

[51] Guang Lu and M. Tomizuka, "A practical solution to the string stability problem in autonomous vehicle following," in *Proceedings of the 2004 American Control Conference*, vol. 1, pp. 780–785 vol.1, 2004.

[52] J. Alleleijn, H. Nijmeijer, S. Öncü, and J. Ploeg, *Lateral string stability of vehicle platoons*. D &;C, Eindhoven University of Technology, 2014. Internship report.

[53] C. J. Taylor, J. Kosecka, R. Blasi, and J. Malik, "A comparative study of vision-based lateral control strategies for autonomous highway driving," *The International Journal of Robotics Research*, vol. 18, no. 5, pp. 442–453, 1999.

[54] T. Toledo and D. Zohar, "Modeling duration of lane changes," *Transportation Research Record*, vol. 1999, no. 1, pp. 71–78, 2007.

[55] S. Company, *Piksi Multi-Product Summary*.

[56] I. Papadimitriou, Gaung Lu, and M. Tomizuka, "Autonomous lateral following consideration for vehicle platoons," in *Proceedings 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2003)*, vol. 1, pp. 401–406 vol.1, 2003.

[57] I. Papadimitriou and M. Tomizuka, "Lateral control of platoons of vehicles on highways: The autonomous following based approach," *Int. J. of Vehicle Design*, vol. 36, pp. 24 – 37, 01 2004.

[58] S. Solyom, A. Idelchi, and B. B. Salamah, "Lateral control of vehicle platoons," in *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 4561–4565, 2013.

[59] D. E. D. Delphi Automotive Systems 2017.

[60] C. Mertz, L. E. Navarro-Serment, R. MacLachlan, P. Rybski, A. Steinfeld, A. Suppé, C. Urmson, N. Vandapel, M. Hebert, C. Thorpe, D. Duggins, and J. Gowdy, "Moving object detection with laser scanners," *J. Field Robot.*, vol. 30, pp. 17–43, Jan. 2013.

[61] C. Premebida, O. Ludwig, and U. Nunes, "Lidar and vision-based pedestrian detection system," *J. Field Robot.*, vol. 26, pp. 696–711, Sept. 2009.

[62] M. Hoy, C. Weng, J. Yuan, and J. Dauwels, "Bayesian tracking of multiple objects with vision and radar," in *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 1–6, Nov 2016.

[63] M. L. Christophe DELEVAL, *Multiple object tracking combining camera and radar*. PhD thesis, Université catholique de Louvain, 2017.

[64] H. Cho, Y. W. Seo, B. V. K. V. Kumar, and R. R. Rajkumar, "A multi-sensor fusion system for moving object detection and tracking in urban driving environments," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1836–1843, May 2014.

[65] M. Michaelis, P. Berthold, D. Meissner, and H. J. Wuensche, "Heterogeneous multi-sensor fusion for extended objects in automotive scenarios using gaussian processes and a gmphd-filter," in *2017 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pp. 1–6, Oct 2017.

APPENDIX A

THE DERIVATION OF CLOSED-LOOP CHARACTERISTIC POLYNOMIAL

In Chapter 3, we defined the transfer function between tire steering command angle(δ_c) and tire steering angle(δ_f) is:

$$H_a(s) = \frac{\delta_f}{\delta_c} = \frac{N_a(s)}{D_a(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}. \tag{A.1}$$

Recall equation (3.26) we can elaborate it as:

$$\begin{bmatrix} m & 0 \\ 0 & I_z \end{bmatrix} \begin{bmatrix} \ddot{e}_{lat} \\ \ddot{\tilde{\theta}} \end{bmatrix} + \begin{bmatrix} \frac{C_f+C_r}{V_0} & \frac{aC_f-bC_r}{V_0} \\ \frac{aC_f-bC_r}{V_0} & \frac{a^2C_f+b^2C_r}{V_0} \end{bmatrix} \begin{bmatrix} \dot{e}_{lat} \\ \dot{\tilde{\theta}} \end{bmatrix} + \begin{bmatrix} 0 & -(C_f+C_r) \\ 0 & -(aC_f - bCr) \end{bmatrix} \begin{bmatrix} e_{lat} \\ \tilde{\theta} \end{bmatrix}$$
$$= \begin{bmatrix} 1 \\ a \end{bmatrix} C_f\delta_f - \frac{1}{R} \begin{bmatrix} aC_f - bC_r + mV_0^2 \\ a^2C_f + b^2C_r \end{bmatrix} \tag{A.2}$$

