

Design of the Adaptive Cruise Control Systems: An Optimal Control Approach

by

Sanggyum Kim

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Masayoshi Tomizuka, Chair
Professor J. Karl Hedrick
Professor John Strain

Spring 2012

Design of the Adaptive Cruise Control Systems: An Optimal Control Approach

Copyright 2012
by
Sanggyum Kim

Abstract

Design of the Adaptive Cruise Control Systems: An Optimal Control Approach

by

Sanggyum Kim

Doctor of Philosophy in Mechanical Engineering

University of California, Berkeley

Professor Masayoshi Tomizuka, Chair

Modern automobiles are equipped with various driver assistance functions which enhance safety and relieve driver fatigue. With the recent development of sensor technology, the Adaptive Cruise Control (ACC) system has been put into practice. This thesis investigates several aspects for the ACC system including (1) smooth reaction of the host vehicle to the cutting in and out of lead vehicles, (2) real-time optimal profile generation for stop-and-go motions, (3) optimal feedback controller design, and (4) extension to Cooperative Adaptive Cruise Control (CACC) systems.

The ACC system should maintain an appropriate relative distance to the lead vehicle and should also maintain the desired speed set by the driver if there is no lead vehicle or if the speed of the lead vehicle is faster than the desired speed. Also, it should react smoothly when the lead vehicle cuts out or if a new lead vehicle cuts in from a side lane. This thesis introduces the virtual lead vehicle scheme to prevent the switching between the distance control and the speed control. By controlling the motion of the virtual lead vehicle to be smooth, the scheme could provide smooth reaction of the host vehicle to the cutting in and out of lead vehicles. Linear Quadratic (LQ) optimal control scheme is utilized to find the control gains for the virtual lead vehicle and the host vehicle. Variable weights are utilized in LQ for the virtual lead vehicle. With the variable weights, the motion of the virtual lead vehicle is controlled to be smooth when there is no safety threat while ensuring that the virtual lead vehicle is still responsive and fast when a dangerous situation occurs. ACC with Stop-and-Go and the Cooperative Adaptive Cruise Control (CACC) system are extensions of the conventional ACC system. Stop-and-Go system is targeted to be used in urban driving situation where the lead vehicle can stop completely. In that case, the Stop-and-Go system should have a capability to stop the host vehicle completely. The constant time-headway policy used to find the appropriate relative distance causes undesirable motion for a complete stop. In this thesis a sliding controller is utilized to control the complete stopping motion. To find the optimal stopping trajectory, a constrained Quadratic Programming (QP) problem is solved. A constrained QP is also used to find the optimal velocity profile when the stopped vehicle is to resume motion. Multi-resolution formulations and the Lemke algorithm are utilized to

find the optimal trajectories in real time. The CACC system utilizes wireless communication so that the vehicles in the network can share information with other vehicles. In this thesis, a centralized controller is designed by LQ optimal control scheme and potential benefits and problems are addressed. A Kalman filter with variable measurement noise covariance is introduced to compensate the lost data through the wireless network associated with the CACC system. The proposed control schemes have been verified through simulations.

To my family

Contents

Contents	ii
List of Figures	v
List of Tables	ix
1 Introduction	1
1.1 Adaptive Cruise Control	1
1.2 Performance Requirements	2
1.3 Control System Structure	3
1.4 Thesis Overview	5
2 Vehicle Model and Baseline Controller	7
2.1 Introduction	7
2.2 Vehicle Model	9
2.2.1 Longitudinal Vehicle Body Dynamics	9
2.2.2 Powertrain and Engine Model	11
2.2.3 Brake Model	13
2.3 Model Based Disturbance Compensator	15
2.3.1 Recursive Least Square Method	15
2.3.2 Disturbance Compensator	19
2.3.3 Simulation Results	22
2.4 Feedback Controller for Vehicle Following	23
2.4.1 Desired Distance	23
2.4.2 Tuning of Feedback Gains	24
2.4.3 Simulation Results	26
2.5 Summary	26
3 Virtual Lead Vehicle Scheme	29
3.1 Introduction	29
3.2 Concept of the Virtual Lead Vehicle Scheme	30
3.2.1 Mode Decision of the ACC system	31

3.2.2	Mode Switching Scheme	32
3.2.3	Virtual Lead Vehicle Scheme	32
3.3	Control of the Virtual Lead Vehicle in Transient	32
3.3.1	Double Integrator Model for the Virtual Lead Vehicle	34
3.3.2	Linear Quadratic Control	35
3.3.3	Comparison with the Model Predictive Control	36
3.4	Linear Quadratic Control With Variable Weights	36
3.4.1	Tuning of the LQ Weights	37
3.4.2	Simulation Results with Constant Weights	37
3.4.3	LQ with Variable Weights	38
3.4.4	Simulation Results with Variable Weights	39
3.5	Stability Analysis of the Linear Quadratic Controller with Variable Weights .	40
3.6	Disturbance Observer for the Virtual Lead Vehicle	46
3.6.1	Model of the Virtual Lead Vehicle	47
3.6.2	Disturbance Observer	47
3.6.3	Utilizing the Lead Vehicle Acceleration	55
3.6.4	Extended State DOB	57
3.7	Summary	59
4	Optimal Profile Generation for Stop-and-Go	61
4.1	Introduction	61
4.2	Optimal Relative Distance Profile during a Complete Stop	63
4.2.1	Optimization Problem Formulation	63
4.2.2	Velocity Profile of a Starting Motion	67
4.3	Sliding Control and Virtual Lead Vehicle Scheme	68
4.3.1	Sliding Controller Design for a Complete Stop	68
4.3.2	Simulation Results for a Complete stop	69
4.3.3	Virtual Lead Vehicle for a Starting Motion	71
4.4	Real Time Calculation of Quadratic Programming Problem	74
4.4.1	Algorithms to Solve the Constrained QP	76
4.4.2	Multi-resolution Optimization Problem Formulation	81
4.5	Summary	86
5	Cooperative Adaptive Cruise Control	87
5.1	Introduction	87
5.2	Linear Quadratic Optimal Controller for a Vehicle Platoon	88
5.2.1	Problem Description	88
5.2.2	Linear Quadratic Optimal Control	89
5.2.3	Simulation Result	90
5.3	Cut In/Out of a Vehicle in a Platoon	91
5.3.1	Problem Description	93
5.3.2	Virtual Lead Vehicle Scheme	94

5.3.3	Simulation Results	96
5.4	CACC in Lossy Network Condition	100
5.4.1	Problem Description	101
5.4.2	Control Schemes for Dropped Data	101
5.4.3	Simulation Results	102
5.4.4	Estimation with Kalman Filter	105
5.4.5	Simulation Result	108
5.5	Limitation of the Centralized Controller	112
5.6	Summary	112
6	Concluding Remarks and Open Issues	114
6.1	Concluding Remarks	114
6.1.1	Smooth Reaction to Cutting In and Out	114
6.1.2	Real-time Optimal Profile Generation for Stop-and-Go	114
6.1.3	Optimal Feedback Controller Design	115
6.1.4	Cooperative Adaptive Cruise Control	115
6.2	Open Issues	115
6.2.1	Lower Level Control and Brake/Engine Switching Logic	115
6.2.2	Mixed Traffic CACC	116
6.2.3	Implementation and Verification	116
	Bibliography	117

List of Figures

1.1	Overall ACC system structure	4
2.1	Free body diagram of the longitudinal vehicle model	9
2.2	Stopping a vehicle with a constant brake torque: Experimental results vs simulation results based on vehicle model with DDT tire model (a) longitudinal acceleration; (b) pitch rate [22]	10
2.3	Powertrain model block diagram	12
2.4	Braking System [13]	13
2.5	Simulation results with fuzzy logic: brake torque	14
2.6	Simulation result with $g_1 = g_2 = g_3 = 1$	20
2.7	Simulation result with $g_1 = g_2 = 0.1$, $g_3 = 1$	20
2.8	Simulation result where there is not enough excitations: with a vector-type forgetting factor without the constant trace method	21
2.9	Simulation result where there is not enough excitations: with a vector-type forgetting factor and a constant trace method	21
2.10	Simulation results with and without the disturbance compensator	22
2.11	Simulation results with the PID and PD controllers	27
2.12	Simulation results of a vehicle platoon with the PID controller	27
3.1	Mode switching logic	31
3.2	Block diagram of the mode switching scheme	33
3.3	Concept of the virtual lead vehicle scheme	33
3.4	Block diagram of the virtual lead vehicle scheme	34
3.5	Simulation results with MPC and LQ control: Position and velocity of the virtual lead vehicle	37
3.6	The symmetric root locus - Arrows indicate decreasing λ_a	38
3.7	Simulation results with constant weights position and velocity of the virtual lead vehicle	39
3.8	Simulation results when a lead vehicle cuts out position and velocity of the virtual lead vehicle	41
3.9	Simulation results when a lead vehicle cuts out variable weights	41

3.10 Simulation results when a lead vehicle cuts out position and velocity of the host vehicle	42
3.11 Simulation results when a lead vehicle cuts in position and velocity of the virtual lead vehicle	42
3.12 Simulation results when a lead vehicle cuts in variable weights	43
3.13 Simulation results when a lead vehicle cuts out position and velocity of the host vehicle	43
3.14 Phase plane plot of the closed loop system	44
3.15 Phase plane plot of the closed loop system at each quadrant comparing the worst case gains and the original gains	45
3.16 Phase plane plot of the closed loop system with the worst case gains	46
3.17 Block diagram of the speed and the position errors between the virtual and the actual lead vehicles	47
3.18 Block diagram of the DOB	48
3.19 Simulation results with and without the DOB when the lead vehicle accelerates: velocity	49
3.20 Simulation results with and without the DOB when the lead vehicle accelerates: position error	49
3.21 Simulation results with and without the DOB when the lead vehicle decelerates: velocity	50
3.22 Simulation results with and without the DOB when the lead vehicle decelerates: position error	50
3.23 Block diagram of the DOB	51
3.24 Simulation with the measurement noise	51
3.25 Simulation results with measurement noise using the velocity measurement: velocity	52
3.26 Simulation results with measurement noise using the velocity measurement: position error	53
3.27 Simulation results with measurement noise using the velocity measurement: acceleration of the virtual lead vehicle	53
3.28 Simulation results with a Q-filter: velocity	54
3.29 Simulation results with a Q-filter: acceleration of the virtual lead vehicle	54
3.30 Simulation results showing the vehicle behavior with and without the feedforward control when the lead vehicle decelerates: the lead and virtual vehicle's velocity	55
3.31 Simulation results showing the vehicle behavior with and without the feedforward control when the lead vehicle decelerates: the host vehicle's position	56
3.32 Simulation results showing the vehicle behavior with and without the feedforward control when the lead vehicle accelerates: the lead and virtual vehicle's velocity	56
3.33 Simulation results showing the vehicle behavior with and without the feedforward control when the lead vehicle accelerates: the host vehicle's position	57
4.1 Velocity of the lead and the host vehicle in case of a complete stop	62
4.2 Sketch of the complete stop	63

4.3	Solution of the QP comparing $l - 1$ and $l - 2$ norms, $\lambda_x = 1, \lambda_a = 1, \lambda_j = 0.5$	66
4.4	Solution of the QP comparing $\lambda_x = 1$ and $\lambda_x = 100, \lambda_a = 1, \lambda_j = 0.5$	67
4.5	Optimal profile generation result for starting motion	68
4.6	Simulation results with the sliding mode controller: relative distance	70
4.7	Simulation results with the sliding mode controller: acceleration and control command	70
4.8	Simulation results with the smooth sliding mode controller: relative distance	71
4.9	Simulation results with the smooth sliding mode controller: acceleration and control command	72
4.10	Simulation result for a starting motion with a sliding mode controller	73
4.11	Simulation result with a virtual lead vehicle utilizing the optimal velocity profile	74
4.12	Simulation result with a virtual lead vehicle and a slow lead vehicle: velocity	75
4.13	Simulation result with a virtual lead vehicle and a slow lead vehicle: distance error	75
4.14	Intermediate and final solutions of the Lemke algorithm	80
4.15	Intermediate and final solutions of the interior point algorithm	80
4.16	Sketch of the multi-resolution problem concept	82
4.17	Solutions of the optimization problems with uniform time step and multi-resolution time step	83
4.18	Concept of Multi-resolution formulation to calculate the optimal trajectory for the sliding control	84
4.19	Solutions of the optimization problems with uniform time step and multi-resolution time step	84
4.20	Simulation result with a sliding controller in case of a complete stop: velocity [m/sec]	85
4.21	Simulation result with a sliding controller in case of a complete stop: position [m]	85
5.1	Simulation results with LQ controllers: velocity (solid lines: with CACC, dashed lines: with ACC)	91
5.2	Simulation results with LQ controllers: error (solid lines: with CACC, dashed lines: with ACC)	92
5.3	Simulation results with LQ controllers: product of the error and the lead vehicle acceleration (solid lines: with CACC, dashed lines: with ACC)	92
5.4	Simulation results when a vehicle cuts out from the platoon	93
5.5	Sketch of the cutting out scenario	94
5.6	Simulation results when a vehicle cuts out from the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)	97
5.7	Simulation results when two vehicles cut out from the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)	97
5.8	Simulation results when two vehicles cut out from the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)	98
5.9	Sketch of the cutting in scenario	98

5.10 Simulation results without the virtual lead vehicle when a vehicle cuts into the platoon	99
5.11 Simulation results when a vehicle cuts into the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)	99
5.12 Simulation results when a vehicle cuts into the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)	100
5.13 Simulation results over lossy network: error for vehicle #1	102
5.14 Simulation results over lossy network: error for vehicle #2	103
5.15 Simulation results over lossy network: error for vehicle #3	103
5.16 Simulation results over lossy network: error for vehicle #4	104
5.17 Simulation results over lossy network: error for vehicle #5	104
5.18 Simulation results over lossy network with $\bar{\gamma} = 0.50$: error for vehicle #3	105
5.19 Simulation results with Kalman Filter over lossy network: error for vehicle #1 . .	109
5.20 Simulation results with Kalman Filter over lossy network: error for vehicle #2 . .	109
5.21 Simulation results with Kalman Filter over lossy network: error for vehicle #3 . .	110
5.22 Simulation results with Kalman Filter over lossy network: error for vehicle #4 . .	110
5.23 Simulation results with Kalman Filter over lossy network: error for vehicle #5 . .	111
5.24 Simulation results with Kalman Filter over lossy network with $\bar{\gamma} = 0.50$: error for vehicle #3	111

List of Tables

2.1	Typical gear ratios of an automatic transmission	12
3.1	Desired performance and weights	40
3.2	Simulation results comparing the DOB and the Extended State DOB(ESDOB) .	59
4.1	Calculation time [sec] of the Lemke algorithm / Interior point algorithm (Initial Condition : $d_0[m], v_0[m/s], a_0[m/s^2]$)	79

Acknowledgments

Writing the acknowledgements is the very last and one of the most difficult part for writing my dissertation. This dissertation may have not been done without the support and encouragement of many people. I would never be able to list all the people, however I want to express my warmest thanks to everyone.

I would like to thank my advisor, Professor Masayoshi Tomizuka, for providing me excellent research opportunities. Without his insightful advice, persistent inspiration, and considerate support, this dissertation would not exist. His inexhaustible academic passion and energy always impress and motivate myself. I would also want to express my thanks to Professor Miwako Tomizuka for her warm care to the entire lab and our families.

I want to acknowledge the superb instruction and support I have received from all of my professors in Berkeley. Special thanks to Professor Karl Hedrick, Professor John Strain, Professor Roberto Horowitz, and Professor Avideh Zakhor for their kind support as my dissertation and/or qualifying exam committee. Their kind advices and valuable comments really helped me improve the quality of my work.

To all my colleagues and friends in Mechanical Systems Control Laboratory, I appreciate all of your support, discussion, conversation, advice, and friendship over the past years: Wenjie, Pedro, Joonbum, Kyoungchul, Xu, Chi-shen, Raeche, Oak, Sandipan, Soo, Chun-Chih, Emma, Hoday, Ben, Ahmed, Wenlong, Cong, Haifei, Nancy, Kiyo, Nora, Omar, Chen-yu, Chung-yen, Junkai, Minghui, Hiroshi, and all others. Special thanks to Mike for his hard work as my English editor. Thank you Evan, Kay, and Kenta for your kind love to my family and being my neighbor. I also want to thank Dr. Shiang-Lung Ku, Dr. Takashi Nagata, Mr. Yasuyuki Matzuda, and Mr. Sumio Sugita for their support and advice on my research.

I would like to thank the Industrial Technology Research Institute in Taiwan for their kind financial support on my research project. Special thanks to Dr. Kuo-Hsiang Cheng for his advice and discussions about the project for past years. Big thanks to my Korean friends for sharing precious moments at Berkeley.

I really want to express my deepest love and respect to my brother Soonkyum. You are the best friend, mentor, and brother in my life. I would like to especially thank to my parents Youngki Kim and Kwangsook Baik for their endless love, support, and advice during my entire life. Also, I appreciate the kind support of my parents-in-law Yoonhyun Park and Samsoon Hong.

Finally, I want to say to my wife, Bo Ae Park that I love you and you are the best gift in my life. Without out your love and dedication, I could not do it. Also, I want to express my love to my children Jae Hee (Amy) and Jae Yoon (Kevin). You are the meaning of my life. Lastly, to my family, I love you all.

Chapter 1

Introduction

1.1 Adaptive Cruise Control

Since the very first modern automobile was invented by Karl Benz in 19th century, automobiles have became affordable, largely due to improvements in mass production technologies, and have been widely used all over the world. In 2009, more than 681 millions of cars were registered in the world [10]. With automobiles, longer and faster travels can be easily achieved and the quality of human life has been greatly enhanced.

Since the first automobile was invented many driver assistance features were introduced to enhance the safety and to reduce the drivers' fatigue. Many of these features are related to the automation of the driving. One of the most popular automation features is the cruise control system. The cruise control system takes over the throttle of the vehicle and maintains the speed of the vehicle at the driver's desired value. This feature is usually utilized in highway driving conditions where speeds remain constant for prolonged periods of time.

Various research and works have been done in both academia and industry to make driving be more automatic. The Automated Highway System (AHS) was one of the most popular works regarding the automation of vehicles. The California Partners for Advanced Transit and Highways (PATH) program has one of the successful research projects on the AHS. The project was focused on the highway driving of specially equipped vehicles in a separate lane from other manually driven vehicles. Various research on the longitudinal and lateral control of the vehicle body were performed in the project. The PATH AHS was successfully demonstrated in tests done in San Diego, California in 1997 [54].

With the recent development in sensor technology, the Adaptive Cruise Control(ACC) system has been introduced. The ACC system is an advanced version of the conventional cruise control system. While the conventional cruise control system maintains the desired speed set by the driver, the ACC system also maintains an appropriate relative distance to the lead vehicle. If the lead vehicle is detected to be driven at a speed lower than the desired speed, the system slows down the host vehicle and maintains an appropriate relative

distance, which usually depends on the states of the host vehicle and the environment.

The ACC system has extensions like stop-and-go and Cooperative Adaptive Cruise Control (CACC). Stop-and-go system is targeted to be used in urban driving conditions where the ACC cannot be easily utilized. The stop-and-go should have the capability to stop the host vehicle completely when there is a stopped lead vehicle [38]. Also in urban driving, the environments are more complex and the requirements for the sensor information are more strict [56]. Since it is expected to be utilizable at low speed in urban driving situations, the stop-and-go should be able to properly react to frequent cutting in and out of lead vehicles.

Cooperative Adaptive Cruise Control(CACC) is another enhanced version of the ACC. CACC utilizes communication between the vehicles and/or between the vehicle and the road structure. This communication allows the control system on a single vehicle to get information about other vehicles in the platoon such as the acceleration, the velocity, the throttle control command, and the brake control command. Various research have been done about the CACC in the area of traffic control and communication which show that the CACC has the potential to improve the traffic flow [3] and the string stability [33]. For a vehicle platoon, the string stability signifies whether spacing errors are increased upstream. A sudden deceleration or acceleration of the first vehicle of a platoon introduces errors for following vehicles. If the platoon is string unstable, the magnitudes of the errors are increased for later vehicles.

1.2 Performance Requirements

Stability

Stability is the most important feature for all control systems. Improper feedback controllers can make stable systems become unstable. In case of ACC systems, stability means that the speed of the host vehicle and the relative distance to the lead vehicle converge to their desired values. In addition to the stability of a single vehicle, string stability should be achieved for a platoon of vehicles. If not, it can cause a problem for the case of a platoon of vehicles equipped with identical ACC systems. If the feedback controller is not properly designed, when the very first vehicle of the platoon accelerates or decelerates, the distance errors increases along the vehicle stream. To guarantee string stability, the desired relative distance should be properly determined [42] or vehicle-to-vehicle communication is required [47].

Distance Control & Speed Control

ACC systems should be able to maintain an appropriate relative distance to the lead vehicle. However, if the lead vehicle is moving faster than the desired speed or no lead vehicle is detected by sensors, ACC systems should maintain the desired speed set by the driver like the conventional cruise control system. It is important that not only both distance controller and speed controller should operate properly and stably, but also the transient performance

should be smooth and safe when the control scheme is switched from one to the other. The transient motion is important for urban driving where there are frequent transitions. For example, when a lead vehicle cuts out, the relative distance to the new lead vehicle may be very large so that the speed control is activated without an appropriate transient controller. In that case, the host vehicle accelerates unnecessarily, which makes the driver uncomfortable and is not fuel efficient.

Passenger Comfort

Accelerations and decelerations are also important parameters in ACC systems, because they are closely related to the comfort of passengers. Accelerations and decelerations are unavoidable during the operation of the ACC system and excessive accelerations or decelerations make passengers feel uncomfortable and uneasy [51]. Therefore, an ACC system must perform so that acceleration and deceleration remain within a certain range. In addition to acceleration and deceleration, jerk, the time derivative of the acceleration, can be used as a parameter to measure the variation of acceleration and deceleration and should be maintained as small as possible to ensure the comfort of passengers. Also, the acceleration or deceleration of the host vehicle should not be the opposite from the drivers' expectation. For example, if the vehicle accelerates when the driver expect it to decelerate the driver would feel very uncomfortable with the ACC system.

Safety

While the comfort of passengers is important, it should not disrupt safety. In the case of an emergency stop, safety is the highest priority. So, the optimal profile of acceleration and deceleration should be determined according to the situation. For example, in the case of an emergency stop, the host vehicle should decelerate quickly to minimize the stopping distance whereas in non-emergency stopping situations, the host vehicle should decelerate slowly so that passengers do not feel discomfort.

1.3 Control System Structure

Figure 1.1 depicts the overall ACC system structure, which consists of the driver, the upper level controller, the lower level controllers, the host vehicle dynamics, and the environment. The driver decides whether the ACC system is turned on or off. The desired velocity and the desired time headway are determined by the preference of the driver. The desired velocity refers to the target velocity of the host vehicle when there is no vehicle or obstacle in front. The time headway represents the amount of time after which the lead vehicle and the host vehicle will collide when the lead vehicle suddenly stops and the host vehicle maintains its original velocity. The time headway determines the desired relative distance to the lead vehicle, and will be examined in Chapter 2.

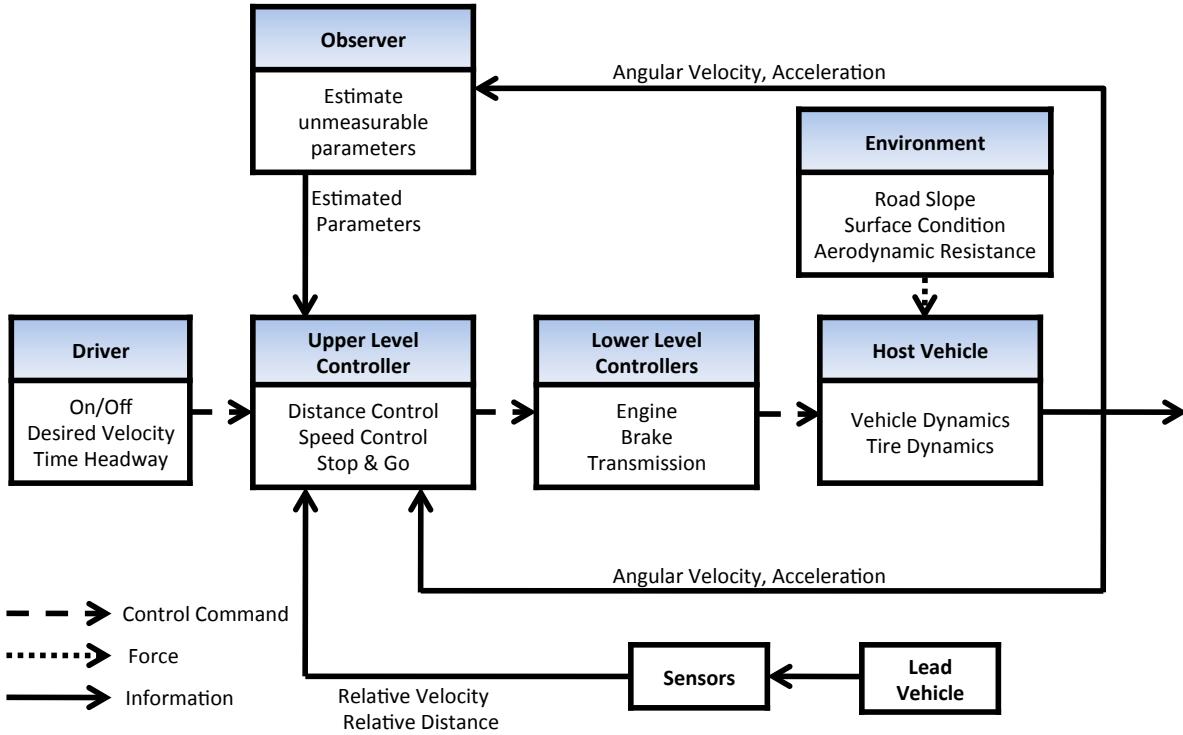


Figure 1.1: Overall ACC system structure

The control system is composed of an observer, an upper level controller, and lower level controllers. The observer directly utilizes measured feedback signals, such as angular velocities of each wheel and acceleration, to estimate unmeasurable parameters such as the vehicle mass, the aerodynamic coefficient, and the road slope. Information about the lead vehicle are also measured by sensors such as the radar and vision sensors. With measured feedback signals, all this information are utilized in the upper controller. The upper level controller analyses the information and decides the desired states of the host vehicle and produces the desired acceleration of the host vehicle accordingly. The desired acceleration commands from the upper level controller manage the lower level controllers such as the engine control unit, the transmission controller and the brake controller. However, the vehicle may not behave as predicted by the controller because there are uncertain environmental parameters such as the road slope, the rolling resistance, and the aerodynamic resistance as well as uncertainties in the vehicle dynamics. To overcome such uncertainties the performance of the vehicle is measured and provided to the control system as feedback signals.

1.4 Thesis Overview

To address the requirements of the ACC system, several control schemes are suggested and studied in this thesis. This thesis is organized as follows.

Chapter 2: Vehicle Model and Baseline Controller

The main purpose of this thesis is designing the upper level controller of the ACC system. However, other elements of the system should be reliable to effectively verify the performance of the designed upper level controller. In this thesis, the performance of the suggested controller is evaluated and verified via simulations. To do that, reliable and not too complicated models of vehicles are necessary.

In Chapter 2, the dynamics models used in simulations are introduced. A vehicle model including the longitudinal vehicle body dynamics, an engine model, a tire model, and a brake model are introduced. An observer which utilizes a recursive least square method is introduced to estimate the vehicle parameters and external disturbances. A model based disturbance compensator is used to compensate the external disturbances. As a basic upper level controller for distance control, a Linear Quadratic (LQ) optimal controller is studied to find optimal feedback control gains.

Chapter 3: Virtual Lead Vehicle Scheme

The main task of the ACC system is to maintain an appropriate relative distance to the lead vehicle. However when no lead vehicle is detected or the lead vehicle is moving faster than the desired speed, the host vehicle should maintain the desired speed set by the driver. The switching between the distance control and the speed control should be stable and smooth. To achieve that, the virtual lead vehicle scheme is introduced.

In Chapter 3, the modes of the ACC system and the concept of the virtual lead vehicle scheme are introduced. An LQ controller is designed to control the virtual lead vehicle. Variable weights are utilized for the LQ controller to better control the virtual lead vehicle during transient motion. The stability of the LQ controller with variable weights is analyzed. A Disturbance Observer (DOB) is utilized to estimate and compensate the acceleration of the lead vehicle.

Chapter 4: Optimal Profile Generation for Stop-and-Go

While the ACC is intended to assist the driver at relatively higher speeds, the Stop-and-Go is for low speeds and should have a capability to stop the vehicle completely if there is a stopped lead vehicle [38]. When the lead vehicle is stopped, the constant time-headway scheme, which determines the desired relative distance to the lead vehicle based on the speed of the host vehicle, may cause very slow stopping motion. To prevent it, a separate control scheme should be introduced when the lead vehicle stops completely.

In Chapter 4, optimization problems are formulated to find the optimal acceleration profile during a complete stop and a starting motion. Once the optimal profiles are achieved, a sliding mode controller is designed for a complete stop and the virtual lead vehicle scheme is utilized for the starting motion. Algorithms solving the optimization problem are studied and multi-resolution formulation is introduced to solve the problem in real time.

Chapter 5: Cooperative Adaptive Cruise Control

Cooperative Adaptive Cruise Control(CACC) is an enhanced version of the ACC. The main difference between CACC and ACC is that CACC utilizes communication between the vehicles and/or between the vehicle and the road structure. This communication allows the ACC system of a vehicle to get information about other vehicles in the platoon. Many previous research show that the CACC has the potential to improve both the traffic flow [3] and the string stability [33].

In Chapter 5, an LQ optimal control problem is solved for a vehicle platoon to find the feedback control gains. The virtual lead vehicle scheme is also utilized in CACC to control the platoon such that it will react smoothly when vehicles cut in or out from the platoon. Since wireless networks used in the CACC are usually lossy, packet loss during wireless communications is considered and various control scheme are compared by simulations.

Chapter 6: Concluding Remarks and Open Issues

In Chapter 6, concluding remarks of this thesis are summarized and some remaining issues are discussed as future works.

Chapter 2

Vehicle Model and Baseline Controller

2.1 Introduction

As described in Section 1.3, the ACC system consists of two control layers. The upper level controller determines the desired acceleration of the host vehicle while the lower level controllers determine the control commands to the engine and the brake based on the desired acceleration and current states of the engine and the brake. The main purpose of this thesis is to design the upper level controller. However, a reliable vehicle model and lower level controllers are necessary to evaluate and verify the performance of the upper level controller by simulations. In this chapter, a longitudinal vehicle model, an engine model, and a brake model are introduced and studied. These models are implemented in MATLAB/Simulink for simulations.