Combine the steering command transfer function $H_a(s)$ and take Laplace transformation, the equation above is :

$$\underbrace{\begin{bmatrix} ms^2 + \frac{C_f+C_r}{V_0}s & \frac{aC_f-bC_r}{V_0}s - (C_f + Cr) \\ \frac{aC_f-bC_r}{V_0}s & Is^2 + \frac{a^2C_f+b^2C_r}{V_0}s - (aC_f - bCr) \end{bmatrix}}_{A} \begin{bmatrix} E_{lat}(s) \\ \tilde{\Theta}(s) \end{bmatrix} \tag{A.3}$$

$$= -\underbrace{\begin{bmatrix} 1 \\ a \end{bmatrix}}_{B} \underbrace{C_f H_a(s) \begin{bmatrix} k_e & k_\omega s + k_\theta \end{bmatrix}}_{C} \begin{bmatrix} E_{lat}(s) \\ \tilde{\Theta}(s) \end{bmatrix} \tag{A.4}$$

The characteristic polynomial is defined by the determinant of:

$$det(A + BC) = 0 \tag{A.5}$$

By Sylvester's determinant theorem, the above determinant can be calculated fast as:

$$det(A(I + A^{-1}BC)) = det(A)(1 + CA^{-1}B) = det(A) + CA_{cof}B = det(A) + H_a(s)\tilde{Q}(s) \quad \text{(A.6)}$$

where A_{cof} is the cofacotor of matrix A.

$$A_{cof} = \begin{bmatrix} Is^2 + \frac{a^2C_f + b^2C_r}{V_0}s - (aC_f - bCr) & -\frac{aC_f - bC_r}{V_0}s + (C_f + Cr) \\ -\frac{aC_f - bC_r}{V_0}s & ms^2 + \frac{C_f + C_r}{V_0}s \end{bmatrix}$$

and

$$\begin{aligned} \tilde{Q}(s) &= C_f \begin{bmatrix} k_e & k_\omega s + k_\theta \end{bmatrix} A_{cof} \begin{bmatrix} 1 \\ a \end{bmatrix} \\ &= C_f k_e (Is^2 + \frac{(a+b)bC_r}{V_0}s + (a+b)C_r) + C_f(k_\omega s + k_\theta)(mas^2 + \frac{(a+b)C_r}{V_0}s) \end{aligned}$$

where the $\det(A)$ is also the determinant of open-loop of transfer function, so

$$\begin{aligned} det(A) &= \Delta_o(s) \\ &= mIs^4 + s^3[\frac{C_f(I + a^2m) + C_r(I + b^2m)}{V_0}] + s^2[\frac{(a+b)^2C_fC_r}{V_0} - m(aC_f - bC_r)] \end{aligned}$$

Hence, the characteristic polynomial of system with EPS dynamics can be present as:

$$\begin{aligned} \Delta(s; \mathbf{K}) &= (s^2 + 2\zeta\omega_n s + \omega_n^2)\Delta_o(s) + k_e\omega_n^2 C_f(Is^2 + \frac{b(a+b)C_r}{V_0}s + (a+b)C_r) \\ &\quad + (k_\theta + k_w s)\omega_n^2 C_f(mas^2 + \frac{(a+b)C_r}{V_0}s). \end{aligned} \quad \text{(A.7)}$$

Note that $\Delta(s, \mathbf{K}) = A_6 s^6 + A_5 s^5 + A_4 s^4 + A_3 s^3 + A_2 s^2 + A_1 s + A_0$, where

$$
\begin{aligned}
A_6 &= \frac{Im}{\omega_n^2}, \\
A_5 &= \frac{2Im\zeta}{\omega_n} + \frac{C_f(I + a^2 m) + C_r(I + b^2 m)}{V_0 \omega_n^2}, \\
A_4 &= \frac{2\zeta(C_f(I + a^2 m) + C_r(I + b^2 m))}{V_0 \omega_n} \\
&+ (\frac{(a+b)^2 C_f C_r}{V_0^2 \omega_n^2} - m(aC_f - bC_r)) + mI, \\
A_3 &= (\frac{2\zeta((a+b)^2 C_f C_r)}{V_0^2 \omega_n} - m(aC_f - bC_r)) \\
&+ \frac{C_f(I + a^2 m) + C_r(I + b^2 m)}{V_0} + C_f m a k_\omega, \\
A_2 &= (\frac{(a+b)^2 C_f C_r}{V_0^2} - m(aC_f - bC_r)) \\
&+ (C_f I k_e + C_f C_r \frac{(a+b)}{V_0} k_\omega + maC_f k_\theta), \\
A_1 &= C_f C_r \frac{(a+b)}{V_0}(bk_e + k_\theta), \\
A_0 &= C_f C_r (a+b) k_e,
\end{aligned}
$$

APPENDIX B

DERIVATION OF STABILIZING GAIN SETS

Firstly, recall the characteristic polynomial of system with EPS dynamics:

$$\begin{aligned}\Delta(s;\mathbf{K}) &= (s^2+2\zeta\omega_n s+\omega_n^2)\Delta_o(s)+k_e\omega_n^2C_f(Is^2+\frac{b(a+b)C_r}{V_0}s+(a+b)C_r)\\ &\quad+(k_\theta+k_ws)\omega_n^2C_f(mas^2+\frac{(a+b)C_r}{V_0}s). \qquad \text{(B.1)}\end{aligned}$$

Note that $\Delta(s,\mathbf{K})=A_6s^6+A_5s^5+A_4s^4+A_3s^3+A_2s^2+A_1s+A_0$,

where

$$\begin{aligned}A_6 &= \frac{Im}{\omega_n^2},\\ A_5 &= \frac{2Im\zeta}{\omega_n}+\frac{C_f(I+a^2m)+C_r(I+b^2m)}{V_0\omega_n^2},\\ A_4 &= \frac{2\zeta(C_f(I+a^2m)+C_r(I+b^2m))}{V_0\omega_n}\\ &+ (\frac{(a+b)^2C_fC_r}{V_0^2\omega_n^2}-m(aC_f-bC_r))+mI,\\ A_3 &= (\frac{2\zeta((a+b)^2C_fC_r)}{V_0^2\omega_n}-m(aC_f-bC_r))\\ &+ \frac{C_f(I+a^2m)+C_r(I+b^2m)}{V_0}+C_fmak_\omega,\\ A_2 &= (\frac{(a+b)^2C_fC_r}{V_0^2}-m(aC_f-bC_r))\\ &+ (C_fIk_e+C_fC_r\frac{(a+b)}{V_0}k_\omega+maC_fk_\theta),\\ A_1 &= C_fC_r\frac{(a+b)}{V_0}(bk_e+k_\theta),\\ A_0 &= C_fC_r(a+b)k_e,\end{aligned}$$

Since $A_6>0, A_5>0$ and $A_4>0$, a necessary condition for stability is that $A_3>0, A_2>$