The longitudinal dynamic model of a vehicle body is based on the force equilibrium of the external forces which include the tire forces, the rolling resistances, the aerodynamic drag force, and the gravity force due to the road slope [42]. Among the external forces, the tire force can be controlled by the lower level controllers through the powertrain and the brake. Due to the compliance of the tire, the tire force and the input torque to the driving shaft have a dynamic relationship and it can be described by a tire model. In general, the tire force depends on the slip ratio and many tire models relating the slip ratio and the tire force have been studied. Very well known examples are the linear tire model and the magic formula tire model. Both models give the relationships between the longitudinal tire force and the slip ratio based on the empirical data, and describe the steady state behavior of the tire [60, 39]. However, for the Stop-and-Go feature which may introduce complete stops, a smooth motion control is important for the comfort of the driver and passengers. To predict the oscillation during a complete stop, a dynamic deflection tire model [22] is used in the simulation for this thesis. The engine and the brake models should be able to reproduce the behaviors of the actual engine and the brake, however do not need to be very complicated. In this thesis, an engine model and a lower level controller in [30, 31] are used. A brake model studied in [13] are used for simulations in this thesis.

Other external forces except the tire force are from environment. Considering the vehicle body as a double integrator whose input is the acceleration and the outputs are the longitudinal velocity and the longitudinal position, these external forces are disturbances which should be effectively compensated. Various research have been done to estimated the vehicle parameters and the environmental disturbances including the vehicle mass, the tire-road friction, and the road slope. Some research have been performed to estimate the vehicle mass and the road slope via the Extended Kalman filtering [27], the Extended Kalman filtering with Model Predictive Controller [59], and the Recursive Least Square method [57]. In the previous research, the vehicle mass and the road slope are simultaneously estimated. It is assumed, however, that the external disturbances such as the aerodynamic resistance and the rolling resistance are known. Also, the torque generated by the engine or the brake is assumed to be known or able to be estimated from a model. In this thesis it is assumed that the Multi-Sensing-Hub (MSH) [52] is used to directly measure the tire force. With the force measurement, a recursive least square method with a vector-type forgetting factor with a constant trace is used to estimate not only the vehicle mass and the road slope but also the aerodynamic coefficient. The estimated vehicle mass is utilized to calculate the desired torque from the desired acceleration. The gravity force due to the road slope and the aerodynamics resistance are compensated via a model based disturbance compensator.

Assuming perfect disturbance compensation and lower level controllers, the longitudinal vehicle dynamics can be modeled as a double integrator whose input is the acceleration. The desired acceleration is determined by an upper level feedback controller. To design a feedback controller, stability is one of the most important thing to consider. In case of ACC, there is a chance that the vehicles equipped with the same ACC system make a platoon and string stability should also be considered. To prevent the string instability, a constant time-headway method is used to calculate the desired relative distance to the lead vehicle [42]. PID (Proportional-Integral-Differential) controller is one of the most popular of feedback controllers and various methods have been studied to tune the feedback gains [50, 17, 16]. In this chapter, the Linear Quadratic (LQ) optimal control theory is used to find an optimal state feedback control gains when the host vehicle follows a lead vehicle with a constant time-headway. The state feedback controller is equivalent to a PD (Proportional-Differential) controller for second order systems and the PID gains can be found by the Linear Quadratic Controller with Integrator(LQI).

This chapter is organized as follows. A vehicle model including the longitudinal vehicle body dynamics, an engine model, a tire model, and a brake model is introduced in Section 2.2. In Section 2.3, a recursive least square method is introduced to estimate the vehicle parameters and external disturbances. A model based disturbance compensator is used to compensate the external disturbances. In Section 2.4, an LQ optimal controller is studied to find feedback control gains. Section 2.5 summarizes the chapter.

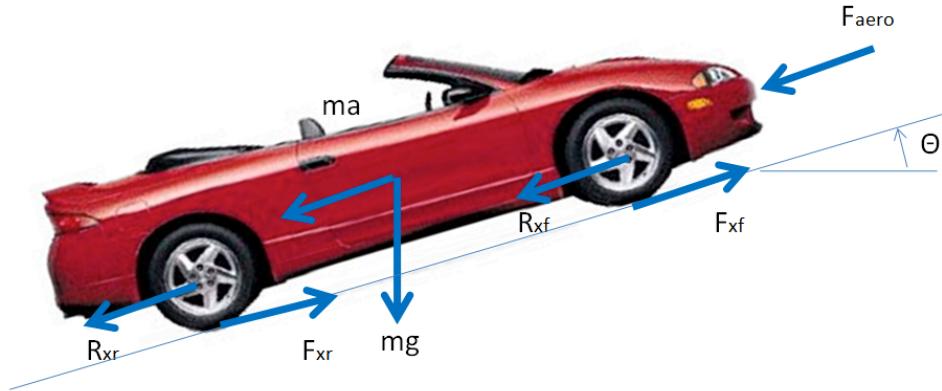


Figure 2.1: Free body diagram of the longitudinal vehicle model

2.2 Vehicle Model

2.2.1 Longitudinal Vehicle Body Dynamics

This section presents a longitudinal dynamics model of the vehicle body. Referring to Figure 2.1, the governing equation of the longitudinal vehicle model is given by:

$$m\ddot{x} = F_{xf} + F_{xr} - F_{aero} - R_{xf} - R_{xr} - mg \sin \theta \quad (2.1)$$

where m is the mass of the vehicle, x is the longitudinal position of the vehicle, F_{xf} and F_{xr} are the tire forces at front and rear tires, F_{aero} is the aerodynamic resistance, R_{xf} and R_{xr} are the rolling resistances at front and rear tires, and θ is the slope angle of the road.

Tire Force

Tire forces, F_{xf} and F_{xr} , are controlled by the engine and the brake torque, however are generated via road-tire interaction. For this reason, the tire model is extremely important in vehicle control. There are many tire models including many empirical models [39]. While they have been successfully integrated with vehicle dynamics in many applications, they are static models and are not suited for transient phenomena in particular at low speeds. The Relaxation Length Tire (RLT) model was introduced to address this problem [29]. The Dynamic Deflection Tire (DDT) model has been developed by S.L. Koo, *et al* for better prediction and description of the tire characteristics, especially at low speeds [22]. In case of Stop-and-Go, the longitudinal oscillation during deceleration and stopping with the brake, which is related to the comfort for passengers, should be considered. Figure 2.2 which is adopted from [22] shows that the longitudinal oscillation can be predicted by using the DDT model.

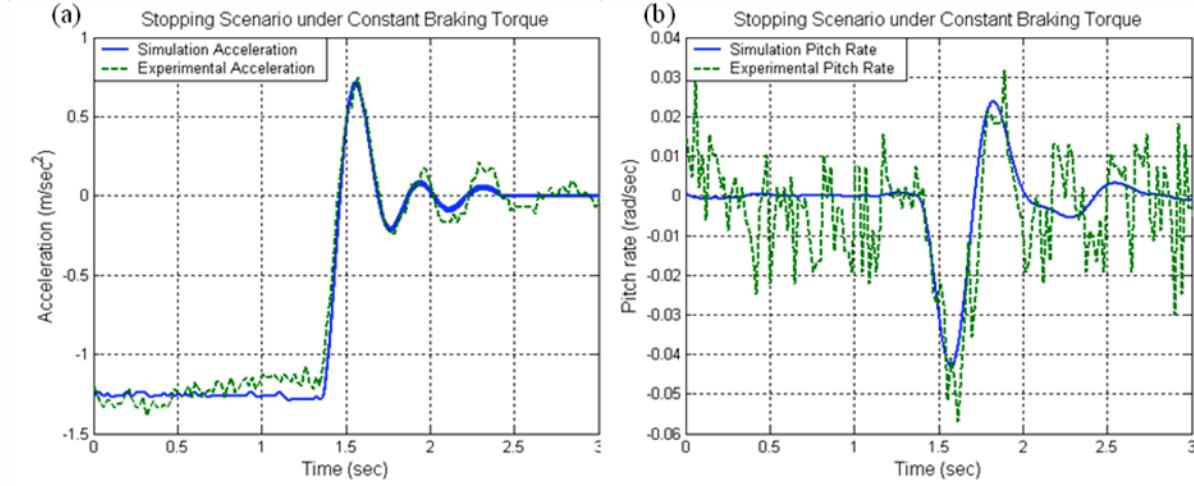


Figure 2.2: Stopping a vehicle with a constant brake torque: Experimental results vs simulation results based on vehicle model with DDT tire model (a) longitudinal acceleration; (b) pitch rate [22]

The front and rear tire forces can be derived from the DDT model. The governing equation is given by:

$$F_{long} = D_{long}\dot{\sigma}_x + C_{long}\sigma_x \quad (2.2)$$

where D_{long} and C_{long} are the damping and the spring constant of the tire, respectively. σ_x represents the longitudinal tire deflection, which is defined by:

$$\sigma_x = x_{cp} - x_w \quad (2.3)$$

where x_{cp} is the longitudinal displacement of the contact patch and x_w is the longitudinal displacement of the wheel. x_w is assumed to be the same with the displacement of the vehicle.

The kinematic equation of x_{cp} can be expressed as:

$$\dot{x}_{cp} = R_0(1 + \lambda)\omega \quad (2.4)$$

where R_0 is the effective radius of the tire and ω is the tire angular velocity. The longitudinal slip λ is defined by:

$$\lambda = \frac{\sigma_x}{\sigma_{long}} \quad (2.5)$$

where σ_{long} is the tire relaxation length.

Torque-Tire Force Relation

The governing equation for the torque and tire force is given by:

$$I_w\dot{\omega} = \tau - R_0F_{long} \quad (2.6)$$

where I_w is the inertia of the wheel.

Aerodynamic Resistance

The aerodynamic resistance on a vehicle can be represented by:

$$F_{aero} = \frac{1}{2} \rho C_d A_f |v_x + v_{wind}| (v_x + v_{wind}) \quad (2.7)$$

where ρ is the mass density of air, C_d is the aerodynamic draw coefficient, A_f is the front projected area of the vehicle, v_x is the longitudinal velocity of the vehicle, and v_{wind} is the wind velocity.

Rolling Resistance

The rolling resistance of each tire is proportional to the normal force of the each tire: i.e.

$$R_x = f F_z \quad (2.8)$$

where f is the rolling resistance coefficient, and F_z is the normal force. The normal forces for front and rear tire are respectively:

$$F_{zf} = \frac{-F_{aero} h_{aero} - m \ddot{x}h - mgh \sin \theta + mgl_r \cos \theta}{l_f + l_r} \quad (2.9)$$

$$F_{zr} = \frac{F_{aero} h_{aero} + m \ddot{x}h + mgh \sin \theta + mgl_f \cos \theta}{l_f + l_r} \quad (2.10)$$

where h_{aero} is the height where the aerodynamic resistance is applied, h is the height of the center of gravity of the vehicle, and l_f and l_r are the lengths from the center of gravity of the vehicle to the front or the rear tire axle, respectively.

2.2.2 Powertrain and Engine Model

Previously, many engine models have been developed and studied. In view that we are primarily interested in the motion of the vehicle body, however, too detailed engine model is not required. In this thesis, a discrete event model [30, 31] is utilized in simulations.

Engine Model Structure

Figure 2.3 shows the interaction inside the powertrain model. The input to the entire powertrain model is the desired engine torque, and the output is the torque delivered to wheels. The first block of the powertrain is the engine torque controller. In this thesis, a simple model based controller which calculates the appropriate throttle angle for the desired engine torque based on the current engine state is used. The throttle angle and angular

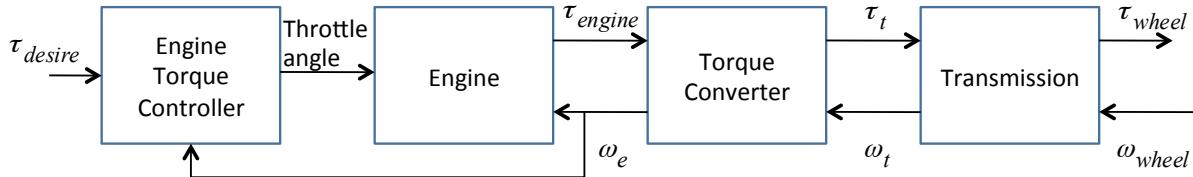


Figure 2.3: Powertrain model block diagram

Gear	Ratio	Remark
1st	2.8:1	
2nd	1.5:1	
3rd	1:1	
4th	0.7:1	Final gear with a ratio of 4:1 is additionally attached after each gear.

Table 2.1: Typical gear ratios of an automatic transmission

velocity of the engine determine the output torque of the engine block [30, 31]. Then, the engine torque goes through the torque converter and transmission to the wheels.

The torque generation process of a combustion engine consists of four different strokes: intake, compression, combustion, and exhaust strokes. The dynamics of each stroke are different from the others. Each stroke is changed to the next stroke after the engine rotates 180 degrees. Since, the engine rotation speed is not constant, the interval between the strokes is not fixed and the dynamics of the engine is hard to describe in a time domain. So the discrete event model describes the operation of the engine as events occur.

Transmission

Since the torque production of a vehicle engine is limited in magnitude, the engine rotates very fast and the transmission gears reduces the speed while amplifying the torque. The transmission is regarded as a gear box which has 3 to 6 different gears inside it. At steady states, the transmission works like a gear with a fixed ratio. During transients, according to the engine speed and the throttle angle, gear is shifted to maintain the engine RPM in an adequate range. Typical gear ratios of an automatic transmission are given in Table 2.1.

At steady states, the torque transmitted to the wheel is:

$$\tau_{wheel} = \frac{1}{R} \tau_t \quad (2.11)$$

where R is the gear ratio including the final gear. i.e. at 1st gear, $R = 2.8 \times 4$

The speed of transmission is:

$$\omega_t = \frac{1}{R} \omega_{wheel} \quad (2.12)$$

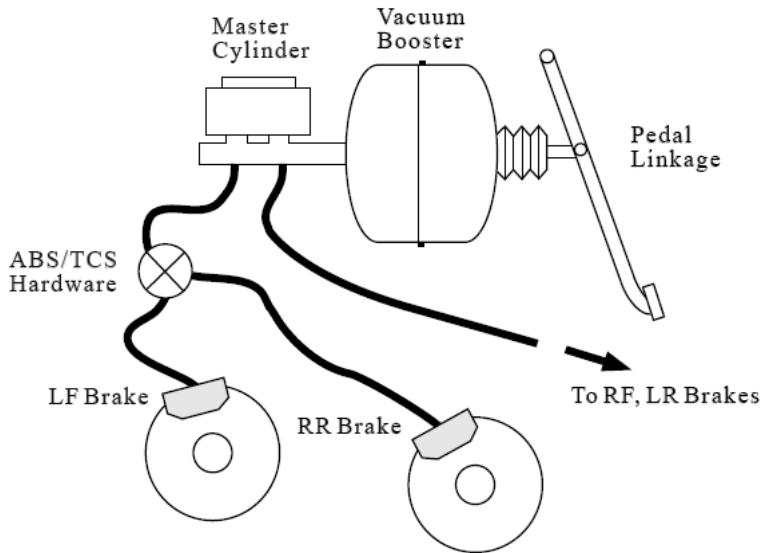


Figure 2.4: Braking System [13]

As shown in (2.11) and (2.12), at a lower gear, engine RPM is higher than a higher gear. However, the transmitted torque at the wheel may be larger at a lower gear.

The dynamics during a gear change is studied in [9], which is complex. An alternative way to express the gear change dynamics [42], which is simpler but adequate for the present study is:

$$T\dot{\tau}_{wheel} + \tau_{wheel} = \frac{1}{R}\tau_t \quad (2.13)$$

$$T\dot{\omega}_t + \omega_t = \frac{1}{R}\omega_{wheel} \quad (2.14)$$

where T is a time constant. The initial value of τ_{wheel} is zero when the gear change is initiated. For ω_t , the initial value is $\frac{1}{R_{old}}\omega_{wheel}$ where R_{old} is the previous gear ratio.

2.2.3 Brake Model

Unlike the conventional cruise control system, the ACC system with Stop-and-Go should utilize the brake to make the host vehicle decelerate. To design and evaluate the brake control system, a brake model which can better represent the characteristics of the actual brake system is required for simulations. In this thesis, the brake model introduced in [13] is used. However, unlike [13], it will be assumed that there is no additional pressure cylinder in the brake system and the brake torque is determined only by the brake pedal force.

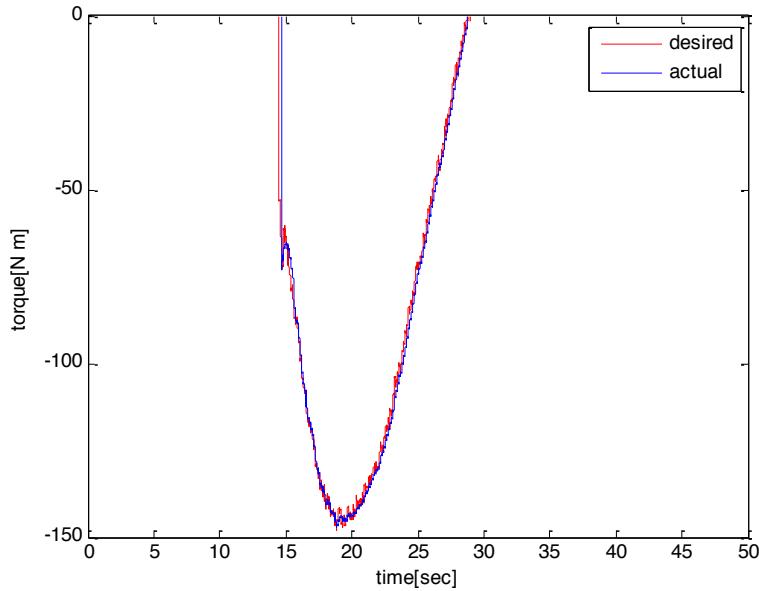


Figure 2.5: Simulation results with fuzzy logic: brake torque

Brake System

Figure 2.4 which is adopted from [13] depicts the structure of an automotive brake system. When the driver wants to decelerate, the driver presses the brake pedal. The force from the driver is amplified by the vacuum booster. The amplified force increases the pressure of the master cylinder which is connected to the brakes of four wheels. Between the master cylinder and the wheel brakes there are Anti-lock Brake System / Traction Control System actuators. In this thesis, the piston masses are ignored and single state hydraulic system model is used.

Brake Controller

In [13], an additional hydraulic actuator was introduced and a sliding controller was designed to control the pressure of the brake pad cylinder and the brake torque. In this thesis, it is assumed that no additional actuator is available. Instead, a Disturbance Observer (DOB) based feedback controller is utilized. Also, to prevent the effect of the vacuum booster hysteresis, a fuzzy switching logic is used. The lower level controller shows a reasonably good performance as shown in Figure 2.5. The brake model is implemented in MATLAB with the parameters adopted from [13].

The simulation results show that the performance of the DOB is acceptable. Also it is not necessary to modify the brake hydraulic system. However, an additional actuator in the brake pedal is required. One main drawback of the DOB is that, due to the nonlinearity of

the brake system, the Q-filter should be designed very conservatively and the performance is worsened with more conservative choices of the cutoff frequency. The Q-filter should be tuned later with experimental data. Also, the critical force to compensate the spring forces inside the vacuum booster is necessary for the fuzzy switching logic and should be also tuned by experiments.

2.3 Model Based Disturbance Compensator

Among the environmental disturbances, the aerodynamic resistance is dominant at high speeds, while the gravity force due to the road slope can be the primary disturbance at low speeds. The disturbances are hardly measurable and is related to the vehicle parameters such as the vehicle mass. Thus, we have conducted a simulation study to estimate the vehicle mass, the aero dynamic coefficient and the road slope by a least square method. The estimated values are used in a model based disturbance compensator to compensate the external disturbances. Simulation studies are performed to show the performance of the disturbance compensator.

2.3.1 Recursive Least Square Method

Model Description

The longitudinal vehicle dynamics of a front wheel driven vehicle may be expressed as:

$$m\ddot{x} = F_{tire} - mg \sin \theta - C_{aero}|\dot{x}|\dot{x} \quad (2.15)$$

where m is the mass of the vehicle, F_{tire} is the force from the tire to the vehicle body including the rolling resistance, g is the gravity acceleration, θ is the road slope, C_{aero} is the aerodynamic coefficient, and x is the longitudinal position of the vehicle. Among the variables, F_{tire} can be measured by a Multi Sensing Hub (MSH) [52], \dot{x} can be obtained from the rear (undriven) wheel angular velocity which can be measured by an encoder, and \ddot{x} can be measured by an accelerometer or estimated from \dot{x} . With these measurements and/or estimates, we would like to estimate or identify parameters M , C_{aero} , and θ .

(2.15) is a model suited for parameter identification, in particular when F_{tire} can be directly measured by MSH. If it is not possible to use the MSH and measure the tire force directly, we will need to measure or know the engine torque and the brake torque as well as other quantities such as the vehicle speed.

Recursive Least Square Problem

In the least square method, the system equation is represented as:

$$y(k) = a_1x_1(k) + a_2x_2(k) + a_3x_3(k) = \theta^T \varphi(k) \quad (2.16)$$

where $\theta = [a_1 \ a_2 \ a_3]^T$, $\varphi(k) = [x_1(k) \ x_2(k) \ x_3(k)]^T$, and k denotes time. From (2.15), the variables are defined as:

$$y(k) = a(k) \quad (2.17)$$

$$\theta = \left[\frac{1}{m} \ -\frac{R_0^2}{m} C_{aero} \ -\sin\theta \right]^T \quad (2.18)$$

$$\varphi(k) = [F_{tire}(k) \ \omega^2(k) \ g] \quad (2.19)$$

where R_0 is the effective radius of the tire and ω is the angular velocity of the tire. At time k , we would like to find the best estimate of θ , $\hat{\theta}(k)$, that minimizes the cost function

$$J = \sum_{i=1}^k [y(i) - \hat{\theta}^T(k)\varphi(i-1)]^2 \quad (2.20)$$

The necessary condition for minimum is:

$$\frac{\partial J}{\partial \hat{\theta}} = -2 \sum_{i=1}^k [y(i) - \hat{\theta}^T(k)\varphi(i-1)] \varphi(i-1) = 0 \quad (2.21)$$

Solving this equation for $\hat{\theta}(k)$, we obtain

$$\hat{\theta}(k) = F(k) \sum_{i=1}^k \varphi(i-1)y(i) \quad (2.22)$$

where, $F(k)$ is the gain matrix, given by

$$F(k) = \left[\sum_{i=1}^k \varphi(i-1)\varphi^T(i-1) \right]^{-1} \quad (2.23)$$

(2.22) is the batch formula for the least square estimate. For real time applications, the recursive formula given in the following form is better suited.

$$\hat{\theta}(k+1) = \hat{\theta}(k) + [\text{correction term}] \quad (2.24)$$

To obtain the recursive least square formula, we first note

$$F^{-1}(k+1) = \sum_{i=1}^{k+1} \varphi(i-1)\varphi^T(i-1) = F^{-1}(k) + \varphi(k)\varphi^T(k) \quad (2.25)$$

(2.25) is in a recursive form. We then note

$$\hat{\theta}(k+1) = F(k+1) \sum_{i=1}^{k+1} \varphi(i-1)y(i) \quad (2.26)$$

$$= F(k+1) \left[\sum_{i=1}^k \varphi(i-1)y(i) + \varphi(k)y(k+1) \right] \quad (2.27)$$

$$= F(k+1)[F^{-1}(k)\hat{\theta}(k) + \varphi(k)y(k+1)] \quad (2.28)$$

$$= F(k+1)[(F^{-1}(k+1) - \varphi(k)\varphi^T(k))\hat{\theta}(k) + \varphi(k)y(k+1)] \quad (2.29)$$

$$(2.30)$$

Thus we obtain

$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k+1)\varphi(k)[y(k+1) - \hat{\theta}^T(k)\varphi(k)] \quad (2.31)$$

By applying matrix inversion lemma to Eq. (2.25),

$$F(k+1) = F(k) - \frac{F(k)\varphi(k)\varphi^T(k)F(k)}{1 + \varphi^T(k)F(k)\varphi(k)} \quad (2.32)$$

(2.31) and (2.32) give the recursive least square formula. With the least square method, we can still get the same result. However, as the number of data is increased, the size of the data matrix is getting larger and the calculation time will be significantly increased. In that case, the real time application is almost impossible. With the recursive least square method [24], the very new data are used to update the estimates and the calculation time remains the same for each iteration.

For time varying parameters, we can introduce a forgetting factor, λ , in the cost function to reduce the effect of older data on the current estimate of the parameter vector:

$$J = \sum_{i=1}^k \lambda^{k-i} \left[y(i) - \hat{\theta}^T(k)\varphi(k-1) \right]^2 \quad (2.33)$$

where $0 < \lambda \leq 1$. In this case, the adaptation gain is updated by:

$$F(k+1) = \frac{1}{\lambda} \left[F(k) - \frac{F(k)\varphi(k)\varphi^T(k)F(k)}{\lambda + \varphi^T(k)F(k)\varphi(k)} \right] \quad (2.34)$$

Constant Trace Method

Recursive least square method with a forgetting factor is effective when the dynamic parameters are not fixed but time varying, e.g. the road slope. However, with the forgetting factor noises can be dominant and the parameter estimates may drift in case of signals with small

amplitude. To overcome this problem, a constant trace method is introduced. Without a forgetting factor, the adaptation gain matrix is updated by:

$$F^{-1}(k+1) = F^{-1}(k) + \varphi(k)\varphi^T(k) \quad (2.35)$$

Its trace is given by:

$$\text{tr}[F^{-1}(k+1)] = \text{tr}[F^{-1}(k)] + |\varphi(k)|^2 \quad (2.36)$$

With a non-zero $\varphi(k)$, the trace of $F^{-1}(k)$ always increases and consequently the trace of $F(k)$ decreases. Since the update law of the estimated parameters is

$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k+1)\varphi(k)[y(k+1) - \hat{\theta}^T(k)\varphi(k)] \quad (2.37)$$

$\hat{\theta}(k)$ is updated lesser in magnitude with a smaller $F(k)$. This is the reason why the forgetting factor is introduced. With a forgetting factor, the gain matrix is updated by:

$$F(k+1) = \frac{1}{\lambda} \left[F(k) - \frac{F(k)\varphi(k)\varphi^T(k)F(k)}{\lambda + \varphi^T(k)F(k)\varphi(k)} \right] \quad (2.38)$$

Its trace is given by:

$$\text{tr}[F(k+1)] = \frac{1}{\lambda} \text{tr}[F(k)] - \frac{F(k)\varphi(k)\varphi^T(k)F(k)}{\lambda^2 + \lambda\varphi^T(k)F(k)\varphi(k)} \quad (2.39)$$

The forgetting factor prevents the gain matrix from decreasing to zero. The gain matrix, however, may blow up if $\varphi(k)$ is not persistently exciting. A constant trace method maintains the trace of the gain matrix at a constant to prevent it from either blowing up or decreasing to zero. The update law is given by:

$$F(k+1) = \frac{1}{\lambda_1(k)} \left[F(k) - \frac{F(k)\varphi(k)\varphi^T(k)F(k)}{\lambda_1(k)/\lambda_2(k) + \varphi^T(k)F(k)\varphi(k)} \right] \quad (2.40)$$

where $\lambda_1(k)$ is updated by:

$$\lambda_1(k) = \frac{1}{\text{tr}[F(0)]} \text{tr}[F(k) - \frac{F(k)\varphi(k)\varphi^T(k)F(k)}{\alpha + \varphi^T(k)F(k)\varphi(k)}] \quad (2.41)$$

where $\alpha = \lambda_1/\lambda_2$ is a constant value. (2.40) and (2.41) guarantee that the trace of the gain matrix is maintained constant and does not blow up nor decrease to zero.

Vector-type Forgetting Factor

With the constant trace method, we can get a more stable estimation result. However, if we apply a same forgetting factor for all parameters, there is an undesirable excitation in the estimates of constant parameters like the vehicle mass when only the road slope is changing. If the estimator has a prior information that the vehicle mass and the aerodynamic coefficient

are time invariant, this problem may be prevented. A vector-type forgetting factor can make this happen [65].

The vector-type forgetting factor method applies different forgetting factors for different parameters. For frequently varying parameters, smaller forgetting factors will be applied to update the parameters more sensitively. In the vector-type forgetting factor method, the gain matrix is updated by:

$$F(k+1) = \Lambda(k) \left[F(k) - \frac{F(k)\varphi(k)\varphi^T(k)F(k)}{1 + \varphi^T(k)F(k)\varphi(k)} \right] \Lambda(k) \quad (2.42)$$

where the forgetting factor matrix $\Lambda(k)$ is given by:

$$\Lambda(k) = \begin{bmatrix} \frac{1}{\lambda_1(k)} & 0 & 0 \\ 0 & \frac{1}{\lambda_2(k)} & 0 \\ 0 & 0 & \frac{1}{\lambda_3(k)} \end{bmatrix} \quad (2.43)$$

Each forgetting factor $\lambda_i(k)$'s are updated by:

$$\lambda_i(k) = \sqrt{\frac{1}{1 + \frac{g_i \varepsilon^o(k+1)^2}{1 + \varphi^T(k)F(k)\varphi(k)}}} \quad (2.44)$$

where $\varepsilon^o(k)$ is an a-priori estimation error, and g_i 's are constants for each parameters. Larger g_i 's are used for frequently varying parameters to make the forgetting factor smaller.

However, the vector-type forgetting factor method is too sensitive to the values of g_i 's. Also, the estimation errors are too large if a sufficient amount of excitation is not given. To resolve these problems, the constant trace method can be introduced again. While using a vector-type forgetting factor, the trace of the gain matrix is maintained constant. With this constant trace method with a vector-type forgetting factor, the estimator works better. Figures 2.6 and 2.7 show the simulation results with different values of g_i 's, and the results are almost the same. Also in Figures 2.8 and 2.9, it is seen that even without enough excitations, the estimator works well.