$0, A_1 > 0, A_0 > 0$, yielding the conditions:

$$k_\omega > \frac{\frac{2\zeta}{\omega_n}\left(\frac{(a+b)^2 C_f C_r}{V_0^2} - m(aC_f - bC_r)\right) + \frac{C_f(I+a^2m)+C_r(I+b^2m)}{V_0}}{C_f a m}, \tag{B.2}$$

$$\begin{aligned} C_f I k_e + C_f C_r \frac{(a+b)}{V_0} k_\omega + maC_f k_\theta &> -\frac{(a+b)^2 C_f C_r}{V_0^2} - m(aC_f - bC_r), && \text{(B.3)} \\ k_\theta &> -bk_e, && \text{(B.4)} \\ k_e &> 0 && \text{(B.5)} \end{aligned}$$

Secondly, we rewrite characteristic equation with gain sets (k_e, k_θ and k_ω) with ζ and ω_n:

$$\begin{aligned} &\Delta(s, \mathbf{K}) = \\ &A_6 s^6 + A_5 s^5 + A_4 s^4 + A_3 s^3 + A_2 s^2 + A_1 s + A_0 \\ &= \underbrace{\frac{\alpha_5}{\omega_n^2} s^6 + (\frac{\alpha_4}{\omega_n^2} + \frac{2\zeta\alpha_5}{\omega}) s^5 + (\frac{\alpha_1}{\omega_n^2} + \frac{2\zeta\alpha_4}{\omega} + \alpha_5) s^4 + (\frac{2\zeta\alpha_1}{\omega} + \alpha_4) s^3 + \alpha_1 s^2}_{D_1} \\ &+ k_e \underbrace{\left((\alpha_2) s^2 + (\frac{\alpha_0}{V_0} b) s + \alpha_0\right)}_{D_2} + k_\theta \underbrace{\left((\alpha_3) s^2 + \frac{\alpha_0}{V_0} s)\right)}_{D_3} + k_\omega \underbrace{\left((\alpha_3) s^3 + \frac{\alpha_0}{V_0} s^2)\right)}_{D_4} \end{aligned}$$

where

$$\begin{aligned} \alpha_5 = Im, \alpha_4 = \frac{C_f(ma^2 + I) + C_r(mb^2 + I)}{V_0} \\ \alpha_3 = C_f a m, \alpha_2 = C_f I, \\ \alpha_1 = \frac{C_f C_r (a+b)^2}{V_0^2} - m(C_f a - C_r b), \alpha_0 = (a+b) C_f C_r, \end{aligned}$$

Thirdly, let us use standard even-odd decomposition of polynomials D_1,D_2,D_3 and D_4:

$$D_1 = D_{1e} + sD_{1o} \tag{B.6}$$

$$D_2 = D_{2e} + sD_{2o} \tag{B.7}$$

$$D_3 = D_{3e} + sD_{3o} \tag{B.8}$$

$$D_4 = D_{4e} + sD_{4o} \tag{B.9}$$

where

$$\begin{aligned}
D_{1e} &= \frac{\alpha_5}{\omega_n^2}s^6 + (\frac{\alpha_1}{\omega_n^2} + \frac{2\zeta\alpha_4}{\omega_n} + \alpha_5)s^4 + \alpha_1 s^2, \\
D_{1o} &= (\frac{\alpha_4}{\omega_n^2} + \frac{2\zeta\alpha_5}{\omega_n})s^5 + (\frac{2\zeta\alpha_1}{\omega_n} + \alpha_4)s^3, \\
D_{2e} &= (\alpha_2)s^2 + \alpha_0, D_{2o} = (\frac{\alpha_0}{V_0}b)s, \\
D_{3e} &= (\alpha_3)s^2, D_{3o} = (\frac{\alpha_0}{V_0}s), \\
D_{4e} &= \frac{\alpha_0}{V_0}s^2, D_{4o} = (\alpha_3)s^3
\end{aligned}$$

With $s = j\omega$, we have:

$$\begin{aligned}
\Delta(j\omega, \mathbf{K}) = D_{1e}(\omega^2) + k_e D_{2e}(\omega^2) + k_\theta D_{3e}(\omega^2) + k_\omega D_{4e}(\omega^2) \\
+j\omega(D_{1o}(j\omega) + k_e D_{2o}(j\omega) + k_\theta D_{3o}(j\omega) + k_\omega D_{4o}(j\omega))
\end{aligned} \tag{B.10}$$

still same, we can represent the even and odd terms with ζ and ω_n terms:

$$\begin{aligned}
D_{1e}(j\omega) &= -\frac{\alpha_5}{\omega_n^2}w^6 + (\frac{\alpha_1}{\omega_n^2} + \frac{\alpha_4}{\omega_n} + \alpha_5)w^4 - (\alpha_1)w^2, \\
D_{1o}(j\omega) &= (\frac{\alpha_4}{\omega} + \frac{2\zeta\alpha_5}{\omega_n})w^4 - (\frac{2\zeta\alpha_1}{\omega} + \alpha_4)w^2, \\
D_{2e}(j\omega) &= -(\alpha_2)w^2 + \alpha_0, D_{2o}(j\omega) = (\frac{\alpha_0}{v_0}b) \\
D_{3e}(j\omega) &= -(\alpha_3)w^2, D_{3o} = \frac{\alpha_0}{V_0} \\
D_{4e}(j\omega) &= -\frac{\alpha_0}{V_0}w^2, D_{4o} = -(\alpha_1)w^2
\end{aligned}$$

The real roots boundary is given by:

$$\Delta(0) = 0 \tag{B.11}$$

$$A_0 = (a+b)C_f C_r k_e = 0 \tag{B.12}$$

The solution will be $k_e = 0$. If k_e is zero, then the characteristic equation can not be Hurwitz because of the violation of stabilizing condition B.5. So this boundary can be skipped.

The complex roots boundary are given by:

$$\Delta(j\omega) = 0, \omega \in (0, \infty) \tag{B.13}$$

For $\omega > 0$, we have

$$D_{1e}(j\omega) + k_e D_{2e}(j\omega) + k_\theta D_{3e}(j\omega) + k_\omega D_{4e}(j\omega) = 0 \tag{B.14}$$

$$D_{1o}(j\omega) + k_e D_{2o}(j\omega) + k_\theta D_{3o}(j\omega) + k_\omega D_{4o}(j\omega) = 0 \tag{B.15}$$

Rewrite above equations into matrix form we have:

$$\underbrace{\begin{bmatrix} D_{3e}(j\omega) & D_{4e}(j\omega) \\ D_{3o}(j\omega) & D_{4o}(j\omega) \end{bmatrix}}_{A_{\theta\omega}} \begin{bmatrix} k_\theta \\ k_\omega \end{bmatrix} = \begin{bmatrix} -D_{1e}(j\omega) - k_e D_{2e}(j\omega) \\ -D_{1o}(j\omega) - k_e D_{2o}(j\omega) \end{bmatrix} \tag{B.16}$$

So the gains k_θ and k_ω can be represent as :

$$\begin{bmatrix} k_\theta \\ k_\omega \end{bmatrix} = \frac{1}{|A_{\theta\omega}|} \begin{bmatrix} D_{4o}(j\omega) & -D_{4e}(j\omega) \\ -D_{3o}(j\omega) & D_{3e}(j\omega) \end{bmatrix} \begin{bmatrix} -D_{1e}(j\omega) - k_e D_{2e}(j\omega) \\ -D_{1o}(j\omega) - k_e D_{2o}(j\omega) \end{bmatrix} \tag{B.17}$$

Then k_θ and k_ω can be represent with ω as:

$$k_\theta = \frac{1}{|A_{\theta\omega}|}(D_{4o}(j\omega)\left(-D_{1e}(j\omega) - keD_{2e}(j\omega)\right) + D_{4e}(j\omega)\left(D_{1o}(j\omega) + k_e D_{2o}(j\omega)\right)) \tag{B.18}$$

$$k_\omega = \frac{1}{|A_{\theta\omega}|}(D_{3o}(j\omega)(D_{1e}(j\omega) + keD_{2e}(j\omega)) + D_{3e}(j\omega)(-D_{1o}(j\omega) - k_e D_{2o}(j\omega))) \tag{B.19}$$