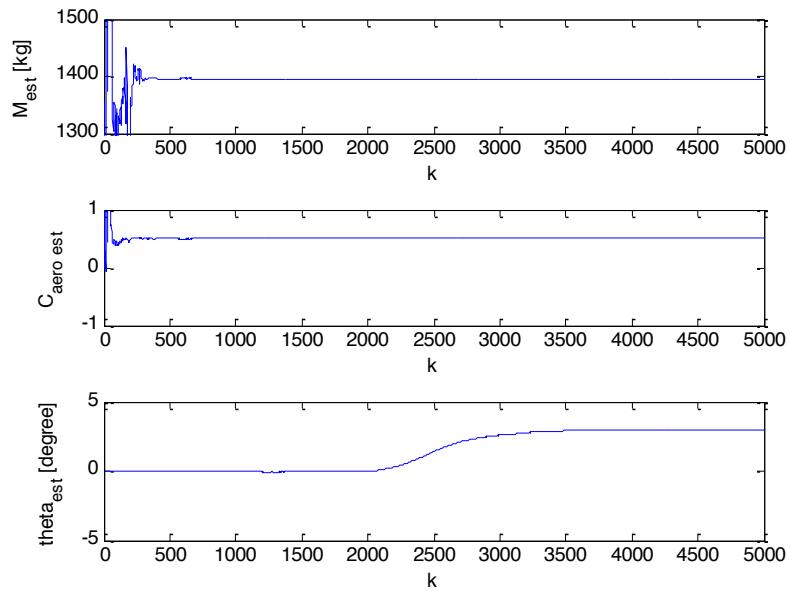
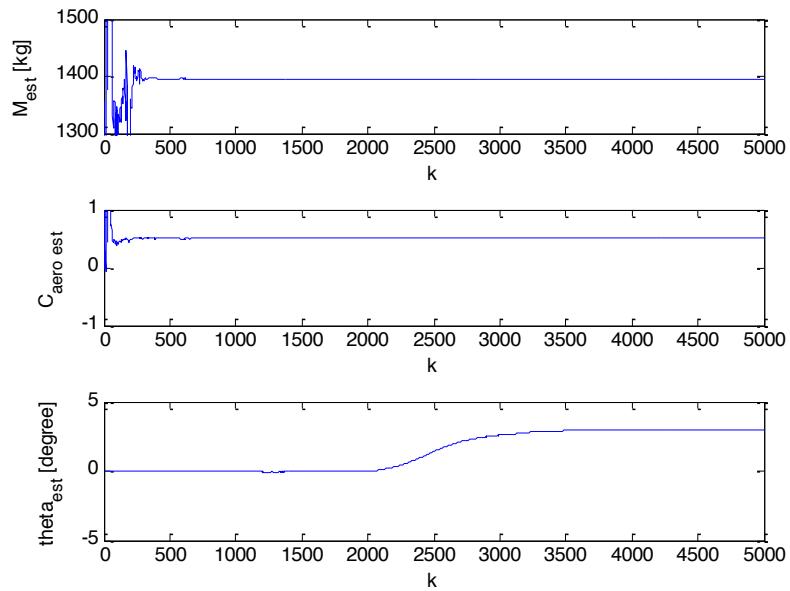
2.3.2 Disturbance Compensator

With the estimated values, it is possible to estimate the disturbances based on the previously introduced model used in the parameter estimation. The three disturbance models are given by:

$$F_{aero} = \frac{1}{2}\rho C_d A_f |v_x + v_{wind}|(v_x + v_{wind}) \quad (2.45)$$

$$F_{grav} = mg \sin \theta \quad (2.46)$$

$$F_{roll} = fF_z = fmg \quad (2.47)$$

Figure 2.6: Simulation result with $g_1 = g_2 = g_3 = 1$ Figure 2.7: Simulation result with $g_1 = g_2 = 0.1, g_3 = 1$

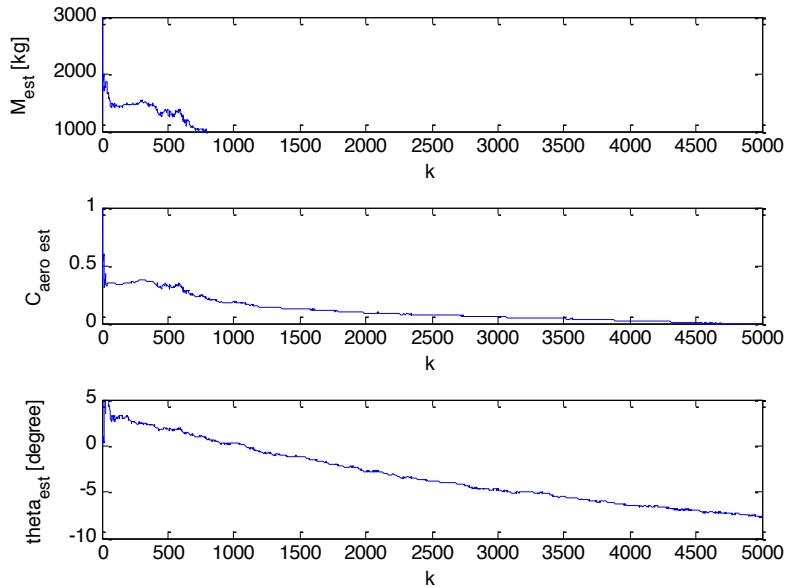


Figure 2.8: Simulation result where there is not enough excitations: with a vector-type forgetting factor without the constant trace method

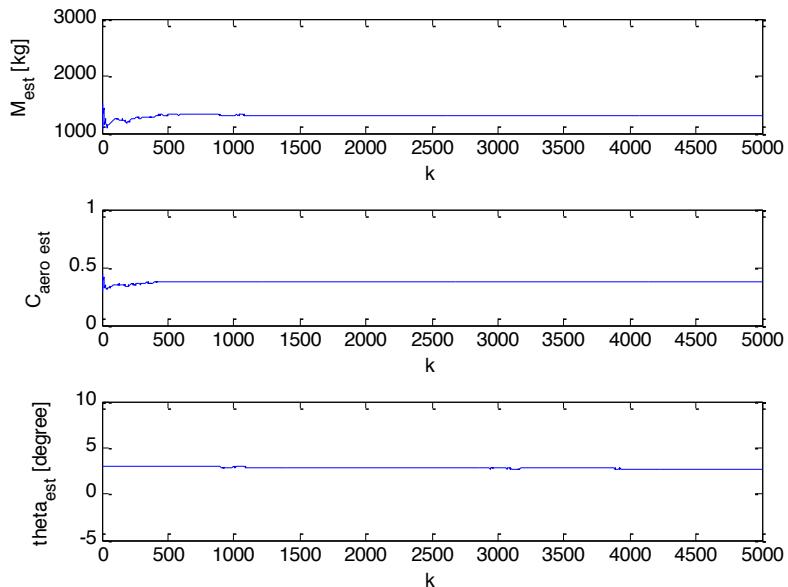


Figure 2.9: Simulation result where there is not enough excitations: with a vector-type forgetting factor and a constant trace method

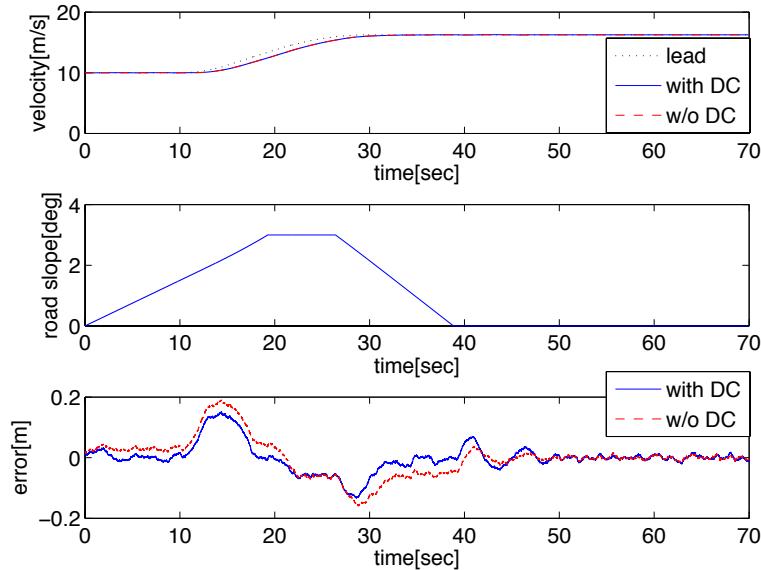


Figure 2.10: Simulation results with and without the disturbance compensator

where F_{aero} is the aerodynamic resistance, F_{grav} is the gravitational resistance, and F_{roll} is the rolling resistance. $\frac{1}{2}\rho C_d A_f$ is the aerodynamic coefficient, m is the vehicle mass, and θ is the road slope which are estimated by the recursive least square method. v_x is the longitudinal velocity of the vehicle and v_{wind} is the longitudinal wind velocity which is unknown and is ignored. f is the rolling resistance coefficient and assumed to be known.

The output from the disturbance compensator is the desired additional acceleration and is given by:

$$a_{dc} = \frac{1}{\hat{m}}(\hat{F}_{aero} + \hat{F}_{grav} + \hat{F}_{roll}) \quad (2.48)$$

This additional acceleration command to compensate for disturbances is added to the feedback command:

$$a_{desired} = a_{feedback} + a_{dc} \quad (2.49)$$

2.3.3 Simulation Results

Simulations are performed to examine the effects of the disturbance compensator. Figure 2.10 shows the simulation results with and without the disturbance compensator. As shown in the figures, with the disturbance compensator the tracking error is reduced especially when the road slope is changing.

2.4 Feedback Controller for Vehicle Following

2.4.1 Desired Distance

The most significant difference between a normal cruise control system and an Advanced Cruise Control (ACC) system is that the ACC system must have a function to maintain the desired distance from the leading vehicle, while the sole function of the normal cruise control system is to maintain the speed of the vehicle at a desired value. Defining and calculating the desired distance which may depend on the state of the vehicle itself as well as its environment, is one of the crucial tasks in developing the ACC system.

When designing the feedback controller, not only the stability of the single system but also the string stability of the vehicle platoon should be considered. String stability should be considered for a platoon of vehicles equipped with identical ACC systems. If the feedback controller is not properly designed, when the very first vehicle of the platoon accelerates or decelerates, the distance errors increases along the vehicle stream. To prevent it, the string stability should be guaranteed.

There are many ways to determine the desired distance. The simplest choice to implement for the desired distance is a constant distance. This method is very simple but has many problems. Intuitively, it is hard to determine a proper value of the constant distance. If the distance is set small, safety becomes an issue at high speeds. On the other hand, a too large distance is inefficient at low speeds. In city driving, large distances will waste very large spaces between cars. Furthermore, the constant distance method can cause a string instability issue [42] or vehicle to vehicle communication is necessary to guarantee the string stability [47].

Instead of the constant distance, the constant time-headway policy is widely used to calculate the desired distance. Time-headway is the time to take for the host vehicle to collide with the lead vehicle when the lead vehicle is suddenly stopped and the host vehicle maintains its speed. If the desired distance is calculated such that the time-headway remains constant, the string stability is guaranteed [42]. In that case, the desired distance is given as:

$$d_{desired} = t_{hw} v_{host} \quad (2.50)$$

where t_{hw} is the time-headway and v_{host} is the velocity of the host vehicle.

However, (2.50) is not valid for complete stops since the desired distance is zero when the host vehicle is completely stopped. Instead, a safety distance which should be maintained between two vehicles when both the following and preceding vehicles are stopped can be introduced[18]. In that case, the desired distance is given as:

$$d_{desired} = t_{hw} v_{host} + d_{offset} \quad (2.51)$$

where d_{offset} is the safety distance. Both the time-headway and the safety distance for a complete stop may vary according to the road condition and the preference of the driver.

2.4.2 Tuning of Feedback Gains

In this section, a Linear Quadratic (LQ) Optimal Control scheme will be studied as a way to find the optimal feedback gains.

Problem Description

In this section, a single host vehicle following a lead vehicle is considered. Each vehicle will be modeled as a double integrator whose input is the acceleration. The desired relative distance is given by (2.51). However, the safety distance will be ignored to make the problem linear. The safety distance can be added later after the state feedback gains are calculated.

For the LQ problem, the output of the system is selected as the error. The state space equations of the system are:

$$\dot{X}(t) = AX(t) + BU(t) \quad (2.52)$$

$$Y(t) = CX(t) \quad (2.53)$$

$$X(t) = [x_l(t) - x(t) \ v_l(t) \ v(t)]^T, \ U(t) = [a_l(t) \ a(t)]^T \quad (2.54)$$

$$A = \begin{bmatrix} 0 & 1 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \ B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \ C = [-1 \ 0 \ t_{hw}] \quad (2.55)$$

where x , v , and a are the position, the velocity, and the acceleration of the host vehicle, respectively and x_l , v_l , and a_l are the position, the velocity, and the acceleration of the lead vehicle, respectively.

Linear Quadratic Optimal Control

If the plant is time invariant, (A, B) is controllable, and (A, C) is observable, the solution of the stationary LQ problem exists for the performance index

$$J = \frac{1}{2} \int_{t_0}^{\infty} X^T(t) Q X(t) + U^T(t) R U(t) dt \quad (2.56)$$

where $Q = C^T C$.

However with the C given in (2.55), (A, C) is not observable. To make it observable with a minimum modification of the performance index, a new C is given as:

$$C = \begin{bmatrix} -1 & 0 & t_{hw} \\ 0 & \epsilon_o & 0 \end{bmatrix} \quad (2.57)$$

The number ϵ_o is a very small number that can be further reduced depending on the available computational resolution.

The input penalty matrix R should be positive definite. Also, it is assumed that we cannot control the lead vehicle acceleration a_l . However if we ignore the lead vehicle acceleration, the

solution of the stationary LQ problem is not guaranteed since in that case (A, B) becomes not controllable or stabilizable. Instead of ignoring the lead vehicle acceleration, we can penalize it a lot so that the magnitude of it is limited to be very small (the actual lead vehicle acceleration will depend on the driver and this can be considered as a disturbance). To do that, R can be selected as:

$$R = \lambda \begin{bmatrix} 1/\epsilon_c & 0 \\ 0 & 1 \end{bmatrix} \quad (2.58)$$

where λ is a design parameter to tune. The number ϵ_c is a very small number that can be further reduced depending on the available computational resolution. The state feedback control law is given as:

$$U(t) = -R^{-1}B^T P_+ X(t) = -KX(t) \quad (2.59)$$

where P_+ is the positive definite solution of the algebraic Riccati equation

$$A^T P + PA - PBR^{-1}B^T P + C^T C = 0 \quad (2.60)$$

With $t_{hw} = 2$, $\lambda = 1$, and $\epsilon_o = \epsilon_c = 1e-6$, the state feedback gain is calculated to be:

$$K = \begin{bmatrix} 1.0102e-6 & 2.5061e-6 & -4.4949e-6 \\ -1.0000 & -0.4495 & 2.4495 \end{bmatrix} \quad (2.61)$$

As desired, the gains for the lead vehicle acceleration are very small. The feedback control law can be written as:

$$a(t) = -(t_{hw}v(t) - (x_l(t) - x(t))) - 0.4495(-(v_l(t) - v(t))) \quad (2.62)$$

Assuming $v(t)$ is constant, the control law is:

$$a(t) = -err(t) - 0.4495err(t) \quad (2.63)$$

where $err(t) = t_{hw}v(t) - (x_l(t) - x(t))$. As shown in (2.63), the state feedback law is the same as the PD controller.

Linear Quadratic Optimal Control with Integrator

To add an integral controller to the LQ controller, a Linear Quadratic Optimal Control with Integrator (LQI) is introduced. The plant is given by (2.52) ~ (2.55) where the matrix C is modified as (2.57). The cost function of the LQI problem is:

$$J = \int_{t_0}^{\infty} [Y(t) - r]^T Q_y [Y(t) - r] + W^T(t) R W(t) dt \quad (2.64)$$

where Q_y and R are symmetric and positive definite weight matrices, r is a constant set point vector, and W is the derivative of U .

The new state vector is defined as $\tilde{X}(t) = [E^T(t) \quad \dot{X}^T(t)]^T$ where $E(t) = Y(t) - r$. Then the state equation is:

$$\frac{d}{dt} \tilde{X} = \begin{bmatrix} 0 & C \\ 0 & A \end{bmatrix} \tilde{X}(t) + \begin{bmatrix} 0 \\ B \end{bmatrix} W(t) = \tilde{A} \tilde{X}(t) + \tilde{B} W(t) \quad (2.65)$$

The cost function is:

$$J = \int_{t_0}^{\infty} \tilde{X}^T(t) \begin{bmatrix} Q_y & 0 \\ 0 & 0 \end{bmatrix} \tilde{X}(t) + W^T(t) R W(t) dt \quad (2.66)$$

and the problem reduces to a standard LQ problem. With $Q_y = \begin{bmatrix} 1 & 0 \\ 0 & \epsilon_o \end{bmatrix}$, (\tilde{A}, \tilde{B}) is controllable and $(\tilde{A}, Q_y^{\frac{1}{2}} [I_{2 \times 2} \quad 0_{2 \times 3}])$ is observable and the Riccati equation has a stationary positive definite solution.

With $Q_y = \begin{bmatrix} 1 & 0 \\ 0 & 1e-6 \end{bmatrix}$ and $R = \begin{bmatrix} 1e6 & 0 \\ 0 & 1 \end{bmatrix}$, the PID control law is:

$$a(t) = -0.9804err(t) - 0.4806e\dot{r}r(t) - \int_{t_0}^t err(\tau) d\tau \quad (2.67)$$

2.4.3 Simulation Results

Figure 2.11 shows the simulation result comparing the PD (LQ) 2.63 and the PID (LQI) 2.67 controllers. Both controllers show stable performances while PID shows better tracking performance when the lead vehicle accelerates or decelerates. Under our assumption, the lead vehicle acceleration is considered as a disturbance. Since the Integral controller has the disturbance rejection property, it shows better performance especially when the lead vehicle acceleration is nonzero.

Figure 2.12 shows the simulation result of a vehicle platoon. In the platoon, the first lead vehicle is assumed to be driven by a human driver. The following vehicles are equipped with the identical ACC systems. For simplicity, the engine and the brake are assumed to be first order and no tire dynamics are included. As shown in the figure, the designed PID controller is string stable and the peak value of the error decreases along the vehicle stream.

2.5 Summary

In this chapter, a longitudinal vehicle model was introduced. External disturbances such as the aerodynamic drag, the rolling resistance, and the gravity force due to the road slope were considered. To better predict the behavior of the vehicle while preventing too complicated structure, the discrete event engine model [30, 31], the single state hydraulic brake model [13], and the dynamic deflection tire model [22] were included in the vehicle model.

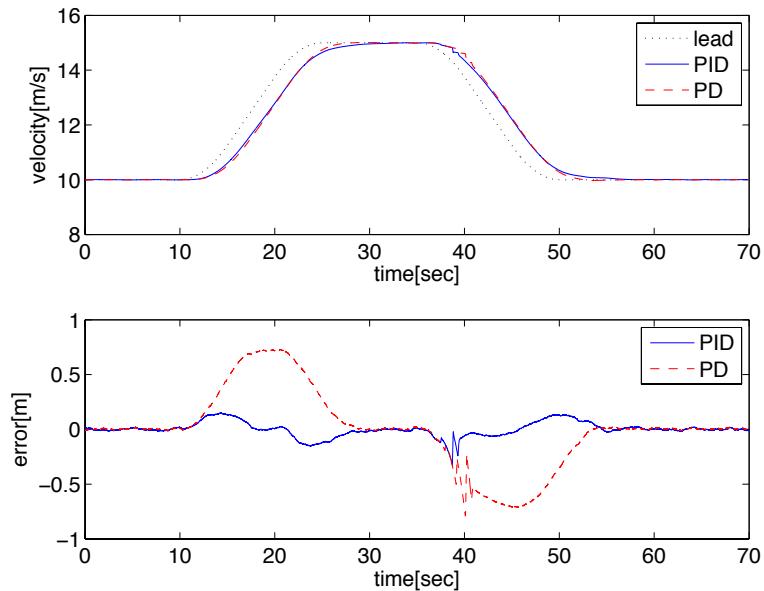


Figure 2.11: Simulation results with the PID and PD controllers

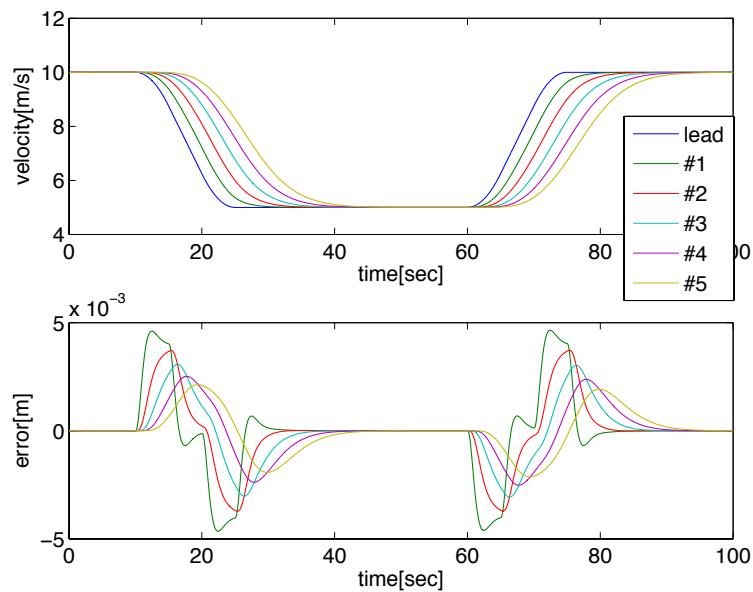


Figure 2.12: Simulation results of a vehicle platoon with the PID controller

A recursive least square method was used to estimate the vehicle mass, the road slope, and the aerodynamic resistance. A vector-type forgetting factor was utilized to estimate the parameter where some of the parameters were changing frequently while others were changing very slowly. The constant trace method was also used to achieve more stable performance with the forgetting factor especially when there was not enough excitation. A model based disturbance compensator was designed to compensate the external disturbances.

Linear Quadratic (LQ) optimal control was studied to find the optimal feedback gains for a single vehicle following case. The vehicle was modeled as a double integrator and the state space realization matrices were slightly modified to make the system controllable and observable. The designed LQ controller was nothing but a PD controller. Linear Quadratic control with Integrator (LQI) was introduced to add the integral action. The LQI controller showed better tracking performance than the LQ controller when the lead vehicle accelerated or decelerated. The constant time-headway policy was used to calculate the desired distance and the simulation result of a vehicle platoon showed string stable performances.

Chapter 3

Virtual Lead Vehicle Scheme

3.1 Introduction

The ACC system should be able to perform two different tasks according to the surrounding situation. If there is a lead vehicle detected, the ACC system should follow the lead vehicle maintaining an appropriate relative distance. When no lead vehicles are detected in front of the host vehicle, the ACC system should maintain the desired velocity like the conventional cruise control system. It is important that not only both modes should operate properly and stably, but also the transient performance should be smooth and safe when the mode is switched from one to the other.

A mode switching scheme is one method which deals with the presence of a lead vehicle. The control algorithm switches between the speed control and the distance control modes based on the relative distance to the lead vehicle [63, 58]. The speed control algorithm is used when no lead vehicle is detected or the lead vehicle is farther than a specified relative distance. When the lead vehicle is close, the distance control algorithm is used. One downside of this method is that two different feedback loops are required: velocity and distance control loops. If conventional PID feedback controllers are used, integrator windup must also be properly taken into consideration.

Another drawback of the mode switching scheme is that the transient maneuver is not considered. When the speed control algorithm is activated, if a new lead vehicle cuts in from a side lane or a slower lead vehicle approaches from the front, the mode is abruptly changed to the distance control which can cause unpleasant motion of the host vehicle. A model predictive control method was studied to control the transitional maneuver of the ACC system [4]. In [4], the optimal motion of the host vehicle was calculated by solving an optimization problem over a time horizon and the initial optimal control input was utilized as the control command. The model predictive controller was appropriate only for the distance control mode or the transition from the speed control to the distance control mode. Still, a separate speed controller was required to control the host vehicle during the speed control mode. Also, the model predictive control requires significant computational efforts and a

fast processor is necessary for real time application.

A virtual lead vehicle scheme [20] is a method which can replace the complicated switching mechanism. When no lead vehicle is detected, the controller generates a virtual vehicle whose speed is the same with the desired speed. When there is a lead vehicle, the speed and the position of the virtual lead vehicle are set to be those of the actual vehicle. The host vehicle always follows the virtual lead vehicle with an appropriate distance control algorithm (i.e. a PID controller) and no switching scheme between the speed and the distance control is necessary.

Additional advantage of the virtual lead vehicle scheme is that the motion of the host vehicle can be smoothly controlled when a new lead vehicle cuts in or the current lead vehicle cuts out. If the host vehicle follows the actual lead vehicle, the cutting in/out of the lead vehicle is a step change of the error signal. However, if the host vehicle follows the virtual lead vehicle and the speed and the position of the virtual lead vehicle merge smoothly from those of the old lead vehicle to those of the new lead vehicle, the host vehicle also moves smoothly.

Since the virtual vehicle has no internal dynamics or delay, the position and the velocity of it can be set without any physical limitations. However, a sudden change of the position or velocity of the virtual lead vehicle can cause an undesirable motion of the host vehicle which can make the passengers uncomfortable. To prevent this, the virtual lead vehicle is modeled as a double integrator whose input is the acceleration and outputs are the velocity and the position. A smooth switching can be achieved by controlling the virtual lead vehicle with limited acceleration and jerk.

This chapter is organized as follows. In Section 3.2, the modes of the ACC system and the concept of the virtual lead vehicle scheme are introduced. In Section 3.3, the linear quadratic controller is designed to control the virtual lead vehicle. In Section 3.4, variable weights are utilized for the linear quadratic controller to better control the virtual lead vehicle during transient motion. The simulation results are shown to support the suggested ideas. The stability of the suggested controller is analyzed in Section 3.5. In Section 3.6, a Disturbance Observer is utilized to estimate and compensate the acceleration of the lead vehicle. Section 3.7 summarizes the chapter.

3.2 Concept of the Virtual Lead Vehicle Scheme

The ACC system should be able to perform two modes: speed control and distance control. The modes depend on the desired and actual distance between the lead and host vehicles. One way to implement these two modes is by designing two different controllers and switching between them. However this can cause various problems. Another way is the virtual lead vehicle scheme which makes the switching between the speed control algorithm and the distance control algorithm unnecessary [20]. With the virtual lead vehicle scheme, the control system structure is simplified.

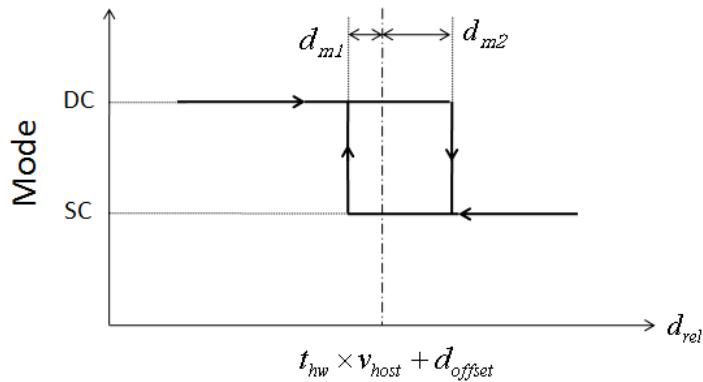


Figure 3.1: Mode switching logic

3.2.1 Mode Decision of the ACC system

As discussed in previous chapters, when the host vehicle follows a lead vehicle the desired distance is given by:

$$d_{desired} = t_{hw}v_{host} + d_{offset} \quad (3.1)$$

where t_{hw} is the time headway, v_{host} is the velocity of the host vehicle, and d_{offset} is the safety distance to be maintained in case of a complete stop. The time headway is the time to take for the host vehicle to collide with the lead vehicle when the lead vehicle is suddenly stopped and the host vehicle maintains its speed. The desired time headway is decided by the driver when the ACC system is activated. Using (3.1), a simple scheme to determine the proper mode is given by:

$$mode = \begin{cases} DC & d_{rel} \leq d_{desired} \\ SC & d_{rel} > d_{desired} \end{cases} \quad (3.2)$$

where DC is the distance control, SC is the speed control, and d_{rel} is the relative distance to the lead vehicle. Using this scheme, however, may cause chattering in the algorithm if the distance between the lead and host vehicle is too close to the desired distance. To prevent chattering, marginal distances are introduced as:

$$mode = \begin{cases} SC \rightarrow DC & d_{rel} - d_{m1} \leq d_{desired} \\ DC \rightarrow SC & d_{rel} + d_{m2} > d_{desired} \end{cases} \quad (3.3)$$

where d_{m1} and d_{m2} are marginal distances. This introduces hysteresis in the mode switching as shown in Figure 3.1. If there is no lead vehicle, d_{rel} is infinite and the speed control mode is utilized.

3.2.2 Mode Switching Scheme

Figure 3.2 shows the block diagram of the mode switching scheme. The input signal to the distance control algorithm is the distance error which is given by:

$$e_1 = \begin{cases} d_{rel} - d_{desired} & mode = DC \\ 0 & mode = SC \end{cases} \quad (3.4)$$

The input signal to the speed control algorithm is the speed error which is given by:

$$e_2 = \begin{cases} v_{desired} - v_{host} & mode = SC \\ 0 & mode = DC \end{cases} \quad (3.5)$$

where $v_{desired}$ is the desired speed set by the driver. There is an additional switch to cancel the output of the inactivated mode:

$$a_d = \begin{cases} a_1 & mode = DC \\ a_2 & mode = SC \end{cases} \quad (3.6)$$

If PID controllers are used for both algorithms, the integral controller of the inactivated mode should be initialized when the mode is switched.

3.2.3 Virtual Lead Vehicle Scheme

Figure 3.3 shows the conceptual sketch of the virtual lead vehicle scheme. In the virtual lead vehicle scheme, the host vehicle follows the virtual lead vehicle with a distance control algorithm in the absence of the actual lead vehicle. When a lead vehicle is detected inside the desired relative distance and the speed of the lead vehicle is slower than the desired speed, the position and the velocity of the virtual lead vehicle is set to be those of the actual lead vehicle. When no lead vehicle is detected or if the speed of the lead vehicle is too fast, the virtual lead vehicle scheme prevents switching to the speed control algorithm by limiting the velocity of the virtual lead vehicle to the desired speed. In that case, the position of the virtual lead vehicle is updated by integrating the velocity.

As a result, the speed control algorithm is not utilized with the virtual lead vehicle scheme and the overall controller structure has only one loop which is the distance control algorithm as described in Figure 3.4.