From equations above, k_θ and k_θ can be plotted. An example is displayed when $k_e = 0.9$, at $V_0 = 15m/s$, the relevant C_f and c_r have been deployed from Fig. 3.7: as Figure B.1 illustrated. The whole $k_\theta - k_\omega$ plane is partitioned by boundaries from inequalities of characteristic polynomials terms and curve of complex root boundary. The whole plane is construct by 10 disjoint root invariant regions. By root checking,only gain set (k_θ, k_ω) from inner contour of curve, i.e. region 4 stabilized system.

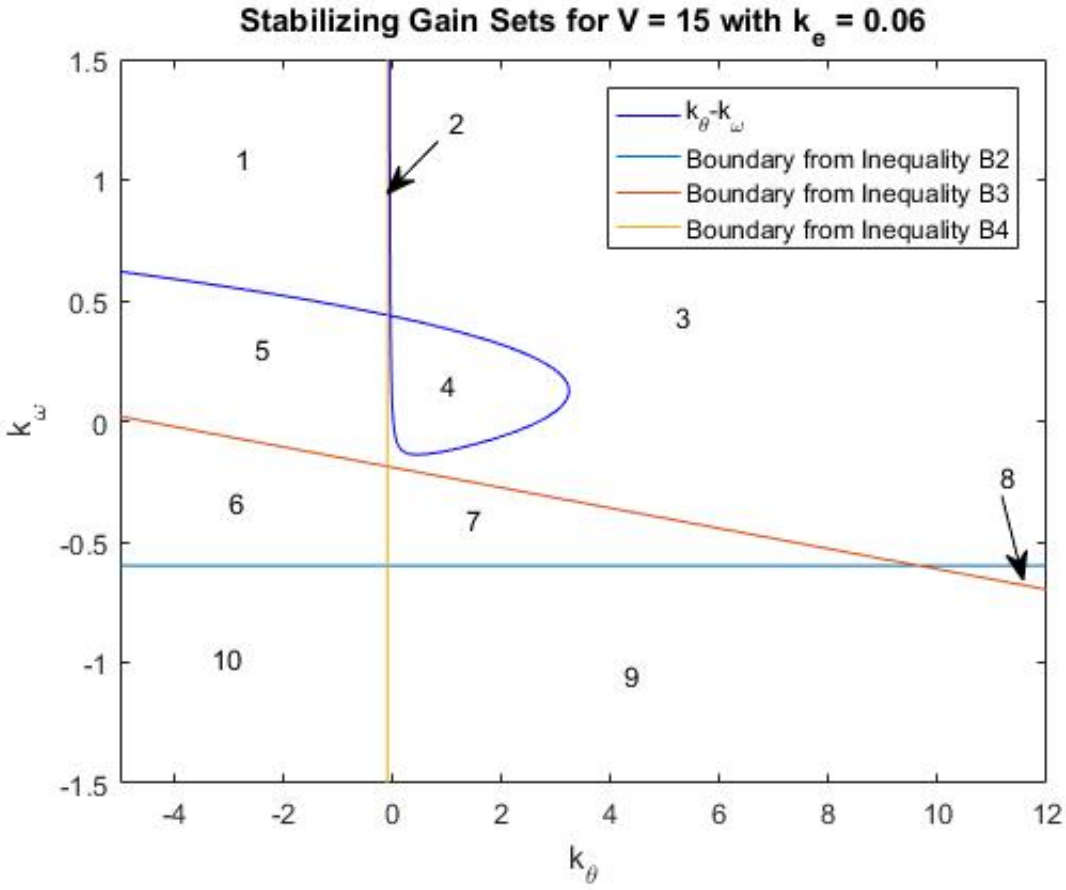

Figure B.1: Stabilizing Gain Set for $V = 15m/s$ with $k_e = 0.06$.