3.3 Control of the Virtual Lead Vehicle in Transient

In this section, a double integrator model will be suggested as the model of the virtual lead vehicle and a linear quadratic controller will be designed to control the speed and the position of the virtual lead vehicle in transient motions. The control scheme deciding the speed and position of the virtual lead vehicle during the speed control and the distance

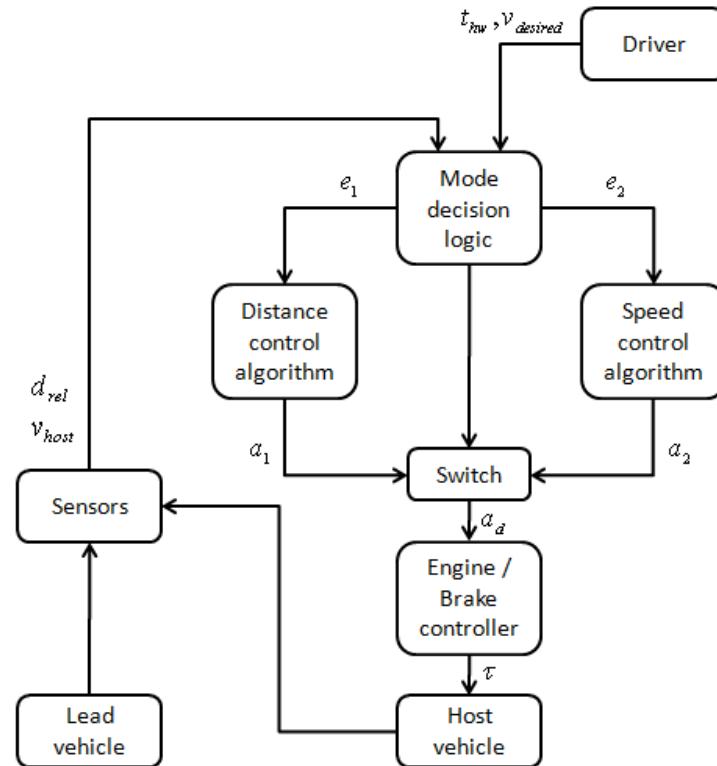


Figure 3.2: Block diagram of the mode switching scheme



Figure 3.3: Concept of the virtual lead vehicle scheme

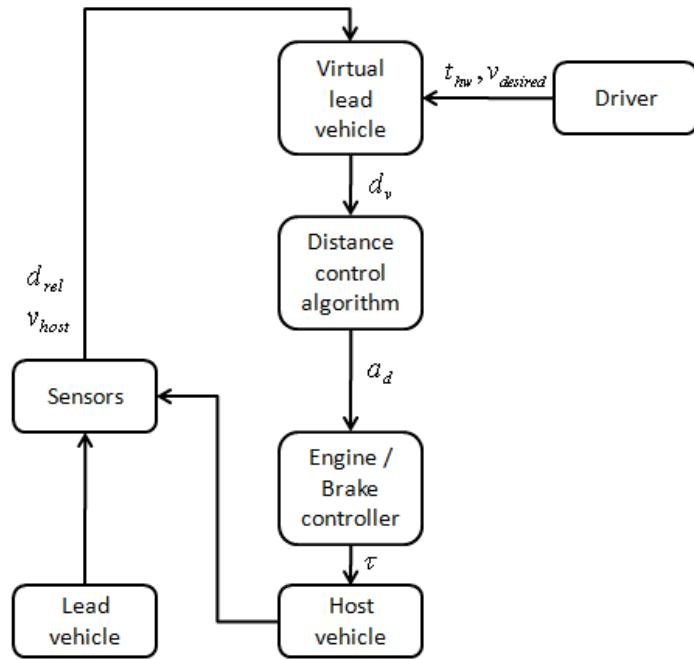


Figure 3.4: Block diagram of the virtual lead vehicle scheme

control algorithms are described in Section 3.2.3. However, when the mode is changed from one to the other due to cutting in or out of the lead vehicle, the speed and the position of the virtual lead vehicle should be smoothly change from those of the old lead vehicle to those of the new lead vehicle. A double integrator model and a linear quadratic control scheme are strong candidates to achieve this desired behavior.

3.3.1 Double Integrator Model for the Virtual Lead Vehicle

When the current lead vehicle cuts out from the current lane or a new lead vehicle cuts in from a side lane, there are step changes in the position and the speed of the lead vehicle. If the host vehicle directly follows the actual lead vehicle, undesirable transient oscillation or overshoots can be generated. With the virtual lead vehicle scheme, however, the virtual vehicle's position and speed are controlled to merge smoothly with those of the new lead vehicle. Hence, the host vehicle will also move smoothly.

In this thesis, the goal is to design a controller which makes the virtual lead vehicle moves smoothly from an initial state to the desired state. The position and the speed of the virtual lead vehicle can be set arbitrarily by the controller. However, the host vehicle should react slowly and smoothly given any initial conditions to avoid producing unnecessary uncomfortable motions. The virtual lead vehicle is considered as a double integrator whose

input is the acceleration. The continuous time model of the virtual lead vehicle is:

$$\begin{bmatrix} \dot{x}_{vl}(t) \\ \dot{v}_{vl}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_{vl}(t) \\ v_{vl}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} a_{vl}(t) = A \begin{bmatrix} x_{vl}(t) \\ v_{vl}(t) \end{bmatrix} + B a_{vl}(t) \quad (3.7)$$

where x_{vl} , v_{vl} , and a_{vl} are the position, speed, and the acceleration of the virtual lead vehicle, respectively.

The task of the controller is to make the virtual vehicle follow the actual lead vehicle smoothly. In this thesis, to follow means that both the position and the velocity of the virtual vehicle converge to those of the actual vehicle. The errors are defined as:

$$\begin{bmatrix} e_x(t) \\ e_v(t) \end{bmatrix} = \begin{bmatrix} x_{vl}(t) - x_{al}(t) \\ v_{vl}(t) - v_{al}(t) \end{bmatrix} \quad (3.8)$$

where x_{al} and v_{al} are the position and the speed of the actual lead vehicle, respectively. The error dynamics is:

$$\begin{bmatrix} \dot{e}_x(t) \\ \dot{e}_v(t) \end{bmatrix} = A \begin{bmatrix} e_x(t) \\ e_v(t) \end{bmatrix} + B(a_{vl}(t) - a_{al}(t)) \quad (3.9)$$

where a_{al} is the acceleration of the actual lead vehicle. It is assumed that the relative distance and velocity of the actual lead vehicle are measurable. Since the position and velocity of the virtual lead vehicle is either known and fixed, or dependent on the actual lead vehicle, the relative distance and the relative velocity between the virtual and the actual lead vehicle can be computed. However, the behavior of the lead vehicle is unknown and hence the actual lead vehicle acceleration a_{al} is considered as a disturbance.

3.3.2 Linear Quadratic Control

The goal is to make the states in (3.9) converge zero. One promising way to control this system is doing state feedback using the steady state Linear Quadratic (LQ) optimal controller. The cost function is a quadratic sum of the position error, the velocity error, and the acceleration given as:

$$J = \frac{1}{2} \int_{t_0}^{\infty} \lambda_x e_x(t)^2 + \lambda_v e_v(t)^2 + \lambda_a a_{vl}(t)^2 dt \quad (3.10)$$

Without the constraints, the problem can be formulated as a stationary LQ problem.

$$J = \frac{1}{2} \int_{t_0}^{\infty} X(t)^T Q X(t) + a_{vl}(t)^T R a_{vl}(t) dt \quad (3.11)$$

where $X(t) = [e_x(t) \ e_v(t)]^T$, $Q = \begin{bmatrix} \lambda_x & 0 \\ 0 & \lambda_v \end{bmatrix}$, and $R = \lambda_a$. Since the system is time invariant, controllable, and observable, the optimal solution of the stationary LQ problem is given as:

$$a_{vl}(t) = -R^{-1}B^T P_+ X(t) \quad (3.12)$$

where P_+ is the positive (semi-)definite solution of the algebraic Riccati equation

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (3.13)$$

In this case, there are only two states and the Riccati equation can be algebraically solved. The feedback control law is given as:

$$a_{LQ}(t) = - \begin{bmatrix} \sqrt{\frac{\lambda_x}{\lambda_a}} & \sqrt{2\sqrt{\frac{\lambda_x}{\lambda_a}} + \frac{\lambda_v}{\lambda_a}} \end{bmatrix} X(t) \quad (3.14)$$

The solution given in (3.14) is for the case without constraints. For actual implementation, however, acceleration and jerk have to be constrained to limit passenger discomfort. These constraints are given by:

$$a_{min} \leq a_{vl}(t) \leq a_{max} \quad (3.15)$$

$$j_{min} \leq j_{vl}(t) \leq j_{max} \quad (3.16)$$

The acceleration of the virtual lead vehicle is given as:

$$a_{vl}(t) = \max\{\min(a_{LQ}, a_{max}, a(t - \Delta t) + j_{max}\Delta t), a_{min}, a(t - \Delta t) + j_{min}\Delta t\} \quad (3.17)$$

where a_{LQ} is the solution of the unconstrained problem given in (3.14), and Δt is the controller sampling time.

3.3.3 Comparison with the Model Predictive Control

In (3.15) ~ (3.17), the acceleration and the jerk constraints are applied to only the current time and not the whole time horizon. Ideally, the constraints should be applied to the whole time horizon. To do so the Model Predictive Control (MPC) method can be applied. The cost function for MPC is the same as (3.10) and the inequality constraints described in (3.15) ~ (3.16) are applied to the whole time horizon. Figure 3.5 compares the simulation results with MPC and LQ control. As shown in the figure, the two results are very similar to each other. However, the MPC method requires a lot more calculation efforts than the LQ. As a result, using the MPC method will require a faster processor for real time application. Since the results are almost identical, the LQ method can efficiently replace the MPC scheme and will be used in this thesis.

3.4 Linear Quadratic Control With Variable Weights

In the cost function of the LQ problem given in (3.10), the weights λ_x , λ_v , and λ_a are parameters to tune by the controller designer. The state feedback gains are defined as functions of these weights. To control the motion of the virtual lead vehicle smoothly, state dependent weights are suggested and simulation result shows better performance in the viewpoint of passenger comfort and fuel efficiency [21].

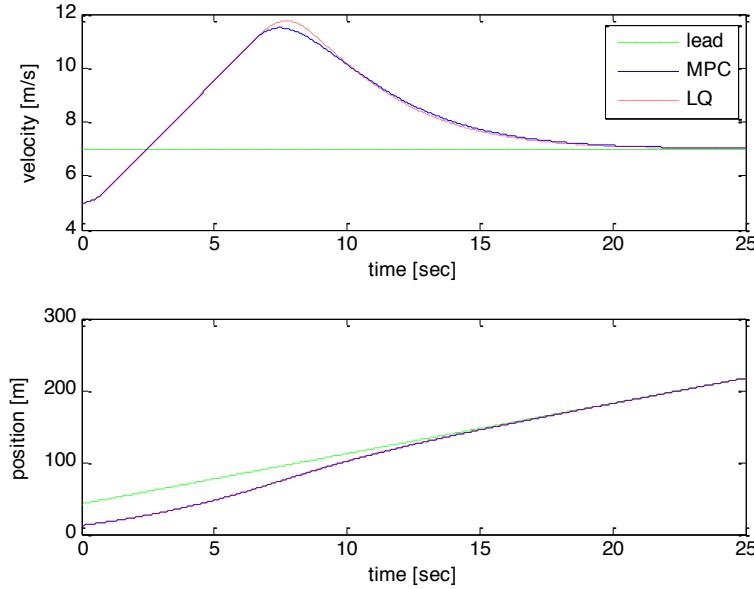


Figure 3.5: Simulation results with MPC and LQ control: Position and velocity of the virtual lead vehicle

3.4.1 Tuning of the LQ Weights

The weights of the cost functions can be tuned using the return difference equation. The closed loop poles of the system are the left half plane solution of the return difference equation [23] :

$$s^4 - \frac{\lambda_v}{\lambda_a} s^2 + \frac{\lambda_x}{\lambda_a} = 0 \quad (3.18)$$

The symmetric root locus is shown in Figure 3.6 varying λ_a from 0 to ∞ . Considering the passenger's comfort and the fuel efficiency, the damping should be large to prevent a large overshoot. Also it is desirable to have the real part of the poles to be as small as possible for faster responses. The point marked in Figure 3.6 is such a point and the weights are:

$$\lambda_x = 1, \lambda_v = 10, \lambda_a = 25 \quad (3.19)$$

3.4.2 Simulation Results with Constant Weights

Figure 3.7 shows the simulation results with the constant weights given in (3.19). The simulation describes the case when the current lead vehicle cuts out and a new lead vehicle is detected. As shown in the figure, the virtual lead vehicle initially accelerates to chase the new lead vehicle and then decelerates to the speed of the new lead vehicle. The initial acceleration is constrained by the acceleration and the jerk constraints. However, the large

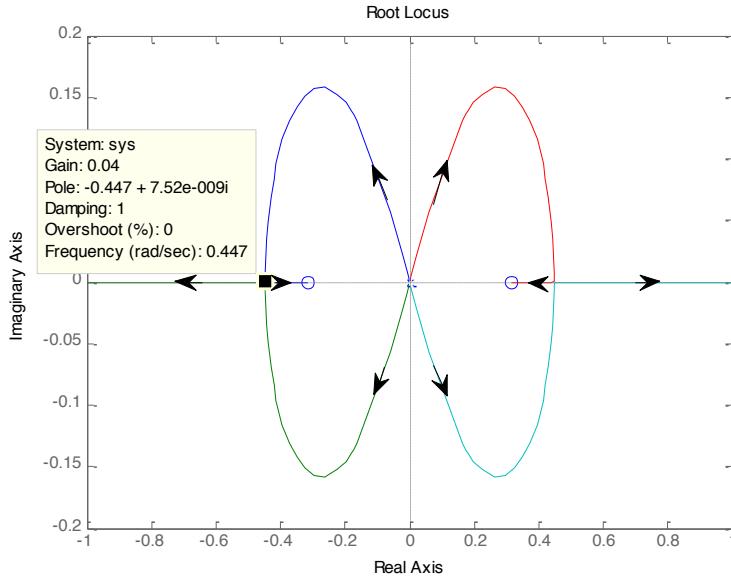


Figure 3.6: The symmetric root locus - Arrows indicate decreasing λ_a

velocity overshoot is undesirable. To prevent the overshoot, the acceleration is penalized more and the simulation results show that the velocity overshoot is reduced as λ_a is increased.

However, if λ_a is increased, the response of the closed loop system gets slower. This can be also found in Figure 3.6 where two poles are moving close to the origin if λ_a is increased more than the critical value. Slow response can make the driver uncomfortable especially when a new lead vehicle cuts in and the relative distance to the new lead vehicle is closer than the desired distance. Another solution is to increase λ_v . In that case, when the velocity error is positive, the feedback term of the velocity error is negative and the acceleration can be reduced. However, when a new lead vehicle cuts in and the position error is positive, the large λ_v can cause undesirable system behavior.

3.4.3 LQ with Variable Weights

To prevent the large velocity overshoot while maintaining a fast response in the cutting in situation, variable weights are introduced. The desired performance of the virtual lead vehicle is described in Table 3.1. When the position error is negative and the velocity error is positive, it is desirable to maintain the speed to prevent a large velocity overshoot. To do that the position error should be less penalized while the velocity error and the acceleration should be penalized more. When the position error is positive and the velocity error is negative, which describes the case when a lead vehicle faster than the virtual lead vehicle cuts in, the virtual lead vehicle should smoothly decelerate and merge to the new lead vehicle.

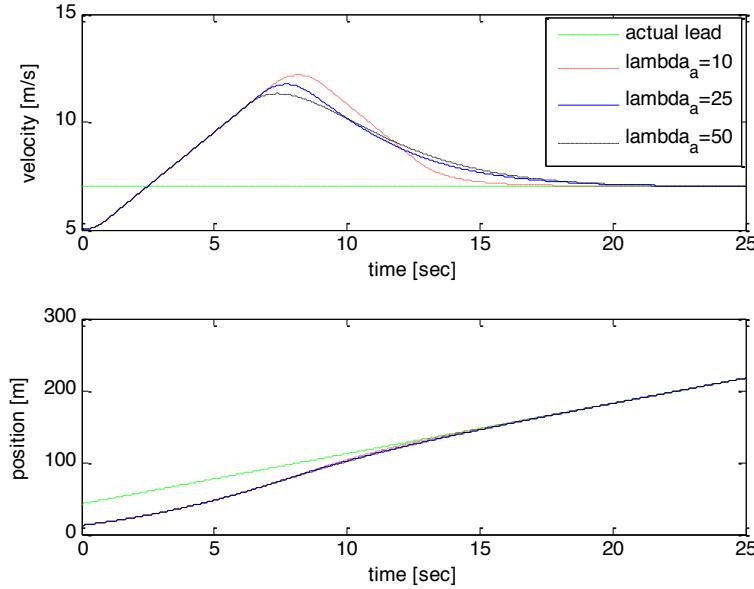


Figure 3.7: Simulation results with constant weights position and velocity of the virtual lead vehicle

The position error should be more penalized than the speed error while the acceleration is penalized more to make the motion smooth. When both the position and the velocity errors are positive, it is a dangerous situation and it is desirable to decelerate the host vehicle quickly. In that case, both errors should be penalized while the acceleration should be less penalized to ensure a fast response.

Since the desired values of the weights are changing according to the sign of the errors, the variable weights are suggested as:

$$\lambda_x = \lambda_{x0} \left(1 + \frac{2}{\pi} \tan^{-1}(\phi_x e_x(t)) \right) \quad (3.20)$$

$$\lambda_v = \lambda_{v0} \left(1 + \frac{2}{\pi} \tan^{-1}(\phi_v e_v(t)) \right) \quad (3.21)$$

$$\lambda_a = \lambda_{a0} \left(1 - \frac{4}{\pi^2} \tan^{-1}(\phi_x e_x(t)) \tan^{-1}(\phi_v e_v(t)) \right) \quad (3.22)$$

The arctangent function is introduced instead of the sign function to make it continuous and ϕ 's are constants describing the slope of the arctangent function near the origin.

3.4.4 Simulation Results with Variable Weights

Figures 3.8 ~ 3.10 show the simulation results with the variable weights given in (3.20) ~ (3.22). As shown in the figure, the velocity peak is reduced with the variable weights. Figure

Sign of errors	Desired performance	Desired λ_x	Desired λ_v	Desired λ_a
$e_x < 0, e_v > 0$	Do not accelerate	small	large	large
$e_x > 0, e_v < 0$	Decelerate smoothly	large		large
$e_x > 0, e_v > 0$	Decelerate fast	large	large	small

Table 3.1: Desired performance and weights

3.9 shows the weights. As shown in Figure 3.9, when the velocity error becomes positive, the weights on the velocity and the acceleration are increased to prevent the velocity overshoot. Figure 3.10 shows the velocity and the fuel consumption of the host vehicle. The engine and the brake dynamics are assumed to be first order with time constants 0.5 and 0.04 respectively. The fuel consumption is calculated with a VT model [2]. As shown in Figure 3.10, with the variable weights, the large velocity overshoot is avoided and fuel efficiency is improved. The square sums of the acceleration and the jerk, which can represent a measure of passenger comfort, are $0.2014[m^2/sec^4]$ and $0.0814[m^2/sec^6]$ respectively with variable weights, and $0.4003[m^2/sec^4]$ and $0.7638[m^2/sec^6]$ respectively with the constant weights.

Figures 3.11 ~ 3.13 show the simulation results with different situation where a slower lead vehicle cuts in. As shown in the figures, with variable weights, the motions of the virtual lead vehicle and the host vehicle are smoother. Also the fuel efficiency is better than the constant weights. The square sums of the acceleration and the jerk are $0.1050[m^2/sec^4]$ and $0.0586[m^2/sec^6]$ respectively with the variable weights, and $0.1527[m^2/sec^4]$ and $0.1865[m^2/sec^6]$ respectively with the constant weights.

3.5 Stability Analysis of the Linear Quadratic Controller with Variable Weights

Since the weights in the cost function of the linear quadratic controller are state dependent, the stability and the robust features of the general linear quadratic controller are not guaranteed anymore. Also the stability of the suggest scheme should be proved.

The closed loop system dynamics is given by:

$$\dot{e}_x(t) = e_v(t) \quad (3.23)$$

$$\dot{e}_v(t) = -\sqrt{\frac{\lambda_x(e_x)}{\lambda_a(e_x, e_v)}} e_x(t) - \sqrt{2\sqrt{\frac{\lambda_x(e_x)}{\lambda_a(e_x, e_v)}} + \frac{\lambda_v(e_v)}{\lambda_a(e_x, e_v)}} e_v(t) \quad (3.24)$$

Since $\sqrt{\frac{\lambda_x}{\lambda_a}}$ and $\sqrt{2\sqrt{\frac{\lambda_x}{\lambda_a}} + \frac{\lambda_v}{\lambda_a}}$ are always positive, the equilibrium point of the system is only the origin and the origin is a stable equilibrium point. However this does not guarantee that the solution converges to zero with an arbitrary initial condition. To show this, a phase plane analysis is performed.

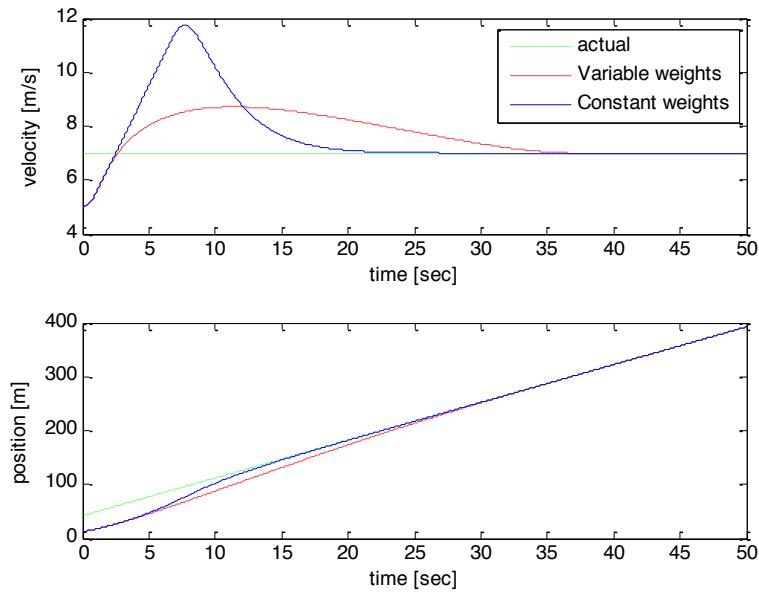


Figure 3.8: Simulation results when a lead vehicle cuts out position and velocity of the virtual lead vehicle

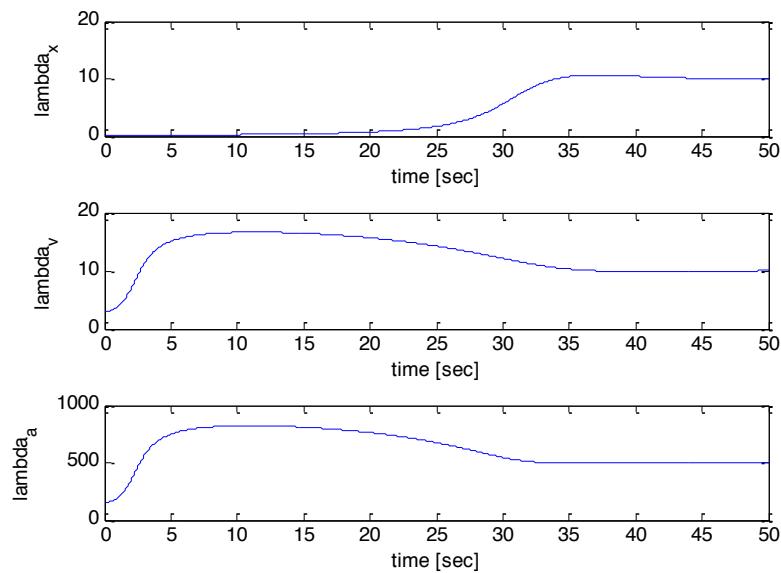


Figure 3.9: Simulation results when a lead vehicle cuts out variable weights

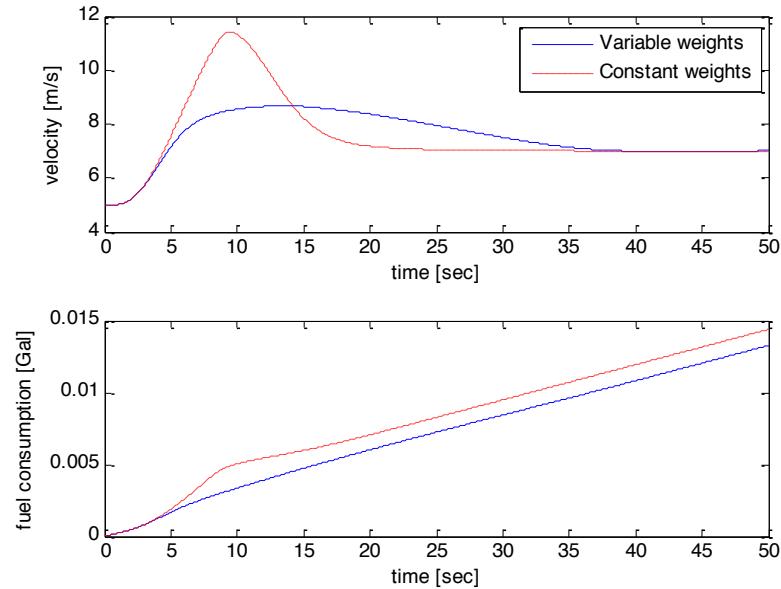


Figure 3.10: Simulation results when a lead vehicle cuts out position and velocity of the host vehicle

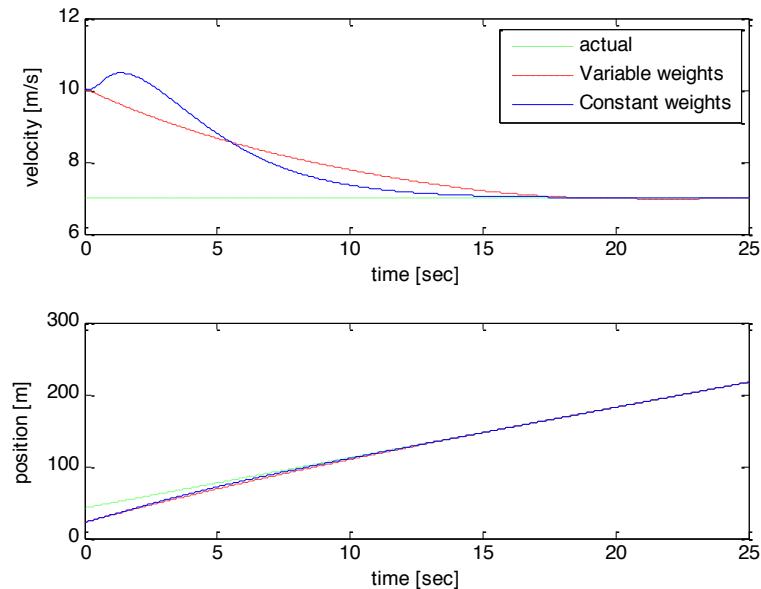


Figure 3.11: Simulation results when a lead vehicle cuts in position and velocity of the virtual lead vehicle

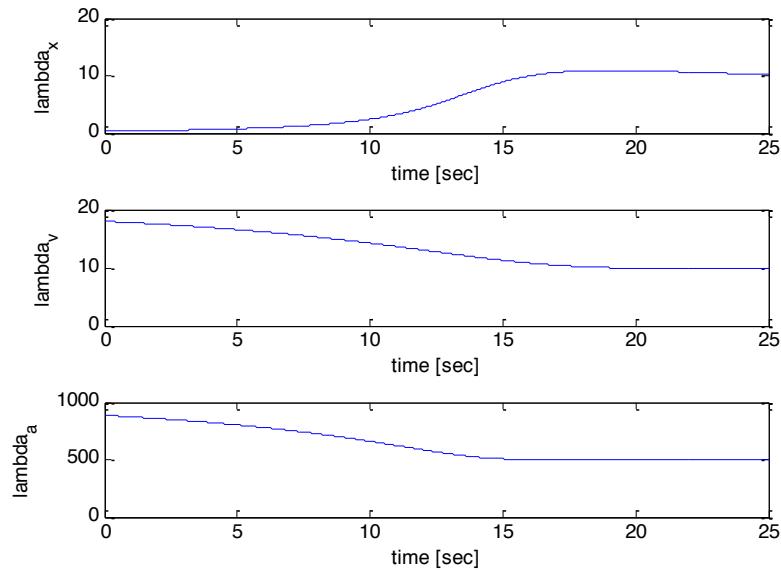


Figure 3.12: Simulation results when a lead vehicle cuts in variable weights

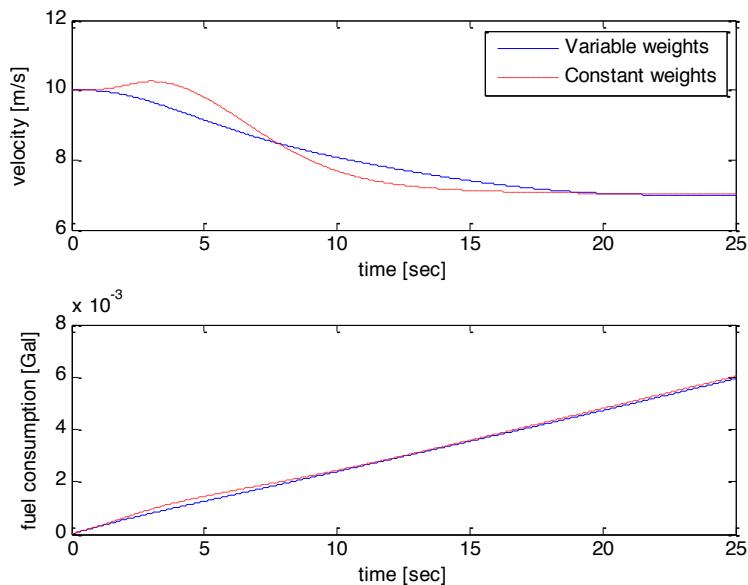


Figure 3.13: Simulation results when a lead vehicle cuts out position and velocity of the host vehicle

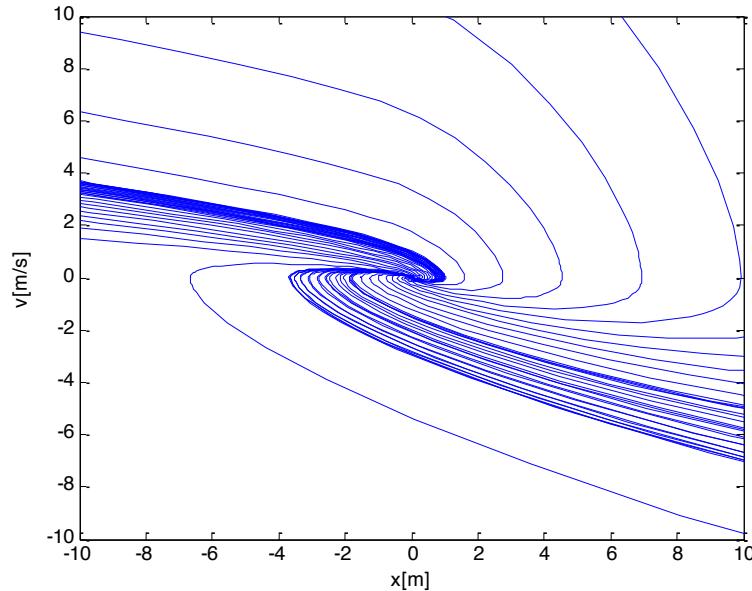


Figure 3.14: Phase plane plot of the closed loop system

Figure 3.14 shows the phase plane plot of the closed loop system. As expected from (3.23) the solution trajectory moves to the positive x direction in the first and the second quadrants, and to the negative x direction in the third and the forth quadrants. Also as expected from (3.24) the solution trajectory goes to the negative v direction in the first quadrant and to the positive v direction in the third quadrant. As a result the solution will meet the x axis at some point. The solution from a point on the positive x axis will meet the positive x axis again. If the second point is closer to the origin than the first point, it can be concluded that all initial points converge to the origin and the system is stable.

Since the system is nonlinear, it is hard to find the solution explicitly. However it is possible to find a linear switching system which always behaves worse than the original system. From (3.23) and (3.24), the slope on the phase plane is:

$$\frac{de_v}{de_x} = -\sqrt{\frac{\lambda_x(e_x)}{\lambda_a(e_x, e_v)}} \frac{e_x(t)}{e_v(t)} - \sqrt{2\sqrt{\frac{\lambda_x(e_x)}{\lambda_a(e_x, e_v)}} + \frac{\lambda_v(e_v)}{\lambda_a(e_x, e_v)}} \quad (3.25)$$

The convergence is slower if the slope is large (steep when positive and flat when negative). With the weights given in (3.20) \sim (3.22), the behavior of the system is always better than the worst case system given by:

$$\dot{e}_x(t) = e_v(t) \quad (3.26)$$

$$\dot{e}_v(t) = -a_1 e_x(t) - a_2 e_v(t) \quad (3.27)$$

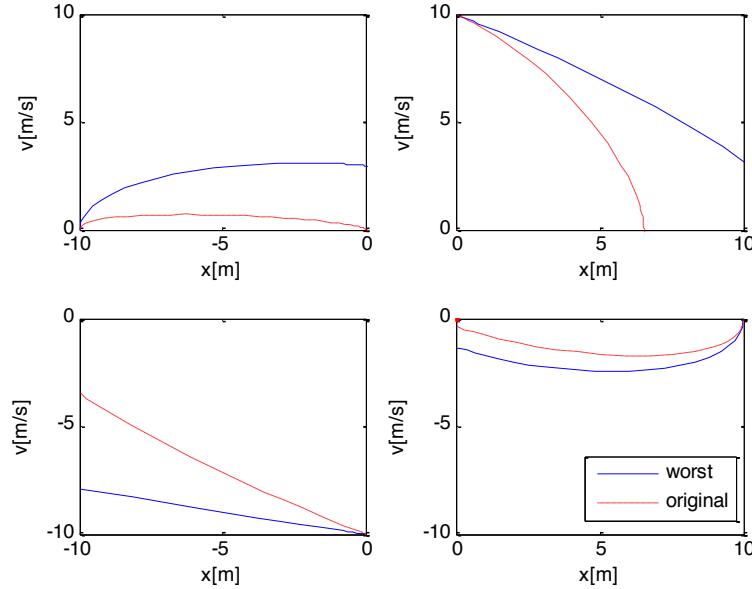


Figure 3.15: Phase plane plot of the closed loop system at each quadrant comparing the worst case gains and the original gains

$$a_1 = \begin{cases} \sqrt{\frac{\lambda_{x0}}{\lambda_{a0}}} & x > 0, v > 0 \\ \sqrt{\frac{2\lambda_{x0}}{\lambda_{a0}}} & x > 0, v < 0 \\ \sqrt{\frac{\lambda_x(x_{min})}{\lambda_{a0}}} & x < 0, v > 0 \\ \sqrt{\frac{\lambda_{x0}}{\lambda_{a0}}} & x < 0, v < 0 \end{cases} \quad (3.28)$$

$$a_2 = \begin{cases} \sqrt{2\sqrt{\frac{\lambda_{x0}}{\lambda_{a0}}} + \frac{\lambda_{v0}}{\lambda_{a0}}} & x > 0, v > 0 \\ \sqrt{2\sqrt{\frac{\lambda_{x0}}{2\lambda_{a0}}} + \frac{\lambda_v(v_{min})}{2\lambda_{a0}}} & x > 0, v < 0 \\ \sqrt{2\sqrt{\frac{\lambda_x(x_{min})}{\lambda_{a0}}} + \frac{\lambda_v(v_{min})}{\lambda_{a0}}} & x < 0, v > 0 \\ \sqrt{2\sqrt{\frac{\lambda_x(x_{min})}{2\lambda_{a0}}} + \frac{\lambda_{v0}}{2\lambda_{a0}}} & x < 0, v < 0 \end{cases} \quad (3.29)$$

where x_{min} and v_{min} are the minimum position error and the minimum velocity error, respectively. In our case v_{min} is $-15[m/sec]$ and x_{min} is $-32[m]$.

Figure 3.15 shows the simulation results comparing the worst case gains and the original gains. As shown in the figures, with the worst case gains the performance is worse at all four quadrants in the sense of converging to the origin.

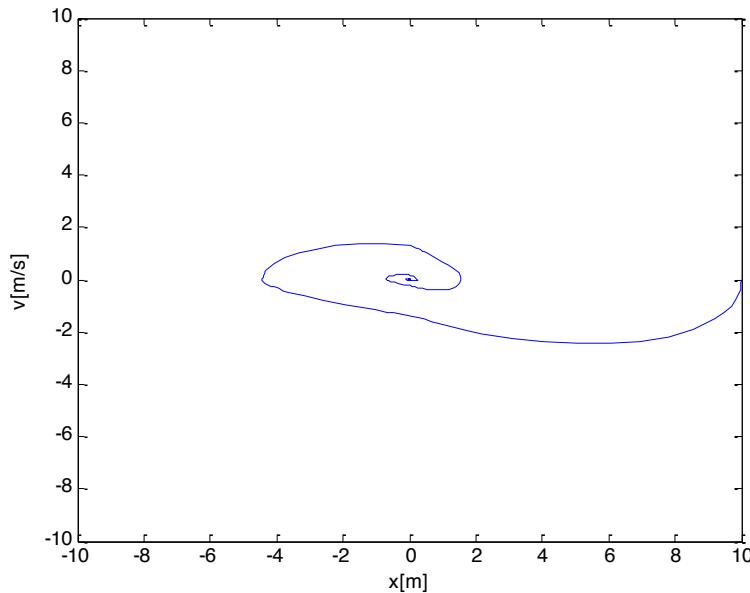


Figure 3.16: Phase plane plot of the closed loop system with the worst case gains

Figure 3.16 shows the simulation result with the worst case system described in (3.26) ~ (3.29). The initial point is $(10, 0)$ and the simulation results shows that the next intersection with the positive x axis is closer to the origin. Since the system is linear at each quadrant, the sequence of intersections with the positive x axis converges to the origin and any initial states will converge to the origin. The original system behaves better than the worst case system at each quadrant, i.e. the solution of the original system is confined by the solution of the worst case system, and the states will converge to the origin faster than the worst case system.

3.6 Disturbance Observer for the Virtual Lead Vehicle

In previous sections, the virtual lead vehicle scheme was introduced and studied as a switching scheme between the distance control and the velocity control modes. Previously, the acceleration of the actual lead vehicle was considered as a disturbance and assumed to be zero which is generally not true. The feedback controller eventually compensated the effect of the nonzero lead vehicle acceleration, however with a time delay. In this section a disturbance observer (DOB) is introduced which can efficiently estimate and compensate for the actual lead vehicle acceleration.

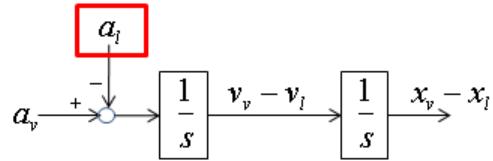


Figure 3.17: Block diagram of the speed and the position errors between the virtual and the actual lead vehicles

3.6.1 Model of the Virtual Lead Vehicle

Figure 3.17 describes the block diagram of the speed and the position errors between the virtual and the actual lead vehicles when an actual lead vehicle exists in a detectable range. The subscript v and l stand for the virtual lead vehicle and the actual lead vehicle, respectively. x , v , and a mean the position, the velocity, and the acceleration, respectively.

In the previous section, the linear quadratic controller calculated the acceleration of the virtual lead vehicle a_v based on the velocity error and the position error. The lead vehicle acceleration a_l was considered as a disturbance and assumed to be zero. Without a disturbance compensation method, nonzero lead vehicle accelerations will cause velocity and position errors, which are eventually compensated by the feedback controller. However, there is a time delay with the compensation. This can be very dangerous particularly if the lead vehicle decelerates very fast. If the lead vehicle and the host vehicle are equipped with communication devices, the control commands from the driver or the cruise control system (if equipped and activated) of the lead vehicle can be transmitted and utilized by the controller of the host vehicle. In general driving situations, it is hard to assume that the lead vehicle is equipped with a communication device compatible with that of the host vehicle. In that case, a DOB constructed for the error model in Figure 3.17 can be utilized to estimate and compensate for the lead vehicle acceleration.

3.6.2 Disturbance Observer

Figure 3.18 describes the block diagram of a typical DOB [37, 55]. $G(z)$ is the transfer function of the actual plant, $G_n(z)$ is the nominal plant transfer function, and r is the relative degree of the plant. Q filter is usually a low pass filter which handles the measurement noise and the model discrepancy. In general, the actual plant dynamics is not precisely known, and the inverse of the nominal plant model is used in DOB.

The actual dynamics of the plant in the present problem is that of a double integrator. Thus, in the present problem, there is no model uncertainty i.e. $G(z) = G_n(z)$. We have two outputs, the velocity error and the position error. When we utilize the position error as the output, noting that the input is the acceleration error, the discrete time plant transfer

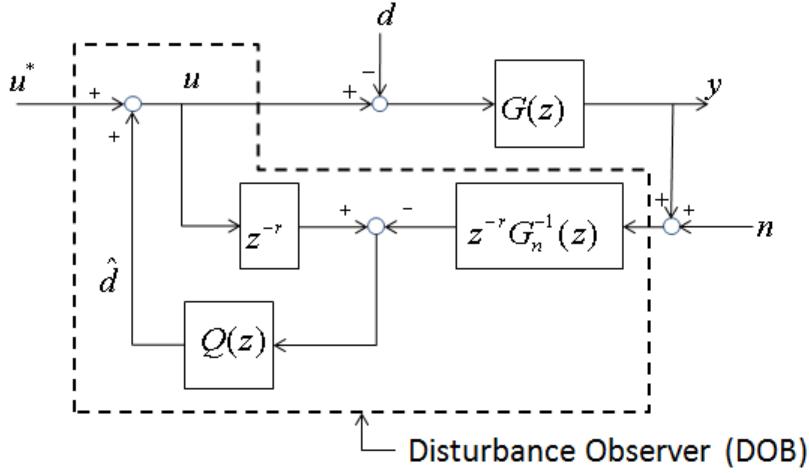


Figure 3.18: Block diagram of the DOB

function is:

$$G_n(z) = \frac{\Delta t^2(z+1)}{2(z-1)^2} \quad (3.30)$$

where Δt is the sampling time. The relative degree r is 1. Furthermore, d and u^* in Figure 3.18 are respectively a_l and $a_{feedback}$, where $a_{feedback}$ is the desired acceleration of the virtual lead vehicle calculated by the feedback controller. With the estimated disturbance, the acceleration of the virtual lead vehicle is:

$$a_v = a_{feedback} + \hat{a}_l \quad (3.31)$$

where \hat{a}_l is the estimated lead vehicle acceleration by DOB. If the measurement noise is neglected, the Q-filter can be set to 1 since there is no model uncertainty.

Figures 3.19 ~ 3.22 compare the simulation results with and without the DOB when there are initial velocity and position errors and the actual lead vehicle accelerates or decelerates. As shown in the figures, the DOB helps the host vehicle react faster to the lead vehicle acceleration and reduces or practically eliminates the overshoot or the undershoot. Since there is no model uncertainty, the closed loop of the DOB is stable with $Q(z) = 1$.

However in real circumstances, the measurement signals are contaminated by noise. With no model uncertainty and $Q(z) = 1$, the transfer function of the closed loop system with the DOB is:

$$G_{a_{feedback} \rightarrow e}(z) = G(z) \quad (3.32)$$

$$G_{a_l \rightarrow e}(z) = (1 - z^{-1})G(z) \quad (3.33)$$

$$G_{n \rightarrow e}(z) = -z^{-1} \quad (3.34)$$

where e is the position error. Figure 3.23 shows the block diagram of the closed loop system.

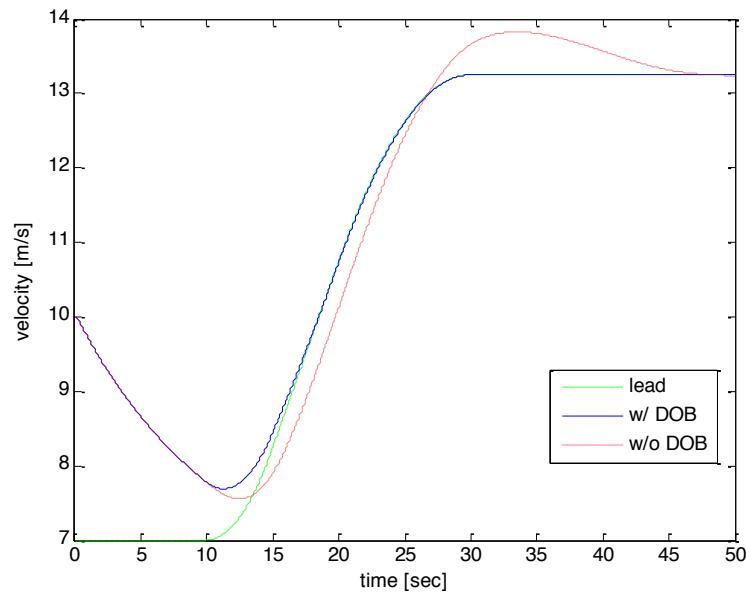


Figure 3.19: Simulation results with and without the DOB when the lead vehicle accelerates: velocity

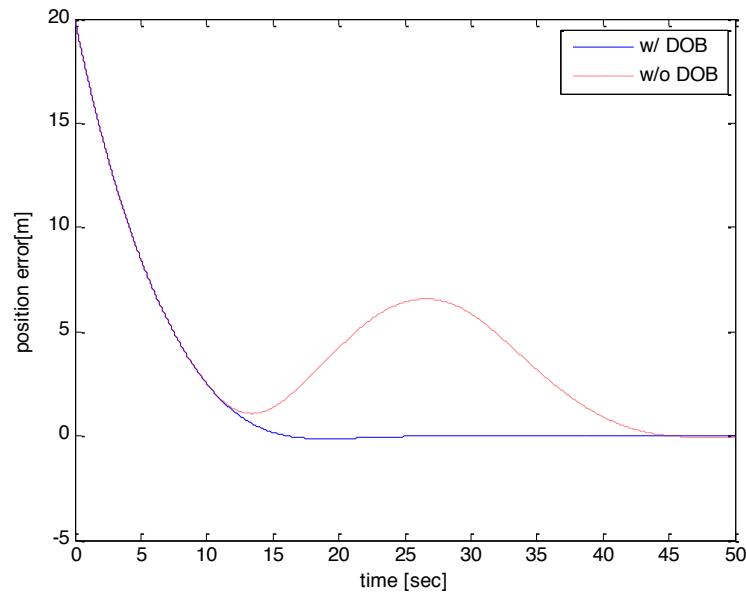


Figure 3.20: Simulation results with and without the DOB when the lead vehicle accelerates: position error

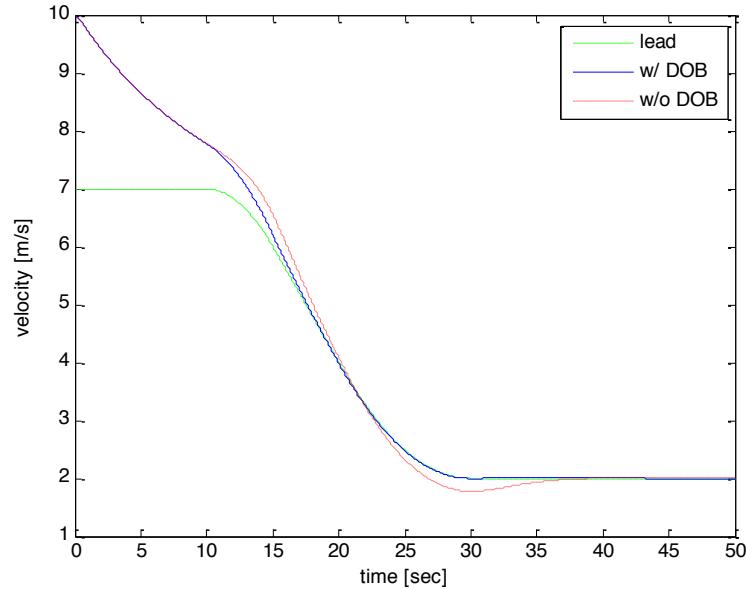


Figure 3.21: Simulation results with and without the DOB when the lead vehicle decelerates: velocity

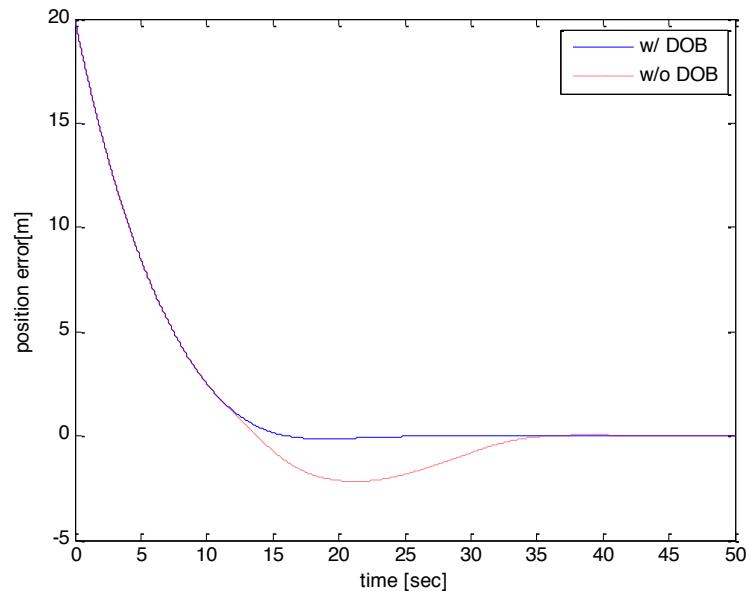


Figure 3.22: Simulation results with and without the DOB when the lead vehicle decelerates: position error

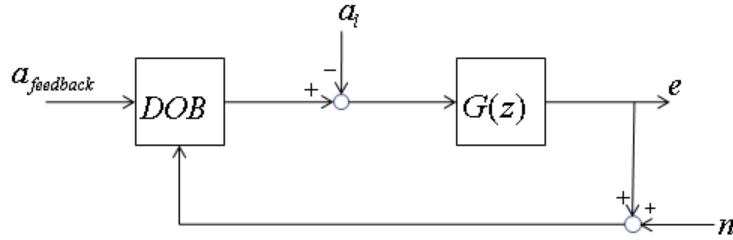


Figure 3.23: Block diagram of the DOB

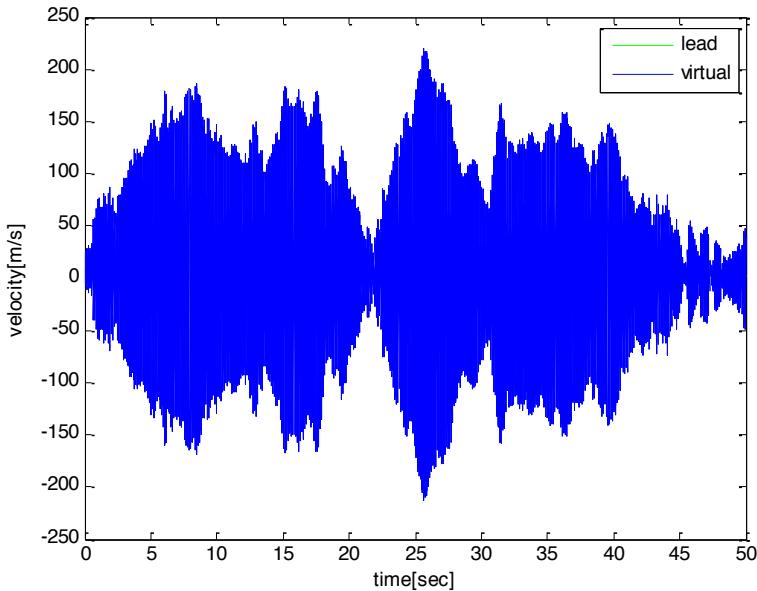


Figure 3.24: Simulation with the measurement noise

(3.32) and (3.33) mean that the transfer function from the control input to the output is the same with the original open loop transfer function and the disturbance is efficiently rejected. These are the typical benefits of the DOB. However (3.34) means that the noise is not rejected at all. As a result, the position of the virtual lead vehicle is contaminated by the measurement noise and the velocity, which is the time derivative of the position, and becomes very noisy.

A simulation study is performed to check the effect of the noise. In the simulation the velocity and the position of the lead vehicle are contaminated by zero-mean white Gaussian noises. It is assumed that the noises of the position and the velocity measurements are independent. As shown in Figure 3.24, the velocity of the virtual lead vehicle is very noisy as expected. This also causes a problem for the feedback controller of the host vehicle. If the feedback controller has a derivative controller, the control signal would be very noisy.

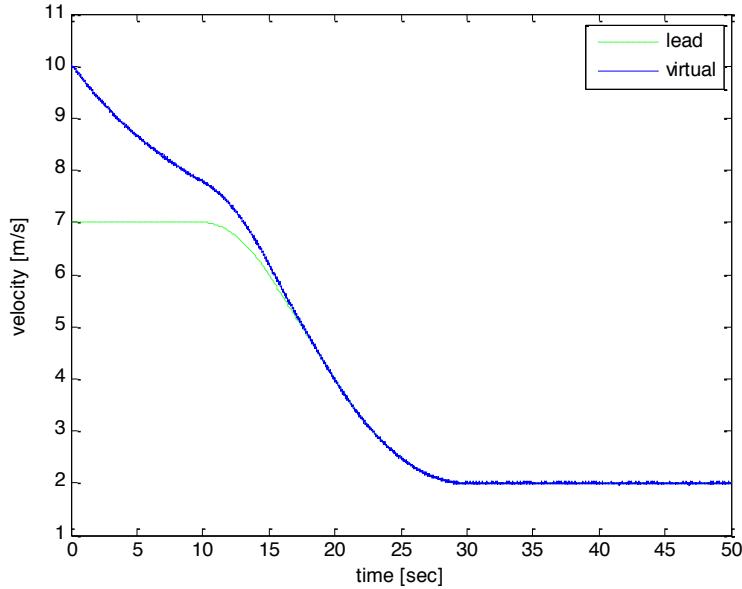


Figure 3.25: Simulation results with measurement noise using the velocity measurement: velocity

To prevent this, the velocity error may be utilized as the output of the system instead of the position error. The transfer function from the acceleration to the velocity is:

$$G_n(z) = \frac{\Delta t}{z - 1} \quad (3.35)$$

Figures 3.25 ~ 3.27 show the simulation results with the plant given by (3.35). As shown in the figures, the position signal of the virtual lead vehicle is very smooth and the velocity of the virtual lead vehicle is a little noisy. Since the feedback controller of the host vehicle utilizes the relative position error, the noise of the velocity signal does not affect much. However, the acceleration of the virtual lead vehicle, which can potentially be utilized as a feedforward control signal, is very noisy as shown in Figure 3.27. To prevent this, the Q-filter is designed to be a low pass filter.

Figures 3.28 and 3.29 show the simulation results with a Q-filter, given by:

$$Q(z) = \left(\frac{0.01}{1 - 0.99z^{-1}} \right)^3 \quad (3.36)$$

The velocity of the virtual lead vehicle is smoother. The acceleration is a little noisy, but compared to the case without Q-filter (Figure 3.27) the noise level has significantly been reduced. For smoother acceleration, the cutoff frequency of the Q filter can be lowered. Doing so, however, slows the response and consequently reduces the performance of the DOB. The Q filter should be tuned based on the properties of the actual measurement noise.

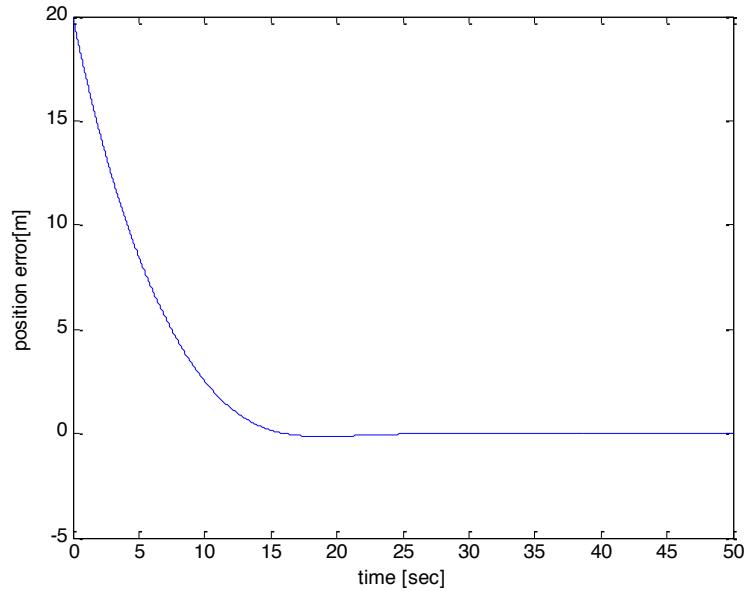


Figure 3.26: Simulation results with measurement noise using the velocity measurement: position error

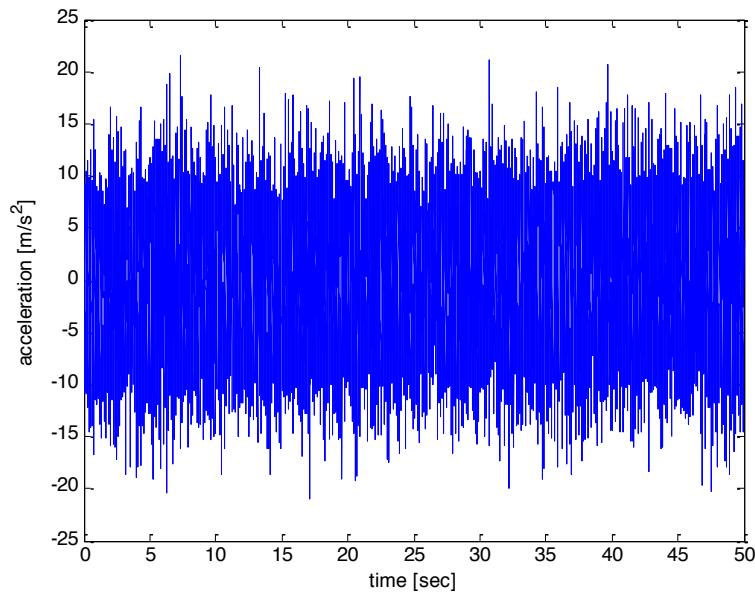


Figure 3.27: Simulation results with measurement noise using the velocity measurement: acceleration of the virtual lead vehicle

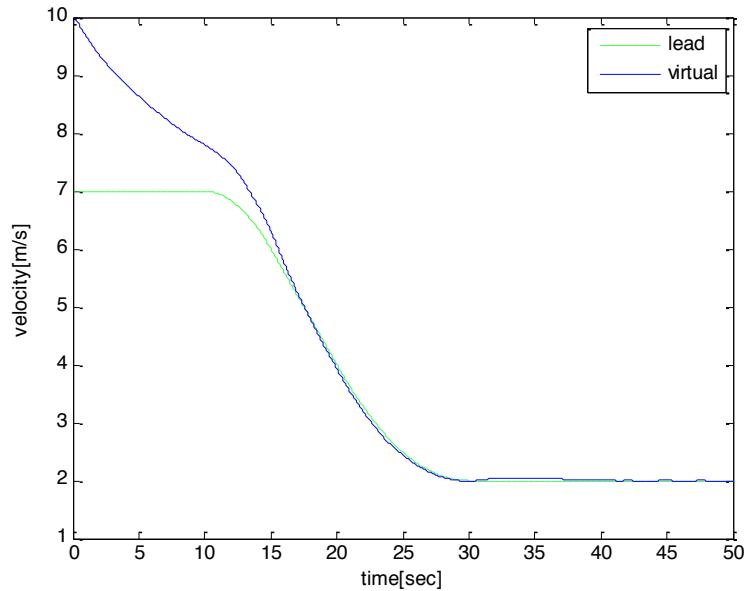


Figure 3.28: Simulation results with a Q-filter: velocity

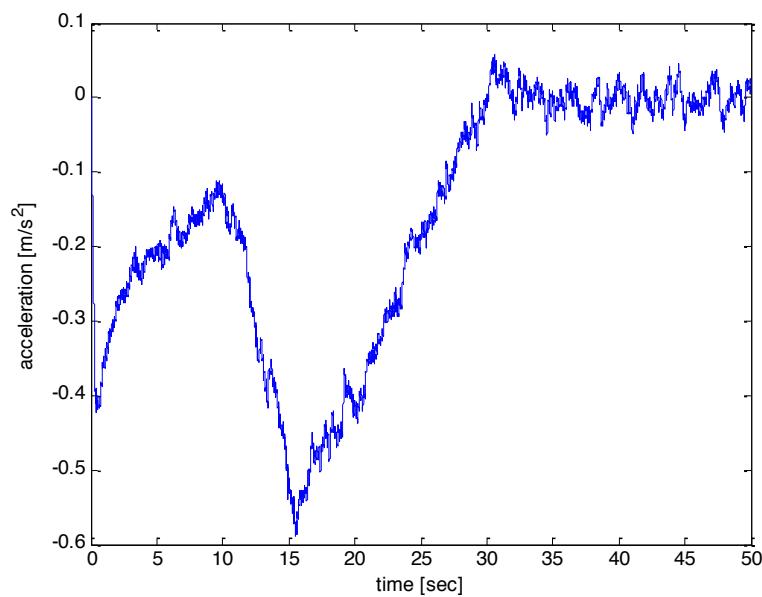


Figure 3.29: Simulation results with a Q-filter: acceleration of the virtual lead vehicle

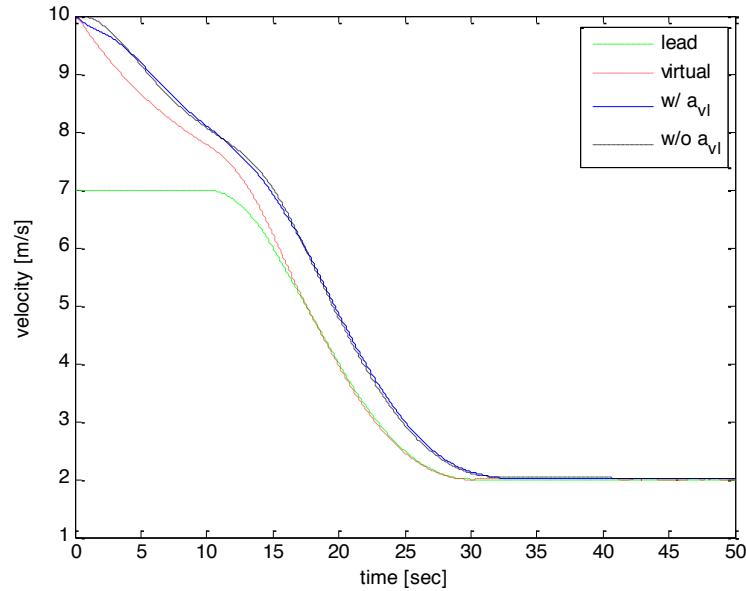


Figure 3.30: Simulation results showing the vehicle behavior with and without the feedforward control when the lead vehicle decelerates: the lead and virtual vehicle's velocity

3.6.3 Utilizing the Lead Vehicle Acceleration

The acceleration of the virtual lead vehicle can be utilized as a feedforward control signal for the host vehicle. Previously, the desired acceleration of the host vehicle was calculated by a feedback controller. The acceleration of the lead vehicle is added to the feedback control signal as:

$$a_{desired} = a_{feedback} + a_{vl} \quad (3.37)$$

where $a_{desired}$, $a_{feedback}$, and a_{vl} are the desired acceleration, the feedback control signal, and the acceleration of the virtual lead vehicle, respectively.

Figures 3.30 ~ 3.33 show the simulation results of the virtual and the host vehicle with and without using the virtual lead vehicle acceleration as a feedforward control signal. As shown in the figures, when the virtual lead vehicle acceleration is utilized as a feedforward signal, the response of the host vehicle is faster and the error is smaller. Another advantage of utilizing the virtual lead vehicle acceleration is that when the lead vehicle decelerates, the position error of the host vehicle remains positive during most of the time. Positive position error means that the relative distance between the host and the lead vehicles is larger than desired and is hence more preferable than the negative errors. In the simulations, it is assumed that the lower level controllers, i.e. the brake controller and the engine controller, make the host vehicle accelerate or decelerate exactly with the desired acceleration.

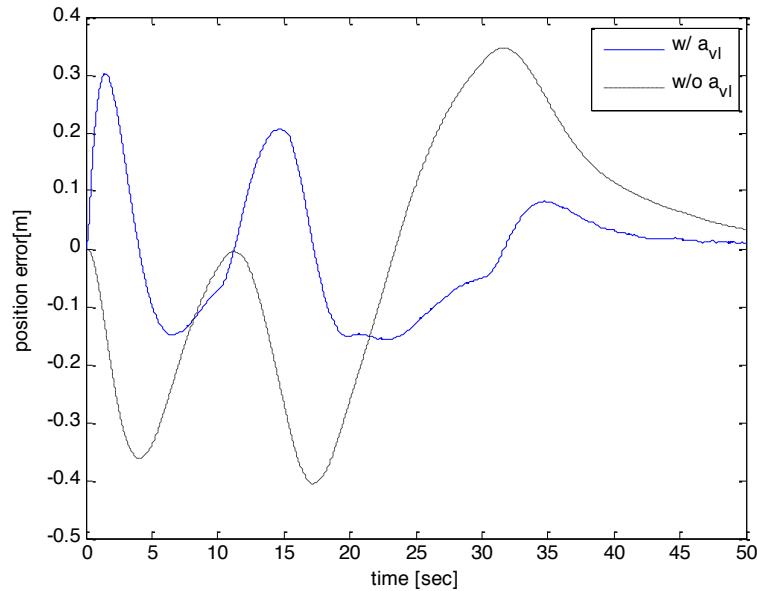


Figure 3.31: Simulation results showing the vehicle behavior with and without the feedforward control when the lead vehicle decelerates: the host vehicle's position

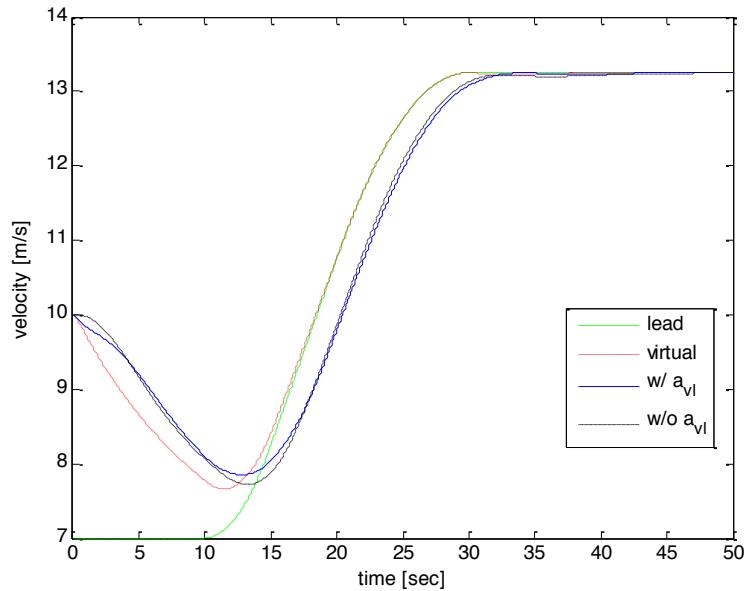


Figure 3.32: Simulation results showing the vehicle behavior with and without the feedforward control when the lead vehicle accelerates: the lead and virtual vehicle's velocity

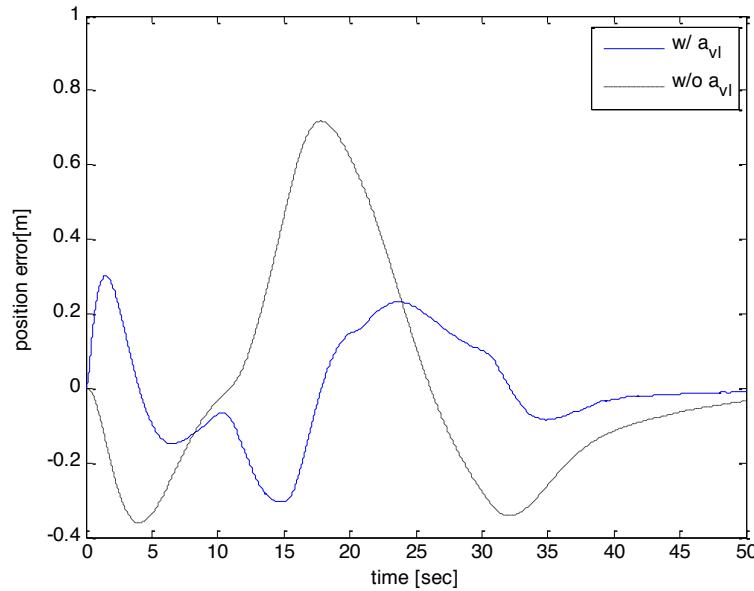


Figure 3.33: Simulation results showing the vehicle behavior with and without the feedforward control when the lead vehicle accelerates: the host vehicle's position

3.6.4 Extended State DOB

In the previous section, only the velocity measurement is utilized to estimate the lead vehicle acceleration. If the position measurement is equally good or better than the velocity measurement in the sense of noise contamination, an estimation method better than DOB may be devised. For this purpose, a Kalman filter with augmented states can be designed. The lead vehicle acceleration is assumed to be continuous and be generated by a stochastic process as:

$$a_l(k+1) = a_l(k) + w(k)\Delta t \quad (3.38)$$

The augmented system model is:

$$\begin{bmatrix} e_x(k+1) \\ e_v(k+1) \\ a_l(k+1) \end{bmatrix} = \begin{bmatrix} 1 & \Delta t & -\frac{1}{2}\Delta t^2 \\ 0 & 1 & -\Delta t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} e_x(k) \\ e_v(k) \\ a_l(k) \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \\ 0 \end{bmatrix} a_v(k) + \begin{bmatrix} 0 \\ 0 \\ \Delta t \end{bmatrix} w(k) \quad (3.39)$$

$$Y(k) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} e_x(k) \\ e_v(k) \\ a_l(k) \end{bmatrix} + \begin{bmatrix} n_x(k) \\ n_v(k) \end{bmatrix} \quad (3.40)$$

where e_x and e_v are the position error and the velocity error, and a_l and a_v are the acceleration of the lead vehicle and the virtual lead vehicle, respectively. w , n_x , and n_v are the zero mean process noise, the position measurement noise, and the velocity measurement

noise, respectively. They are uncorrelated and their covariances are Λ_{ww} , $\Lambda_{n_x n_x}$, and $\Lambda_{n_v n_v}$, respectively.

Assuming that the process noise and the measurement noises are stationary, a steady state Kalman filter can be designed to estimate the augmented states. In DOB, the output of the Q-filter is the estimated disturbance which in this case is the lead vehicle acceleration. In the Extended State DOB(ESDOB) the lead vehicle acceleration is included in the augmented states and estimated by the Kalman filter. Like the DOB, the estimated lead vehicle acceleration is added to the feedback acceleration.

The DOB introduced in the previous sections utilizes only the velocity measurement. The Kalman filter designed above utilizes both the velocity and the distance measurements. It can be expected that the performance of the Kalman filter would be better than the DOB since it uses more information. However such an expectation holds only if the position measurement is as good as or better than the velocity measurement in terms of measurement noise. This does not mean that the covariance of n_x is smaller than that of n_v . Since the velocity is related to the position by a differentiator given by:

$$v(k+1) = \frac{x(k+1) - x(k)}{\Delta t} \quad (3.41)$$

The covariance of the position measurement noise is translated to the velocity measurement noise by:

$$\Lambda_{n_v n_v}^* = \frac{2}{\Delta t^2} \Lambda_{n_x n_x} \quad (3.42)$$

If the covariance of the position measurement is fairly smaller than the velocity measurement noise covariance, i.e. $\Lambda_{n_v n_v}^* < \Lambda_{n_v n_v}$, the augmented Kalman filter should show better performance than DOB.

Table 3.2 compares simulation results of different DOBs with same noise data where the position measurement is better than the velocity measurement in the sense of noise covariances. In the simulations, initial velocity and position errors are zero and the lead vehicle decelerates. The root mean square (RMS) errors of the position and the velocity are used to compare the performance and the RMS jerk of the estimated lead vehicle acceleration is used to measure how much the control signals are noisy. Two different Q-filters are used to compare both the performance and the noise cancellation.

$$Q_1(z) = \left(\frac{0.01}{1 - 0.99z^{-1}} \right)^2 \quad (3.43)$$

$$Q_2(z) = \left(\frac{0.01}{1 - 0.99z^{-1}} \right)^3 \quad (3.44)$$

The augmented Kalman filter is given by:

$$\tilde{Y}^o(k) = Y(k) - C \hat{X}^o(k) \quad (3.45)$$

	RMS Position Error [m]	RMS Velocity Error [m/s]	RMS Jerk [m/s ³]
DOB with Q_1	0.0910	0.0252	2.4855
DOB with Q_2	0.1316	0.0345	0.9976
ESDOB	0.0760	0.0203	0.9911

Table 3.2: Simulation results comparing the DOB and the Extended State DOB(ESDOB)

$$\hat{X}(k) = \hat{X}^o(k) + MC^T[CMC^T + V]^{-1}\tilde{Y}^o(k) \quad (3.46)$$

$$\hat{X}^o(k+1) = A\hat{X}(k) + Ba_v(k) \quad (3.47)$$

where X is the augmented state, i.e. $X(k) = [e_x(k) \ e_v(k) \ a_l(k)]^T$ and M is the stationary a-priori state estimation covariance which is the solution of:

$$AMA^T - AMC^T[CMC^T + V]^{-1}CMA^T - M + B_wWB_w^T = 0 \quad (3.48)$$

A , B , C , and B_w are from the system model:

$$A = \begin{bmatrix} 1 & \Delta t & -\frac{1}{2}\Delta t^2 \\ 0 & 1 & -\Delta t \\ 0 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \\ 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B_w = \begin{bmatrix} 0 \\ 0 \\ \Delta t \end{bmatrix} \quad (3.49)$$

The input noise covariance and the measurement noise covariance are:

$$W = \Lambda_{ww}, V = \begin{bmatrix} \Lambda_{n_x n_x} & 0 \\ 0 & \Lambda_{n_v n_v} \end{bmatrix} \quad (3.50)$$

This Kalman filter is called the extended state DOB in this thesis.

As shown in the simulation results, the performance of extended state DOB is better than the typical DOB for both the disturbance compensation and the noise cancellation. However, the extended state DOB is useful only when the position measurement is minimally contaminated by noise. Note that with a short sampling time, $\Lambda_{n_x n_x}$ must be very small for $\Lambda_{n_v n_v}^* < \Lambda_{n_v n_v}$ to be hold. For further progress and implementation, more information about the measurement sensor noise is required to check the effectiveness of the extended state DOB.

3.7 Summary

In this chapter, the virtual lead vehicle scheme was introduced. Instead of following the actual lead vehicle, the host vehicle followed the virtual lead vehicle generated by the ACC system. When there was a lead vehicle, the speed and the position of the virtual lead vehicle were set to be the same with the speed and the position of the actual lead vehicle. When no lead vehicle was detected, the speed of the virtual lead vehicle was set to be the desired

speed and the switching between the distance control and the velocity control modes was prevented.

Also the virtual lead vehicle scheme was utilized for smooth transient during cutting in/out of a lead vehicle. When the lead vehicle cut out or a new lead vehicle cut in from a side lane, the virtual lead vehicle merged smoothly to the new lead vehicle and the host vehicle followed the virtual lead vehicle and moved smoothly. A linear quadratic controller with variable weights was designed to control the virtual vehicle in transient. With the variable weights, the virtual vehicle was controlled to move smoothly when it was a safe situation and react faster when it was a dangerous situation. The stability of the linear quadratic control with variable weights was proved by a phase portrait method.

In the transient motion, the virtual lead vehicle was modeled as a double integrator and the lead vehicle acceleration was assumed to be zero which is not always true. A disturbance observer (DOB) was utilized to estimate and compensate the actual lead vehicle acceleration. When the lead vehicle acceleration was estimated and compensated by adding a feedforward control input to the virtual lead vehicle, the virtual lead vehicle showed faster and safer performance. Extended state disturbance observer was studied for the case when both the position and the velocity measurements were utilized in the DOB. Extended state DOB showed better performance than the original DOB when the position measurement noise covariance was smaller than the equivalent velocity measurement noise covariance.

Chapter 4

Optimal Profile Generation for Stop-and-Go

4.1 Introduction

While the conventional ACC is intended to assist the driver at relatively higher speeds, the ACC system with Stop-and-Go is intended to be used at low speeds and should have a capability to stop the vehicle completely if there is a stopped lead vehicle [38]. Since it is expected to be utilizable at low speed in urban driving situations, the ACC system with Stop-and-Go should be able to properly react to frequent cutting in and out of lead vehicles and this feature is studied with the virtual lead vehicle scheme in 3. Another necessary feature is the complete stop.

Various research have been done about the ACC with Stop-and-Go. Fuzzy control algorithms were used in [11] and [32]. Linear quadratic optimal controller with mode switching between the speed control algorithm and the distance control algorithm was used in [63, 64]. Sliding controller was utilized in [1] and [6]. Various feedback control laws based on the velocities of the lead and the host vehicles and the relative distance were also used [28, 41]. Explicit Model Predictive Control was also applied to ACC with Stop-and-Go in [34]. In most of the previous research, the ACC with Stop-and-Go systems were designed and verified that they were utilizable in certain speed range including low speeds. However, the passenger comfort was not considered during a complete stop.

As studied in Section 2.4, the desired relative distance to the lead vehicle is calculated from the time-headway and the safety distance. i.e. $d_{desired} = t_{hw}v_{host} + d_{offset}$. With the constant time-headway, the string stability is guaranteed. However, it can cause a problem when the lead vehicle is stopped completely. When the lead vehicle is stopped completely, the host vehicle should be stopped d_{offset} behind the lead vehicle. However at the moment when the lead vehicle is stopped, the desired distance, which is proportional to the velocity of the host vehicle, is larger than d_{offset} and the controller tries to maintain larger distance to the lead vehicle than it actually should. Figure 4.1 shows the simulation result for a

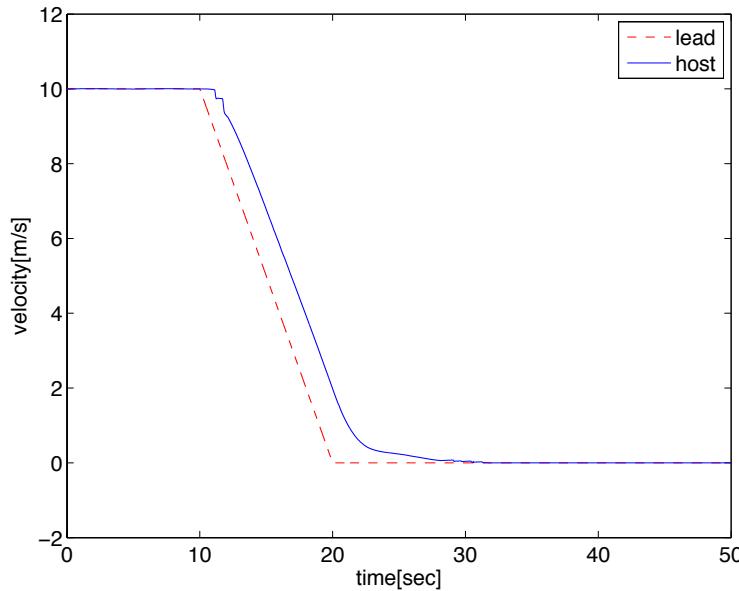


Figure 4.1: Velocity of the lead and the host vehicle in case of a complete stop

complete stop when a PID controller is used and the desired relative distance is determined by the constant time-headway. As shown in the figure, when the lead vehicle is stopped, initially the host vehicle still decelerates very fast and maintains very slow velocity until it stops completely. It takes more than 10 seconds for the host vehicle to completely stop after the lead vehicle has stopped. This motion can make the driver feel uncomfortable and can cause more wear on brakes.

To prevent this unnecessary motion when the lead vehicle completely stops, an optimal acceleration profile should be calculated and the PID feedback controller should be replaced by an alternative controller which has a better tracking capability. Sliding controller is a possible candidate. However, the optimization problem should be solved in real time to calculate the optimal profile for the sliding controller. Fortunately, the optimization problem formulated in this chapter is a quadratic programming problem and can be converted to a linear complementarity problem. The linear complementarity problem can be solved fast with the Lemke algorithm if the dimension of the problem is small. In this thesis, a multi-resolution formulation is suggested to reduce the dimension of the problem without changing the fundamental characteristic of the problem and to solve it very fast.

This chapter is organized as follows. Optimization problems are formulated to find the optimal relative distance profile during a complete stop and the optimal velocity profile during a starting motion in Section 4.2. In Section 4.3, a sliding mode controller is designed for a complete stop and the virtual lead vehicle scheme is also utilized for the starting motion. In Section 4.4, algorithms solving the optimization problem are studied and multi-resolution



Figure 4.2: Sketch of the complete stop

formulation is suggested to solve the problem in real time. Section 4.5 summarizes the chapter.

4.2 Optimal Relative Distance Profile during a Complete Stop

4.2.1 Optimization Problem Formulation

To find the optimal trajectory, an optimization problem is formulated and solved. In this section, a linear model is used to formulate the problem into a Quadratic Programming (QP) optimization problem.

The complete stop scenario is described in Figure 4.2. The lead vehicle is completely stopped in front of the host vehicle at position x_{lead} . d_{offset} is the desired relative distance to the lead vehicle when the host vehicle is completely stopped behind the lead vehicle. The relation between the position, the velocity, and the acceleration of the host vehicle is a double integrator given by:

$$v(t) = v(0) + \int_0^t a(\tau) d\tau \quad (4.1)$$

$$x(t) = x(0) + \int_0^t v(\tau) d\tau \quad (4.2)$$

where $x(t)$, $v(t)$ and $a(t)$ are the position, the velocity, and the acceleration at time t , respectively. Using (4.1) and (4.2), however, makes the problem infinite dimensional and generally hard to solve. To get the discrete time formulation, it is assumed that the acceleration is constant for each time step Δt . Then the double integrator can be expressed as:

$$\begin{bmatrix} x_{k+1} \\ v_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ v_k \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix} a_k \quad (4.3)$$

where x_k and v_k are the position and the velocity at each time, i.e. $x_k = x(k\Delta t)$, and the a_k is the constant acceleration during the time step, i.e. $a(t) = a_k$ for $(k-1)\Delta t \leq t < k\Delta t$.

The time horizon should be fixed as a constant $T_f < \infty$ to make the problem finite dimensional and to find the solution such that the host vehicle stops in finite time after a reasonable time duration. The initial conditions $x(0)$ and $v(0)$ are assumed to be known, and the final conditions are $x(T_f) = x_{lead} - d_{offset}$, $v(T_f) = 0$, and $a(T_f) = 0$. The cost which is minimized in the optimization problem should penalize not only the stopping time but also the acceleration and jerk to improve the passenger comfort. There should also be maximum and minimum limits for the acceleration and the jerk. One simple cost function choice to make the optimization problem easily solvable is to use a quadratic cost function that penalizes distance error, acceleration, and jerk given by:

$$J(x, a, j) = \lambda_x \|x - x_l\|_2^2 + \lambda_a \|a\|_2^2 + \lambda_j \|j\|_2^2 \quad (4.4)$$

where $x - x_l = [x_1 - x_l \ x_2 - x_l \ \cdots \ x_N - x_l]^T$, $x_l = x_{lead} - d_{offset}$, $a = [a_1 \ a_2 \ \cdots \ a_N]^T$, and $j = [j_1 \ j_2 \ \cdots \ j_{N-1}]^T$. j_k is the jerk which is given as $j_k = (a_{k+1} - a_k)/\Delta t$ and $N = T_f/\Delta t$. Since the jerk can be represented as a linear function of the acceleration, the cost function can be represented only as a function of the acceleration by:

$$J(x, a) = \lambda_x \|x - x_l\|_2^2 + a^T (\lambda_a I_{N \times N} + \lambda_j J^T J) a \quad (4.5)$$

where

$$J = \frac{1}{\Delta t} \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -1 & 1 \end{bmatrix} \quad (4.6)$$

Since, the cost function in (4.5) is quadratic in terms of the acceleration and is strictly convex, it can be solved in finite time[7]. The algorithm will be reviewed in Section 4.4. To reduce the time to completely stop, $x - x_l$ should converge to zero quickly. To enforce it, l^0 norm can be introduced. The l^0 norm of a vector is defined as the number of nonzero elements of it. If the l^0 norm of a vector is penalized in the optimization problem, the solution becomes sparse i.e. many elements are zero. However, the l^0 norm of a vector is not convex and it is not guaranteed that the optimization problem is solvable. Instead the l^1 norm which is a convex relaxation of the l^0 norm can be used. The l^1 norm of a vector is defined as:

$$\|s\|_1 = \sum_{i=1}^N |s_i| \quad (4.7)$$

By penalizing the l^1 norm of the relative distance in the cost function, the relative distance of the solution becomes sparse. Since the relative distance should always be larger than or equal to zero, penalizing the l^1 norm of the relative distance works as a time penalization. The modified cost function is:

$$J(x, a) = a^T (\lambda_a I_{N \times N} + \lambda_j J^T J) a + \lambda_x \|x - x_l\|_1 \quad (4.8)$$

Considering the passenger comfort, the inequality constraints for the acceleration and the jerk are introduced. Also, to prevent unreasonable motions, i.e. moving backward, the inequality constraints for the velocity are introduced. Also, the linear model equation and the initial conditions are added as equality constraints.

$$x_i \leq x_l \quad (4.9)$$

$$v_i \geq 0 \quad (4.10)$$

$$a_{min} \leq a_i \leq a_{max} \quad (4.11)$$

$$j_{min} \leq j_i \leq j_{max} \quad (4.12)$$

$$\begin{bmatrix} x_{k+1} \\ v_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ v_k \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix} a_k \quad (4.13)$$

$$\begin{bmatrix} x_1 \\ v_1 \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ v_0 \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix} a_0 \quad (4.14)$$

where x_0 , v_0 , and a_0 are the initial position, velocity, and acceleration of the host vehicle.

The optimization problem is given with (4.8) and (4.9) ~ (4.14). However it can be simplified to be solved efficiently. The problem has $3N$ variables, i.e. $x \in R^N$, $v \in R^N$, and $a \in R^N$. However (4.8) is only a function of the acceleration while the position and the velocity are related through (4.13) and (4.14). The position vector and the velocity vector can be expressed as an affine function of the acceleration vector as (4.15) and (4.16) to reduce the problem dimension.

$$x = \begin{bmatrix} x_0 + v_0\Delta t \\ x_0 + 2v_0\Delta t \\ x_0 + 3v_0\Delta t \\ \vdots \\ x_0 + Nv_0\Delta t \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2 & 0 & 0 & \cdots & 0 \\ \frac{3}{2}\Delta t^2 & \frac{1}{2}\Delta t^2 & 0 & \ddots & \vdots \\ \frac{5}{2}\Delta t^2 & \frac{3}{2}\Delta t^2 & \frac{1}{2}\Delta t^2 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \frac{2N-1}{2}\Delta t^2 & \frac{2N-3}{2}\Delta t^2 & \cdots & \frac{3}{2}\Delta t^2 & \frac{1}{2}\Delta t^2 \end{bmatrix} a \triangleq X_0 + Xa \quad (4.15)$$

$$v = \begin{bmatrix} v_0 \\ v_0 \\ \vdots \\ v_0 \end{bmatrix} + \begin{bmatrix} \Delta t & 0 & \cdots & 0 \\ \Delta t & \Delta t & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ \Delta t & \Delta t & \cdots & \Delta t \end{bmatrix} a \triangleq V_0 + Va \quad (4.16)$$

From (4.9), $\|x - x_l\|_1$ in the cost function can be represented as $-1_N^T(x - x_l)$ where $1_N = [1 \ \cdots \ 1]^T \in R^{N \times 1}$. Also from (4.10), the velocity is always greater than or equal to zero and hence, (4.9) can be reduced to a single constraint $x_N - x_l \leq 0$. Finally, the

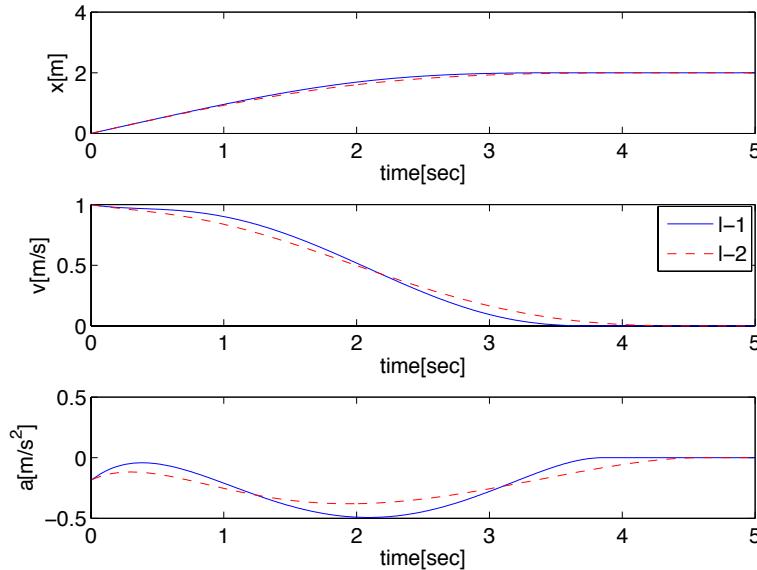


Figure 4.3: Solution of the QP comparing $l - 1$ and $l - 2$ norms, $\lambda_x = 1, \lambda_a = 1, \lambda_j = 0.5$

problem can be expressed as:

$$\begin{aligned}
 & \underset{a \in R^N}{\text{minimize}} \quad a^T (\lambda_a I_{N \times N} + \lambda_j J^T J) a - \lambda_x 1_N^T X a \\
 & \text{subject to} \quad (X_0 + Xa)_N \leq x_l \\
 & \quad (V_0 + Va)_i \geq 0 \\
 & \quad a_{min} \leq a_i \leq a_{max} \\
 & \quad j_{min} \leq (Ja)_i \leq j_{max}
 \end{aligned} \tag{4.17}$$

The final problem is in a form of a linear constrained Quadratic Program (QP).

Figure 4.3 shows the solutions of the QP with $l - 1$ and $l - 2$ norms of the $x - x_l$. As shown in the figure, with $l - 1$ norm, the velocity converges to zero faster. This comes from the sparsity of the $\lambda_x \|x - x_l\|_1$. Figure 4.4 shows the solutions with different λ_x 's. With very larger λ_x , the solution is very similar to the bang-bang control which only utilizes the maximum or the minimum values of the acceleration and the jerk. With larger λ_x or smaller λ_a and λ_j , the trajectory is similar to aggressive driving while with smaller λ_x or larger λ_a and λ_j , it is more like conservative driving. These weights can be tuned based on the preference of the driver.

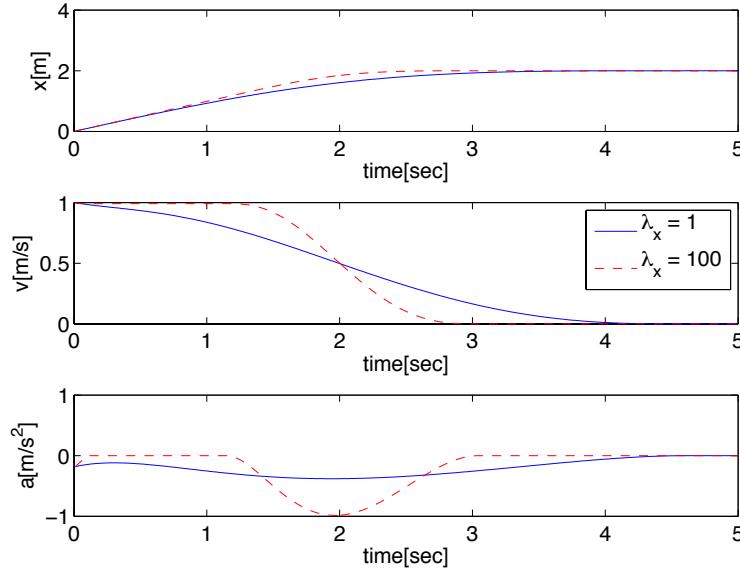


Figure 4.4: Solution of the QP comparing $\lambda_x = 1$ and $\lambda_x = 100, \lambda_a = 1, \lambda_j = 0.5$

4.2.2 Velocity Profile of a Starting Motion

Once the vehicle completely stops, when the lead vehicle resumes its motion the host vehicle should also resume its motion and follow the lead vehicle. However, due to safety and comfort issue, it is not preferable to directly follow the lead vehicle. Instead, an optimal velocity profile can be designed and the host vehicle or the virtual lead vehicle can start their motion with the optimal profile. In this case the acceleration, the jerk, and the velocity error should be considered in the cost function:

$$\begin{aligned}
 & \underset{a \in R^N}{\text{minimize}} \quad \lambda_a \|a\|_2^2 + \lambda_j \|j\|_2^2 + \lambda_v \|Va - v_f\|_1 \\
 & \text{subject to} \quad (Va)_i \leq v_f \\
 & \quad a_{min} \leq a_i \leq a_{max} \\
 & \quad j_{min} \leq (Ja)_i \leq j_{max}
 \end{aligned} \tag{4.18}$$

where v_f is the desired final velocity and V is the matrix defined in (4.16).

Figure 4.5 shows the simulation result with various λ_j 's. The initial velocity, the initial acceleration, and the desired final velocity are $0[m/sec]$, $0[m/sec^2]$, and $5[m/sec]$, respectively. The maximum acceleration is given by $1[m/sec^2]$, and the maximum magnitude of the jerk is $2[m/sec^3]$. The final time is 10 sec. As shown in Figure 4.5 with larger λ_j , the jerk is penalized more and as a result the acceleration profile becomes smoother. However, it will make the velocity converge to its final value a little slower.

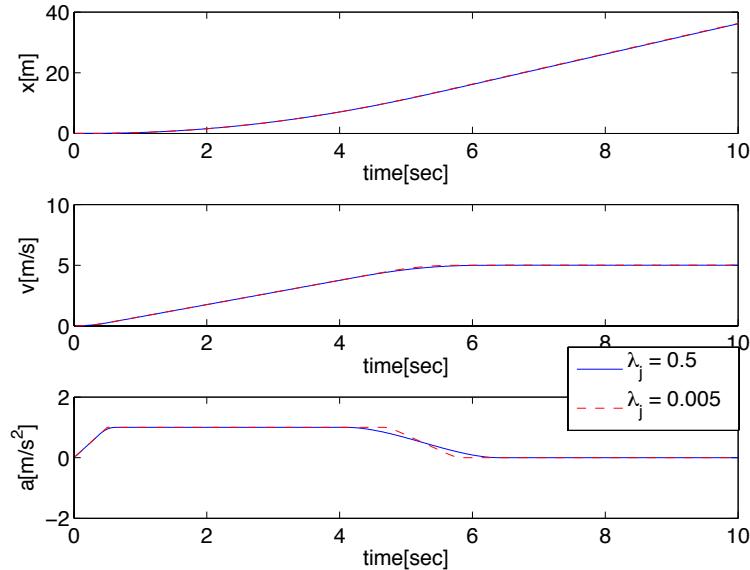


Figure 4.5: Optimal profile generation result for starting motion

4.3 Sliding Control and Virtual Lead Vehicle Scheme

In Section 4.2, the optimal relative distance profile during a complete stop and the optimal velocity profile during a starting motion were designed. The design problem to follow is tracking the desired optimal profiles, for which sliding mode control is promising. Sliding mode control is a tracking control method which guarantees a convergence to the desired states in a finite time [48]. In this section, a sliding mode controller for velocity tracking is designed and verified by simulations. To remove a high-frequency oscillation due to the discontinuity of sliding mode controller, a smooth sliding mode controller is designed. Simulation results show that the smooth sliding mode controller shows a good tracking performance. When the vehicle starts moving after a complete stop, the optimal velocity profile can be utilized with the sliding controller. However, in many cases it is expected that there is an actual lead vehicle moving slower than the desired speed. To react this lead vehicle, instead of the sliding controller, a virtual lead vehicle utilizes the optimal velocity and acceleration.

4.3.1 Sliding Controller Design for a Complete Stop

To design a sliding mode controller, the dynamics of the system should be examined. For simplicity, it is assumed that only the brake is utilized during the stopping motion, and the dynamics of the brake is first order:

$$G_{brake}(s) = \frac{1}{\tau_b s + 1} \quad (4.19)$$

where τ_b is the time constant of the brake. The state space equations of the system are given by:

$$x_1 = d_{relative} = d_0 - x \quad (4.20)$$

$$\dot{x}_1 = x_2 = -\dot{x} = -v \quad (4.21)$$

$$\dot{x}_2 = x_3 = -\ddot{x} = -a \quad (4.22)$$

$$\dot{x}_3 = x_4 = -\ddot{x} = -\frac{1}{\tau_b}u + \frac{1}{\tau_b}a \quad (4.23)$$

where d_0 is the relative distance to the stopped lead vehicle, x is the position of the host vehicle from the initial position, v is the velocity of the host vehicle, a is the acceleration of the host vehicle, and u is the control input. The objective of the sliding control is to make the x_1 track the previously designed desired trajectory. The relative degree of the system is three, and we can define a sliding surface as:

$$S = \left(\frac{d}{dt} + \lambda \right)^2 (\dot{x}_1 - \dot{d}_d) \quad (4.24)$$

where d_d is the desired relative distance trajectory. The sliding condition is represented by:

$$S\dot{S} \leq -\eta|S| \quad (4.25)$$

To satisfy this condition, the time derivative of the sliding surface can be chosen as:

$$\dot{S} = -\eta sgn(S) \quad (4.26)$$

Since, \dot{S} is a function of the control input u , the control input is determined from (4.26) as:

$$u = \tau_b \left(-\ddot{d}_d + \frac{1}{\tau_b}a + 2\lambda(-\ddot{d}_d - a) + \lambda^2(-\dot{d}_d - v) + \eta sgn(S) \right) \quad (4.27)$$

(4.27) gives the sliding control law.

4.3.2 Simulation Results for a Complete stop

Figures 4.6 and 4.7 show the simulation result of the sliding mode controller with $\lambda = 0.1$ and $\eta = 1$. As shown in Figure 4.6, the sliding mode controller let the relative distance follow the desired trajectory very accurately. However, there is an undesirable oscillation in the acceleration and the control input as shown in Figure 4.7. Also the relative distance error does not converge exactly to zero but keeps oscillating near zero and that makes the control signal nonzero as shown in Figure 4.7. This is due to the sign function in (4.27) which introduces discontinuity in the control input.

To prevent the undesired oscillation, a smooth sliding controller can be introduced [48]. Instead of the sign function, a saturation function is used:

$$u = \tau_b \left(-\ddot{d}_d + \frac{1}{\tau_b}a + 2\lambda(-\ddot{d}_d - a) + \lambda^2(-\dot{d}_d - v) + \eta sat\left(\frac{S}{\phi}\right) \right) \quad (4.28)$$

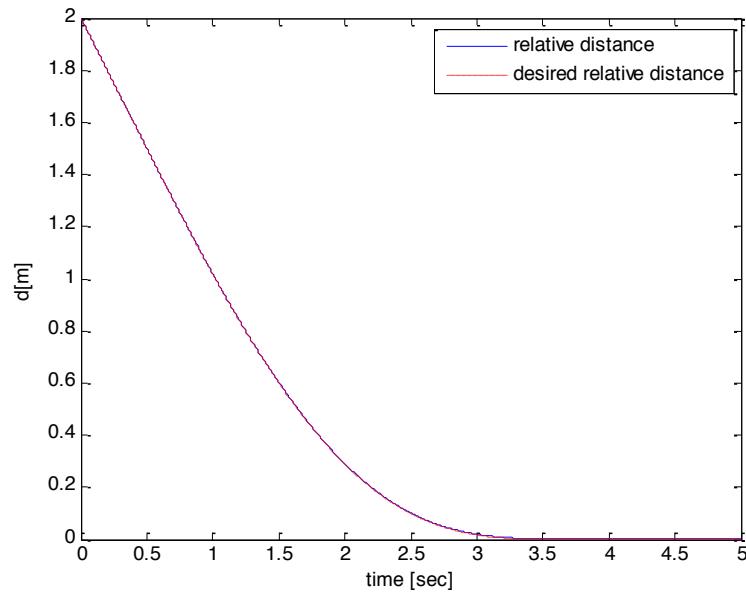


Figure 4.6: Simulation results with the sliding mode controller: relative distance

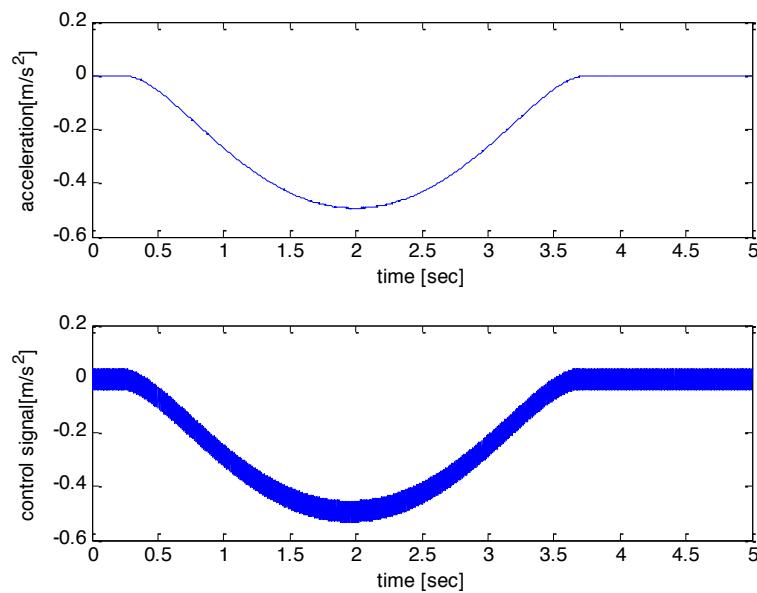


Figure 4.7: Simulation results with the sliding mode controller: acceleration and control command

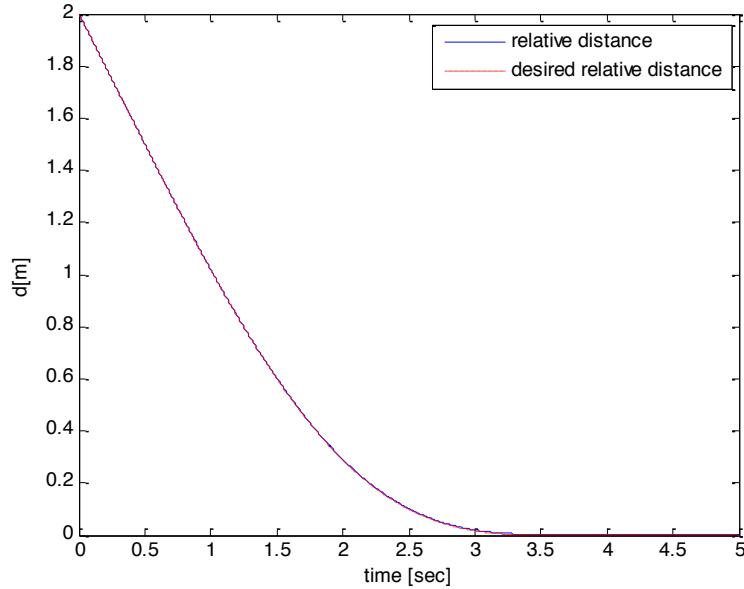


Figure 4.8: Simulation results with the smooth sliding mode controller: relative distance

where the saturation function is:

$$sat(x) = \begin{cases} 1 & x \geq 1 \\ x & -1 < x < 1 \\ -1 & x \leq -1 \end{cases} \quad (4.29)$$

ϕ in (4.28) defines the boundary layer thickness. Large ϕ reduces the oscillation but worsens the guaranteed precision [48]. In the simulation $\phi = 0.1$ is used.

Figures 4.8 and 4.9 show the result with the smooth sliding control law. The tracking performance is almost the same as the sliding mode controller with the sign function, while there is no undesirable oscillation.

4.3.3 Virtual Lead Vehicle for a Starting Motion

Sliding Mode Controller for Starting Motion

The sliding mode control scheme can be applied for a starting motion. The velocity profile is designed by solving an optimization problem introduced in Section 4.2.2. For the starting motion, it is assumed that only the engine is utilized and the dynamics of the engine is first order:

$$G_{engine}(s) = \frac{1}{\tau_e s + 1} \quad (4.30)$$

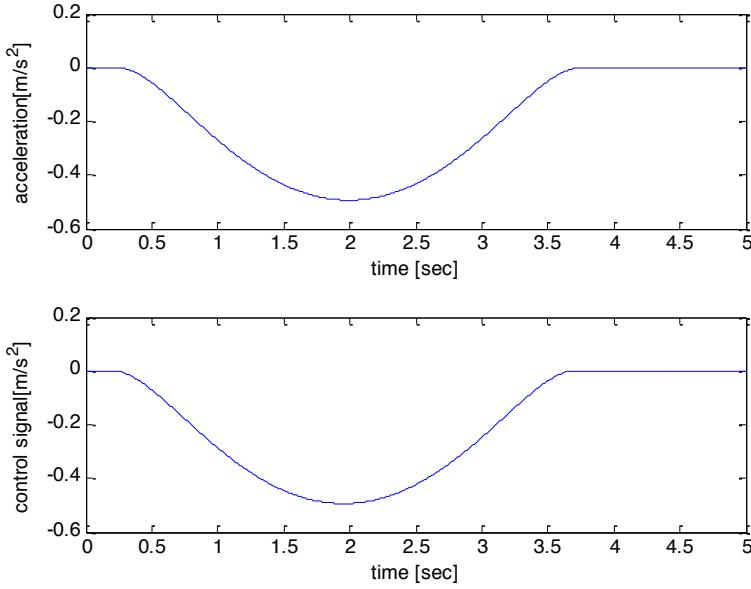


Figure 4.9: Simulation results with the smooth sliding mode controller: acceleration and control command

where τ_e is the time constant of the engine. The state space equations of the system are given by:

$$x_1 = v \quad (4.31)$$

$$\dot{x}_1 = x_2 = a \quad (4.32)$$

$$\dot{x}_2 = x_3 = \dot{a} = \frac{1}{\tau_e}u - \frac{1}{\tau_e}a \quad (4.33)$$

The relative degree of the system is two, and we can define a sliding surface as:

$$S = \left(\frac{d}{dt} + \lambda \right) (\dot{x}_1 - \dot{v}_d) \quad (4.34)$$

where v_d is the desired velocity. The sliding surface can be chosen as:

$$\dot{S} = -\eta sgn(S) \quad (4.35)$$

To prevent the undesired oscillation, the smooth sliding control input is:

$$u = \tau_e \left(\ddot{v}_d + \frac{1}{\tau_e}a + \lambda(\dot{v}_d - a) - \eta \text{sat} \left(\frac{S}{\phi} \right) \right) \quad (4.36)$$

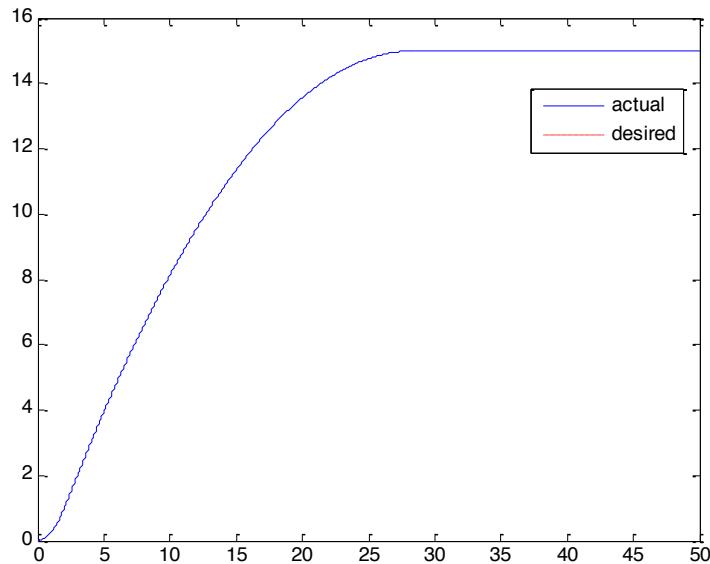


Figure 4.10: Simulation result for a starting motion with a sliding mode controller

Simulation Results and Virtual Lead Vehicle Implementation for a Starting Motion

Figure 4.10 shows the simulation result with the sliding mode controller. As shown in the figure, the host vehicle follows the desired velocity profile very accurately. In this simulation, the desired final velocity is 15[m/sec]. However, in the actual driving situation, the velocity of the lead vehicle can be slower than the desired velocity and in that case if the lead vehicle is close to the host vehicle, the host vehicle should follow the lead vehicle rather than tracking the desired velocity profile. One possible solution is introducing a switch which utilizes the sliding controller when the lead vehicle is far ahead and uses the PID controller when the lead vehicle is close. However, this method requires an additional switch and makes the structure of the system more complicated. Instead of using a switch, the virtual lead vehicle scheme unnecessitates such a switch.

Figure 4.11 shows the simulation result with a virtual lead vehicle driven at the optimal velocity profile. As shown in the figure, the host vehicle follows the virtual lead vehicle and the velocity, the acceleration, and the jerk profiles of the host vehicle are similar to the optimal profiles. The virtual lead vehicle scheme has its benefit especially when there is an actual lead vehicle driven slowly.

Figures 4.12 and 4.13 show the simulation result of the virtual lead vehicle scheme when there is a slower lead vehicle. As shown in the figure, initially, the virtual lead vehicle accelerates according to the optimal velocity profile. However, when the actual lead vehicle is detected the virtual lead vehicle decelerates and follows the actual lead vehicle. The

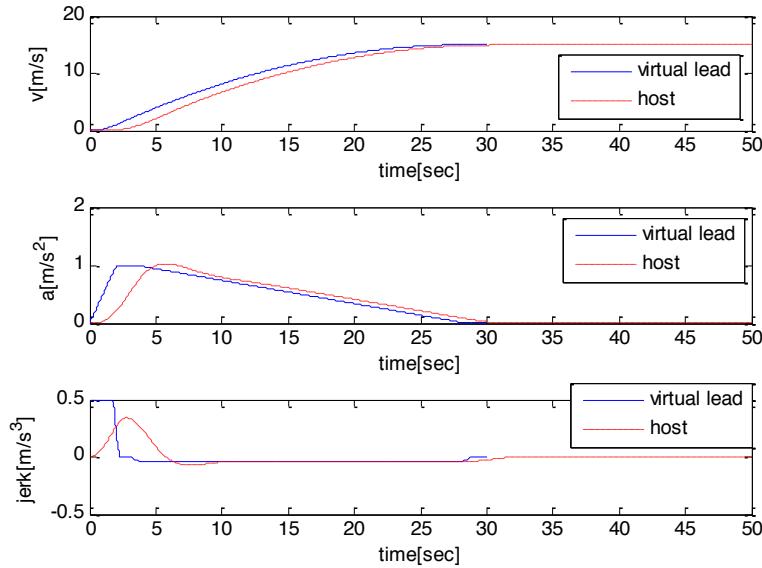


Figure 4.11: Simulation result with a virtual lead vehicle utilizing the optimal velocity profile

host vehicle follows the virtual lead vehicle and eventually follows the actual lead vehicle smoothly.

4.4 Real Time Calculation of Quadratic Programming Problem

Since the optimization problems introduced in the previous sections are all convex and linear or quadratic programming problems, they are solvable using general convex optimization tools like CVX. However, CVX is a tool designed for general convex optimization problems and is too large and slow to embed in a controller. Quadratic programming problems with linear equality and linear inequality constraints can be transformed to linear complementary problems if the matrix representing the quadratic term in the cost function is positive definite. The linear complementary problem can be solved by Lemke algorithm which shows faster computation speed if the variable is not too large [12].

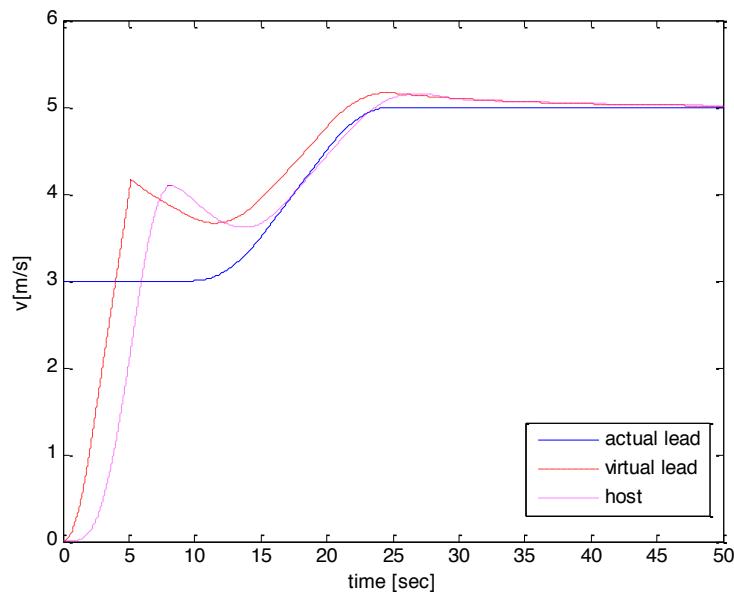


Figure 4.12: Simulation result with a virtual lead vehicle and a slow lead vehicle: velocity

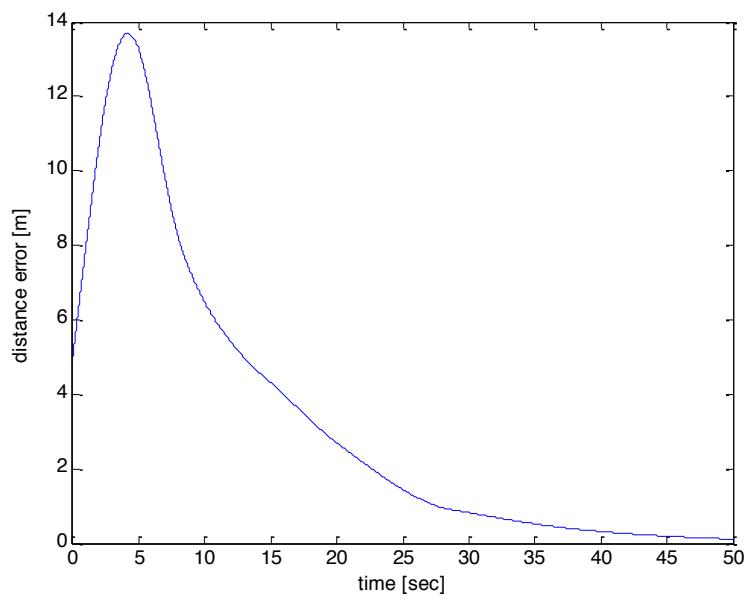


Figure 4.13: Simulation result with a virtual lead vehicle and a slow lead vehicle: distance error

4.4.1 Algorithms to Solve the Constrained QP

The optimization problem (4.17) can be expressed as a standard quadratic programming problem with linear constraints as:

$$\begin{aligned} & \underset{y}{\text{minimize}} \quad \frac{1}{2}y^T Q y + f^T y \\ & \text{subject to} \quad A_{ineq} y - b_{ineq} \leq 0 \end{aligned} \tag{4.37}$$

where Q is a positive definite symmetric matrix.

Since (4.37) is (strictly) convex, it can be solved effectively. One approach is the interior point algorithm which is generally used to solve convex optimization problems, and another is the Lemke algorithm which can solve the corresponding linear complementarity problem of (4.17). In this section, both algorithms will be introduced and compared.

Interior Point Algorithm (Barrier Method) [7]

The interior point algorithm (barrier method) is one of the most popular algorithms solving convex optimization problems. The interior point algorithm converts the problem with inequality constraints to a unconstrained problem by introducing the log barrier function instead of the inequality constraints and starting with a strictly feasible initial point. It is a reliable method and has some better features than the Lemke algorithm especially for real time application.

Generally, a constrained optimization problem is more difficult to solve than a problem without constraints. Some problems like Linear Programming (LP) problems can be solved by the simplex method without modifying the constraints. However it is hard to find the algorithm which can generally solve optimization problems with inequality constraints. Instead, the problems are converted to an unconstrained problem which can be solved by many different algorithms such as the steepest descent method or the Newton method. The interior point method introduces the log barrier function to the cost instead of the inequality constraints. In the quadratic programming problem described in the previous section, the linear inequality constraints are:

$$A_{ineq} y - b_{ineq} \leq 0 \tag{4.38}$$

where $A_{ineq} = [a_{ineq,1} \cdots a_{ineq,m}]^T$. The log barrier function for these inequality constraints is:

$$\phi(y) = - \sum_{j=1}^m \log(-a_{ineq,j}^T y + b_{ineq,j}) \tag{4.39}$$

Due to the property of the log function, if the inequality constraints are about to be violated, the log barrier function becomes arbitrarily large. The log barrier function is added to the cost function to make the problem with only equality constraints as:

$$\underset{y}{\text{minimize}} \quad t(\frac{1}{2}y^T Q y + f^T y) + \phi(y) \tag{4.40}$$

t in the cost function is the weight between the original cost and the log barrier function. Initially, t is small and is increased as the iteration goes on. With a very large t , the solution is almost the same with the original problem.

The unconstrained optimization problem (4.40) can be solved by the Newton method. In the Newton method, the gradient and the Hessian are used to calculate the searching direction. From (4.39) and (4.40), the gradient and the Hessian are:

$$\nabla \left(t \left(\frac{1}{2} y^T Q y + f^T y \right) + \phi(y) \right) = t(Qy + f) + A_{ineq}^T d \quad (4.41)$$

$$\nabla^2 \left(t \left(\frac{1}{2} y^T Q y + f^T y \right) + \phi(y) \right) = tQ + A_{ineq}^T diag(d)^2 A_{ineq} \quad (4.42)$$

where $d_i = 1/(b_{ineq,i} - a_{ineq,i}^T y)$. For each iteration, the gradient and the Hessian are calculated and the descent direction is given as:

$$v^k = -(tQ + A_{ineq}^T diag(d)^2 A_{ineq})^{-1} (t(Qy + f) + A_{ineq}^T d) \quad (4.43)$$

The solution point is updated by:

$$y^{k+1} = y^k + \alpha^k v^k \quad (4.44)$$

where α^k is the step size at k -th iteration. There are many different ways to decide the step size such as the constants step size, the line search, and Armijo rule. In this section, Armijo rule [5] is used. After each iteration, the t is increased and the iteration stops when the difference between the new solution and the previous solution is smaller than the tolerance or the iteration number reaches its maximum value. This procedure is called Phase II.

Since the *log* function is not defined for the values smaller than or equal to zero, the Phase II should start with a strictly feasible initial point. To find the strictly feasible point, another optimization problem is solved, which is called Phase I. The problem is:

$$\begin{aligned} & \underset{y,s}{\text{minimize}} && s \\ & \text{subject to} && A_{ineq}y - b_{ineq} \leq s1_m \end{aligned} \quad (4.45)$$

where $1_m = [1 \ \cdots \ 1]^T \in R^{m \times 1}$.

The Phase I problem can also be solved by the same scheme with Phase II where the initial point can be $y = 0$, $s = \max(-b_i)$. It is not necessary to solve the Phase I completely and the iteration can be stopped when the value of s becomes negative. If the optimal s is nonnegative, the problem is infeasible. In that case, the constraints needs to either be relaxed, i.e. $b_{ineq,new} = \mu b_{ineq}$, $\mu > 1$, or reformulated.

Lemke Algorithm

A QP problem with linear constraints can be converted to a linear complementarity problem which can be efficiently solved by the Lemke algorithm. The Lemke algorithm utilizes tableaux and pivoting as a linear programming solver and is efficient especially when the problem size is small.

Since the problem is convex with only linear constraints, strong duality holds and the Karush-Kuhn-Tucker (KKT) conditions are sufficient for the primal and dual optimal point. The KKT conditions for the problem (4.17) are:

$$A_{ineq}y^* - b_{ineq} \leq 0 \quad (4.46)$$

$$\lambda^* \geq 0 \quad (4.47)$$

$$\lambda_i^* a_{ineq,i}^T y^* - b_{ineq,i} = 0 \text{ for } i = 1, 2, \dots, m \quad (4.48)$$

$$Qy^* + f + A_{ineq}^T \lambda^* = 0 \quad (4.49)$$

where $a_{ineq,i}^T$ s are rows of A_{ineq} . Since Q is strictly positive definite, x^* can be represented as affine functions of λ^* and the KKT conditions (4.46) ~ (4.49) are reduced to:

$$\lambda^* \geq 0 \quad (4.50)$$

$$M\lambda^* + q \geq 0 \quad (4.51)$$

$$(M\lambda^* + q)^T \lambda^* = 0 \quad (4.52)$$

where

$$M = A_{ineq}Q^{-1}A_{ineq}^T \quad (4.53)$$

$$q = b_{ineq} + A_{ineq}Q^{-1}f \quad (4.54)$$

The problem described by (4.50) ~ (4.52) is in the form of a standard linear complementarity problem.

Lemke algorithm is a method solving the linear complementarity problem by utilizing the tableaux and the pivoting scheme. The complexity of the Lemke algorithm is about twice of Gaussian elimination [12] which is efficient when the size of the problem is small. In this research, the lemke algorithm program introduced in [12] is modified and used.

Comparing Interior Point Algorithm and Lemke Algorithm

Table 4.1 compares the calculation time of the Lemke algorithm and the interior point algorithm. As shown in the table, the calculation time of the interior point algorithm increases almost linearly with the problem size while that of the Lemke algorithm increases exponentially. So, the interior point method has its advantage in the calculation time when the size of the variable is large. However if the dimension of the problem is small, the Lemke algorithm can solve the problem faster than the interior point method.

$\Delta t/T_f$	0.05 / 10	0.05 / 5	0.1 / 10	0.1 / 5
I.C.				
1 / 0.5 / -0.1	8.745 / 2.176	0.4363 / 0.7434	1.008 / 0.8485	0.1756 / 0.9189
2 / 1 / -0.2	7.818 / 1.941	0.2906 / 0.7234	0.8616 / 0.6474	0.0830 / 0.9787
2 / 1 / 0	8.313 / 1.804	0.3528 / 0.6694	0.8750 / 0.6917	0.1048 / 0.9707
5 / 3 / -0.7	25.86 / 1.819	0.4796 / 0.5480	1.406 / 0.7114	0.1589 / 0.9041

Table 4.1: Calculation time [sec] of the Lemke algorithm / Interior point algorithm (Initial Condition : $d_0[m], v_0[m/s], a_0[m/s^2]$)

Observing the result in the last column, it can be observed that the calculation time of the interior point algorithm is longer than the second last column whose dimension is larger. If checking the calculation time of the Phase I and Phase II of the interior point method separately, it is found that the majority of the time is consumed in Phase I to find the feasible initial point. Since the interior point method should start with a strictly feasible initial point, the shorter time horizon and the longer time step make it hard to find a strictly feasible point due to the acceleration and jerk constraints. Hence, it takes longer for the interior point method to solve the problem.

However, it does not necessarily mean that the Lemke algorithm is better than the interior point method when the initial point is close to the infeasible region. One drawback of the Lemke algorithm is that the performance of the Lemke algorithm is poor if the initial condition is poorly conditioned. If the initial distance is very short and the initial velocity is fast, the host vehicle cannot stop exactly at the desired point with the limited acceleration and jerk. In that case, the initial condition is infeasible and no solution exists for the optimization problem. However, the Lemke algorithm cannot distinguish the infeasible initial condition until it solves the problem and finds out that the solution is infeasible. Also, for some initial conditions which are feasible but close to being infeasible, the solution of the Lemke algorithm does not converge. Figures 4.14 and 4.15 show the simulation results with the initial conditions $(2[m], 1[m/sec], 0.05[m/sec^2])$ and $\Delta t = 0.25[\text{sec}]$ which is very close to an infeasible problem. Although the interior point took longer to solve the problem, the Lemke algorithm failed to solve the problem.

Also, due to the log barrier, the intermediate solutions of the interior point algorithm are always feasible. Figures 4.14 and 4.15 show that the intermediate solutions of the interior point methods are always feasible while those of the Lemke algorithm are infeasible. Since the intermediate solutions of the interior point algorithm are always feasible, the interior point algorithm has an advantage for real time applications. Also, Figures 4.14 and 4.15 show that the the interior point algorithm converges exponentially to the final solution while the Lemke algorithm converges linearly. In the real time application, even if the calculation is not finished on time, the intermediate solution at that time can be utilized since it is still feasible and close to the optimal solution. For real time applications, the accuracy of the solution can be sacrificed to reduce the calculation time. To do that, the iteration number

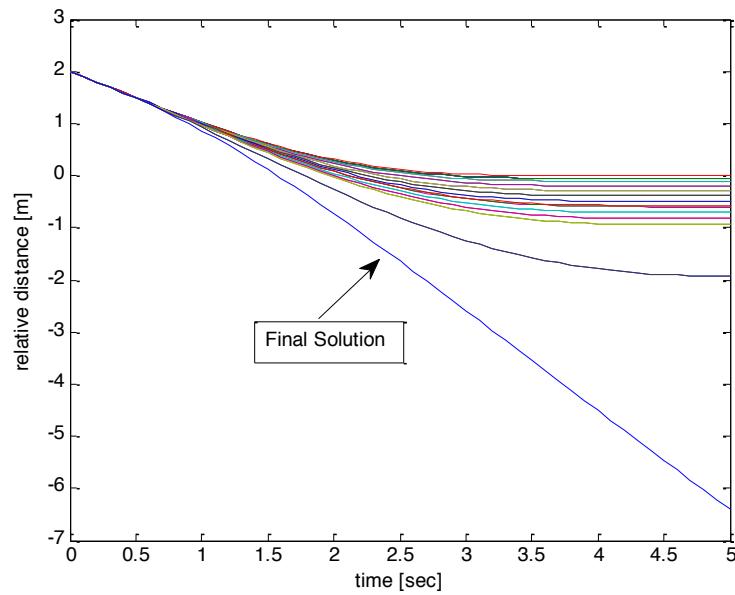


Figure 4.14: Intermediate and final solutions of the Lemke algorithm

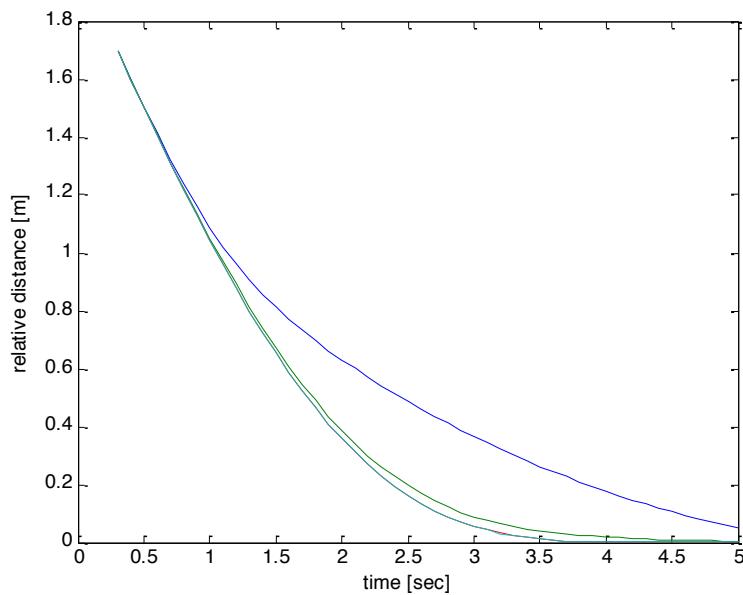


Figure 4.15: Intermediate and final solutions of the interior point algorithm

or the calculation time can be limited or the algorithm can be terminated once the tolerance has been met. However those schemes are not applicable with the Lemke algorithm since the intermediate solutions are infeasible.

One of the biggest advantages of the Lemke algorithm is that it is very fast when the size of the problem is small. In this thesis, instead of using the intermediate solution, the size of the problem will be reduced and the Lemke algorithm will be utilized. The scheme to reduce the problem size will be introduced in the next section.

4.4.2 Multi-resolution Optimization Problem Formulation

In the previous section, the interior point method and the Lemke algorithm are suggested as candidates for the solving the quadratic programming problem (4.37). As shown in 4.1, the dimension of the problem has a significant effect on the calculation time. In this section, a multi-resolution problem formulation will be introduced as a method to find a suboptimal solution in real time. The multi-resolution scheme has been used to reduce the number of meshes for computational fluid dynamics [53] and to compress the size of the image in image processing [44]. However, it has not been utilized in the area of real time optimization or real time trajectory generation. The suggested multi-resolution scheme is also utilizable for Model Predictive Control problems.

Applications of the Optimal Solution

In previous sections, sliding control and virtual lead vehicle scheme were suggested as candidate control schemes which can control the vehicle to follow the optimal velocity or position profile. Both control schemes utilize the optimal profile as a trajectory that the output of the system should follow. The optimal profile and its time derivatives are used in the sliding controller. When sliding controller is used, the optimization problem is solved only once and the solution is saved and utilized for the following time steps. One thing to notice is that in both controller, only the first part of the optimization problem is utilized at the time when the optimization problem is solved.

Multi-resolution Problem Formulation

As shown in the previous section, the calculation time significantly depends on the dimension of the optimization problem. If the controller time step Δt is smaller, the dimension of the optimization problem becomes larger and it takes longer to solve the problem. However, since both controller utilize the first part of the optimal solution, it is possible to decrease the dimension of the optimization problem by using larger time step for later time horizon.

$$\begin{bmatrix} x_{k+1} \\ v_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ v_k \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix} a_k \quad (4.55)$$

(4.55) was utilized in previous sections and it was assumed that the accelerations are constant during each time step Δt . In that case the time step was a fixed constant and the dimension

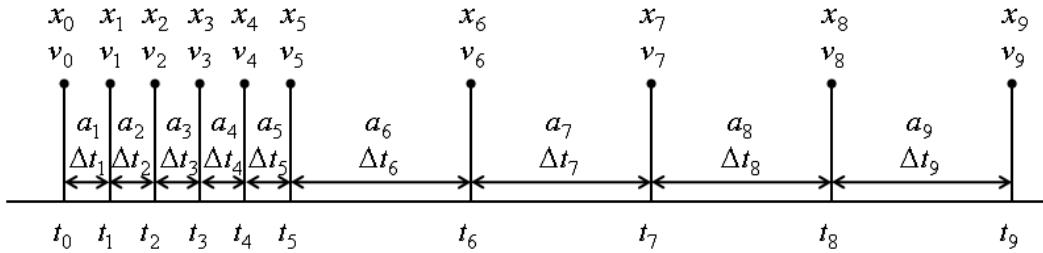


Figure 4.16: Sketch of the multi-resolution problem concept

of the problem was $T_f/\Delta t$. However if a smaller time step is used for the initial time horizon and a longer time step for later time, the problem size can be decreased. Figure 4.16 shows the concept of the multi-resolution problem. If the time horizons is 5 second and a constant time step of 0.2 second is used, the size of the problem is 25. However, if the time step is 0.2 second for the first 1 second and 1 for the later 4 seconds, the size of the problem becomes 9.

The optimization problem is in a same form with (4.37), however the matrices in the weighting function and the constraints are modified according to the multi-resolution time step.

Applications and Simulation Results

Figure 4.17 compares the solution with a uniform time step and a multi-resolution formulation. The uniform time step has a sampling time of 10[msec]. For the multi-resolution formulation, 10[msec] is used for the first 100[msec], 100[msec] is used from 0.1[sec] to 1[sec], and 1[sec] is used from 2 to 5[sec]. The calculation time were 3.9729[sec] for the uniform time step and 0.0045[sec] for the multi-resolution formulation. As shown in the figure, the optimal solutions are different in the later time steps, but are very similar in the first 0.1[sec] where the time steps are the same.

When the sliding controller is used, the desired trajectory should be given. If the desired trajectory is calculated in real time, the optimization problem should be solved once at the very first time step to calculate the desired trajectory. However at the very first time step, only the values of the very first part of the trajectory are utilized and the overall solution of the optimization problem is saved as a desired trajectory for later usage. Since the solution of the later time horizon does not need to be calculated very accurately at that very first time step, the multi-resolution formulation is also applicable to calculate the desired trajectory for the sliding controller.

Figure 4.18 describes the concept of using the multi-resolution problem formulation to solve for the optimal trajectory. As shown in the figure, a shorter time step (same with the control time step) is used for the first part of the time horizon while a longer time step is

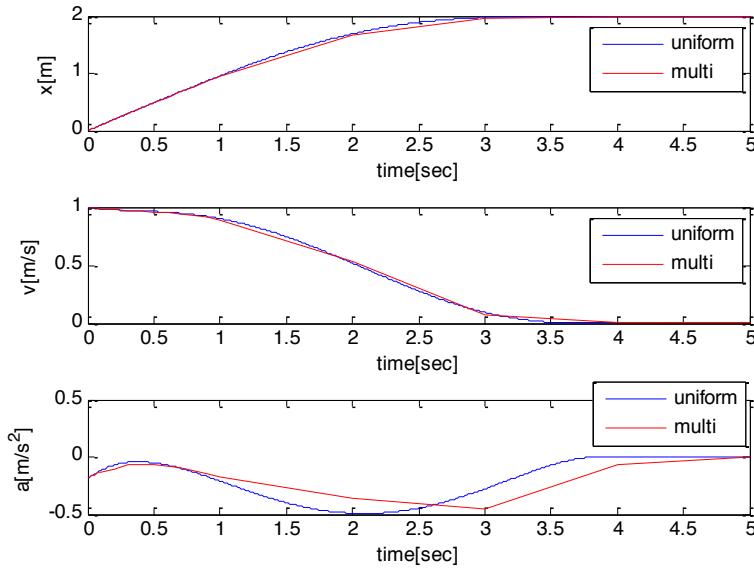


Figure 4.17: Solutions of the optimization problems with uniform time step and multi-resolution time step

used for the later part. At the next time step, the time horizon with the shorter time step is shifted but is overlapped with the previous time horizon with the shorter time step. The solution of the overlapped time from the previous problem is applied as equality constraints for the next problem. Note that at this moment, the controller still utilizes the trajectory from the first problem solved in the previous time step. However the controller solves the optimization problem for later time. The overlap is to prevent the discontinuity of the final solution.

Figure 4.19 compares the solutions of the uniform time step and the multi-resolution time step. 10[msec] sampling time is used for the uniform time step. For the multi-resolution formulation, 10[msec] is used for the first 0.2[sec], 0.1[sec] is used from 0.2[sec] to 1[sec], and 1[sec] is used for the later time horizon. Only the solutions of the first 0.2[sec] time horizon are saved and used. The calculation time of the uniform time horizon problem was 3.9842[sec] and the average calculation time of the multi-resolution problems was 0.005039[sec]. As shown in the figure, two solutions are similar.

Figures 4.20 and 4.21 show the simulation result with the sliding controller whose desired trajectory is calculated in real time. As shown in the figures, with the sliding controller the vehicle stops completely faster than with the PID controller.

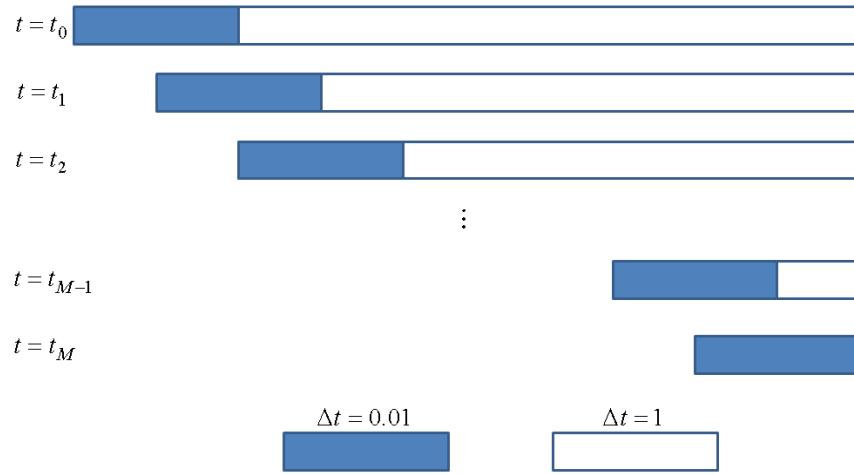


Figure 4.18: Concept of Multi-resolution formulation to calculate the optimal trajectory for the sliding control

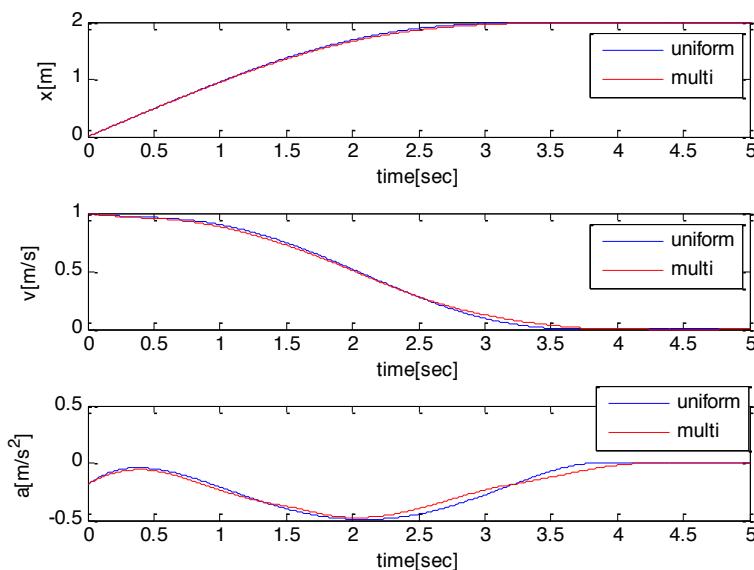


Figure 4.19: Solutions of the optimization problems with uniform time step and multi-resolution time step

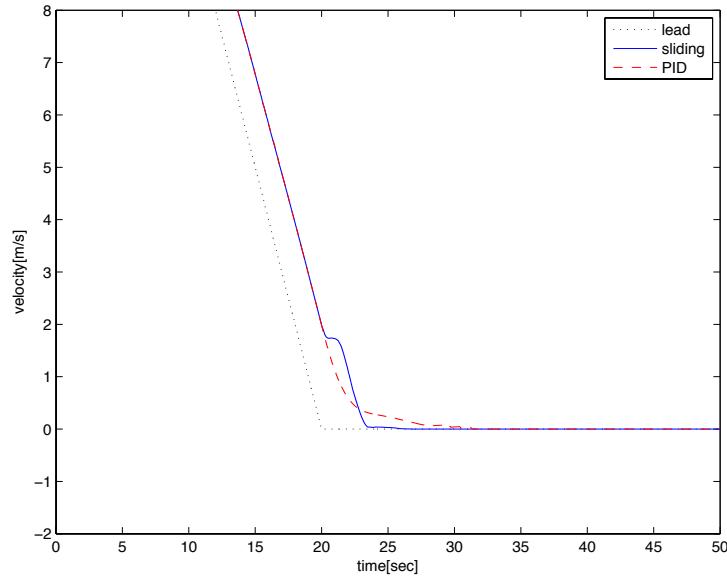


Figure 4.20: Simulation result with a sliding controller in case of a complete stop: velocity [m/sec]

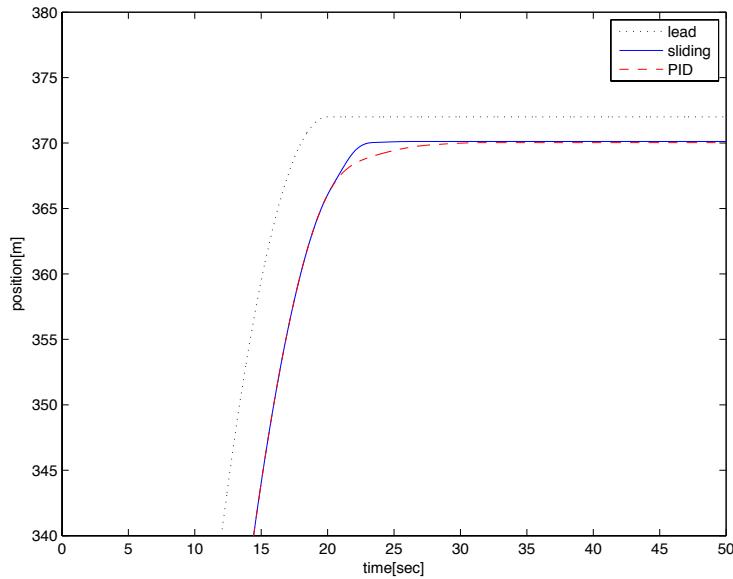


Figure 4.21: Simulation result with a sliding controller in case of a complete stop: position [m]

4.5 Summary

In this chapter, optimal trajectories for a complete stop and a starting motion were calculated from a constrained Quadratic Programming (QP) problem. The constant time-headway policy caused the too slow motion when the vehicle stopped completely. A QP problem was formulated to find the optimal velocity profile for the complete stop. l^1 norm was introduced as a convex relaxation of the l^0 norm. Thanks to the sparse property, the l^1 norm made the velocity converge to zero faster.

Sliding controller was designed to control the host vehicle to follow the optimal trajectory during a complete stop. Smooth response was achieved by introducing saturation function instead of the sign function. For the starting motion, the optimal trajectory was utilized by a virtual lead vehicle to better react to a slow lead vehicle.

Interior point method and Lemke algorithm were studied as the methods to solve the optimization problem. The interior point method showed faster performance than the Lemke algorithm when the size of the problem was large. Also, the interim solutions of the interior point method were feasible. However, the Lemke algorithm was faster than the interior point method when the size of the problem was small and the calculation time was almost linear to the size of the problem.

To solve the optimization problem in real time, a multi-resolution formulation was suggested to reduce the problem size and shorten the calculation time. The multi-resolution formulation was utilized to calculate the optimal trajectory in real time for sliding controller. With the sliding controller, the host vehicle stopped completely faster than with a PID controller whose desired relative distance was calculated by the constant time-headway policy.

Chapter 5

Cooperative Adaptive Cruise Control

5.1 Introduction

Cooperative Adaptive Cruise Control(CACC) is an enhanced version of the ACC. The main difference of the CACC from the ACC is that CACC utilizes communication between the vehicles and/or between the vehicle and the road structure. This communication allows the control system on any single vehicle to get information about other vehicles in the platoon like the acceleration, the velocity, the throttle control command, and the brake control command. Recently, various research have been done about the CACC in the area of traffic control and communication. Many previous research show that the CACC has the potential to improve both the traffic flow [3] and the string stability [33].

Many existing research on the CACC system follow a hierarchical controller structure where the system consists of an upper lever controller and a lower level controller [8]. The upper level controller determines the desired acceleration based on the information acquired from the sensors and through inter vehicle communications. Based on the desired acceleration, the lower level controller determines the throttle and/or brake control command. Most of the research about the CACC have been focused on designing the upper level control.

One approach for the longitudinal control of the CACC is utilizing the front vehicle acceleration in the feedforward controller [3, 33, 43]. Another approach is building a world model based on the information from multiple vehicles and find an appropriate acceleration command based on the world model [8, 25, 40]. In existing studies, the accelerations were determined either based on the states of the lead vehicle which causes the largest potential danger [8], or from a learning algorithm [25]. Another longitudinal control approach was to find the desired acceleration based on the velocity and the distance errors from some vehicles in the upstream of the vehicle string [45]. The CACC is also used for the lateral control and the case when the vehicles merge in intersections [40, 61, 19].

In this chapter, only the longitudinal control problem will be considered. Instead of utilizing only the front vehicle acceleration or determining the desired acceleration based on the largest potential danger, the vehicle platoon itself is considered as a single system whose

inputs are the accelerations of all the vehicles in the platoon. In Section 2.4, the Linear Quadratic optimal control problem was solved to find the feedback gains. In this chapter, a vehicle platoon is modeled as a set of double integrator and the LQ problem is solved. As a result, the optimal feedback control law is a state feedback control law where the states are the relative positions and velocities of all vehicles in the platoon.

When the vehicle platoon is considered as a single system, cutting in and out of vehicles can be an issue. Virtual lead vehicle scheme was studied in Section 3 in case of a single host vehicle. The virtual lead vehicle scheme is also utilizable in case of a vehicle platoon. Instead of utilizing the actual relative distance, the relative distance to the virtual lead vehicle is utilized when a vehicle cuts in or out to prevent a sudden acceleration and deceleration.

Each vehicle in a platoon can measure the relative distance to the front vehicle and its own velocity. However, the other relative distances and velocities should be transmitted through wireless communication networks. In this chapter, only vehicle-to-vehicle communication will be considered and it will be assumed that there is no communication between the vehicles and the road structure. The information shared by each vehicle will be the relative distance to the lead vehicle, the velocity of the lead vehicle, and the velocity and the acceleration of itself. The wireless network, however, is usually lossy and it is not guaranteed that all the data are communicated for every single control time step. In this chapter, various control schemes when the data are lost are compared by simulations.

This chapter is organized as follows. In Section 5.2, an LQ optimal control problem is solved for a vehicle platoon to find feedback control gains. In Section 5.3, the virtual lead vehicle scheme is utilized to control the platoon react smoothly to cutting in or out vehicles. In Section 5.4, packet loss during wireless communications is considered and various control scheme are compared by simulations. Section 5.6 summarizes the chapter.

5.2 Linear Quadratic Optimal Controller for a Vehicle Platoon

In Section 2.4, the LQ optimal control problem was formulated and solved for one host vehicle and one lead vehicle. In this section, the problem will be solved for a platoon of vehicles equipped with identical CACC systems.

5.2.1 Problem Description

In this section, a vehicle platoon with length N is considered. The first vehicle is a lead vehicle driven by a human driver and $N - 1$ vehicles equipped with identical CACC systems are following the lead vehicle. The linear model can be formulated as:

$$\dot{X}(t) = AX(t) + BU(t) \quad (5.1)$$

$$Y(t) = CX(t) \quad (5.2)$$

$$X(t) = \begin{bmatrix} x_l(t) - x_1(t) \\ x_1(t) - x_2(t) \\ \vdots \\ x_{N-2}(t) - x_{N-1}(t) \\ v_l(t) \\ v_1(t) \\ \vdots \\ v_{N-1}(t) \end{bmatrix} \quad (5.3)$$

$$U(t) = [a_l(t) \ a_1(t) \ \cdots \ a_{N-1}(t)]^T \quad (5.4)$$

$$Y(t) = \begin{bmatrix} t_{hw}v_1(t) - (x_l(t) - x_1(t)) \\ \vdots \\ t_{hw}v_{N-1}(t) - (x_{N-2}(t) - x_{N-1}(t)) \end{bmatrix} \quad (5.5)$$

$$A = \begin{bmatrix} & 1 & -1 & 0 & \cdots & 0 \\ 0_{(N-1) \times (N-1)} & 0 & 1 & -1 & \ddots & \vdots \\ & \vdots & \ddots & \ddots & \ddots & 0 \\ & 0 & \cdots & 0 & 1 & -1 \\ 0_{N \times (N-1)} & & & & 0_{N \times N} & \end{bmatrix} \quad (5.6)$$

$$B = \begin{bmatrix} 0_{(N-1) \times N} \\ I_{N \times N} \end{bmatrix} \quad (5.7)$$

$$C = [-I_{(N-1) \times (N-1)} \ 0_{(N-1) \times 1} \ t_{hw}I_{(N-1) \times (N-1)}] \quad (5.8)$$

where x_i , v_i , and a_i are the position, the velocity, and the acceleration of the i^{th} vehicle.

5.2.2 Linear Quadratic Optimal Control

If the plant is time invariant, (A, B) is controllable, and (A, C) is observable, the solution of the stationary LQ problem exists for the performance index

$$J = \frac{1}{2} \int_{t_0}^{\infty} X^T(t) Q X(t) + U^T(t) R U(t) dt \quad (5.9)$$

where $Q = C^T C$.

However with the C given in (5.8), (A, C) is not observable. To make it observable, while not modifying the performance index, a new C is given as:

$$C = \begin{bmatrix} -I_{(N-1) \times (N-1)} & 0_{(N-1) \times 1} & t_{hw}I_{(N-1) \times (N-1)} \\ 0_{1 \times (N-1)} & \epsilon_o & 0_{1 \times (N-1)} \end{bmatrix} \quad (5.10)$$

As mentioned in Section 2.4, the number ϵ_o is a very small number that can be further reduced depending on the available computational resolution.

Similar to Section 2, the input penalty matrix R is:

$$R = \lambda \begin{bmatrix} 1/\epsilon_c & 0_{1 \times (N-1)} \\ 0_{(N-1) \times 1} & I_{(N-1) \times (N-1)} \end{bmatrix} \quad (5.11)$$

where λ is a design parameter to tune. The number ϵ_c is a very small number that can be further increased depending on the available computational resolution. The state feedback control law is given as:

$$U(t) = -R^{-1}B^T P_+ X(t) = -KX(t) \quad (5.12)$$

where P_+ is the positive definite solution of the algebraic Riccati equation

$$A^T P + PA - PBR^{-1}B^T P + C^T C = 0 \quad (5.13)$$

For example, with $N = 5$, $t_{hw} = 2$, $\lambda = 1$, and $\epsilon_o = \epsilon_c = 1e-5$ the state feedback gain is calculated to be:

$$K = \begin{bmatrix} 1.1e-6 & -4.6e-7 & -4.3e-7 & -1.8e-7 & 5.4e-6 & -4.7e-6 & -7.7e-7 & -8.5e-8 & 4.2e-8 \\ -0.9952 & 0.0974 & 0.0098 & -1.2e-4 & -0.4726 & 2.4788 & -0.1996 & -0.0256 & -0.0048 \\ -0.0960 & -0.9906 & 0.0967 & 0.0074 & -0.0765 & -0.1996 & 2.4685 & -0.2019 & -0.0252 \\ -0.0190 & -0.0942 & -0.9912 & 0.0914 & -0.0085 & -0.0256 & -0.2019 & 2.4662 & -0.2043 \\ -0.0023 & -0.0160 & -0.0903 & -0.9958 & 0.0042 & -0.0048 & -0.0252 & -0.2043 & 2.4391 \end{bmatrix} \quad (5.14)$$

As described in (5.14), for each vehicle in the platoon, the largest gains are applied for its own velocity and the relative distance to its lead vehicle. Relatively higher gain is also applied for the velocity of the lead vehicle. With these three gains, the control law is similar to a PD controller. Additionally, the velocity of the vehicle right behind, the relative distances of the lead vehicle to its lead vehicle, and the relative distance to the vehicle behind are also multiplied to relatively larger gains and considered in the feedback control law. The first row which will be related to the lead vehicle acceleration will be rounded off to zero.

5.2.3 Simulation Result

Figures 5.1 ~ 5.3 show the simulation results comparing the CACC system designed in the previous section and the ACC system designed in Section 2.4. In the simulation, each vehicle is modelled as a double integrator whose input is the acceleration. Aerodynamic drag and rolling resistance are included in the model and the brake/engine is modeled as a first order system. Disturbance observer is utilized for each vehicle to compensate the disturbances and the unmodeled actuator dynamics. In the simulation, a vehicle platoon of 6 vehicles is considered. The lead vehicle is assumed to be driven by a human driver and the following vehicles are assumed to be controlled by a CACC system. Solid lines are from the simulation result when the CACC controller with the feedback gain given in (5.14) is used and dashed lines are from the simulation result when each host vehicle utilizes the ACC controller whose gain was from the LQ controller designed in Section 2.4.

Figure 5.1 shows the velocities. As shown in the figure, the lead vehicle decelerates from 10[m/sec] to 5[m/sec], maintains the speed, and accelerates back to 10[m/sec]. The

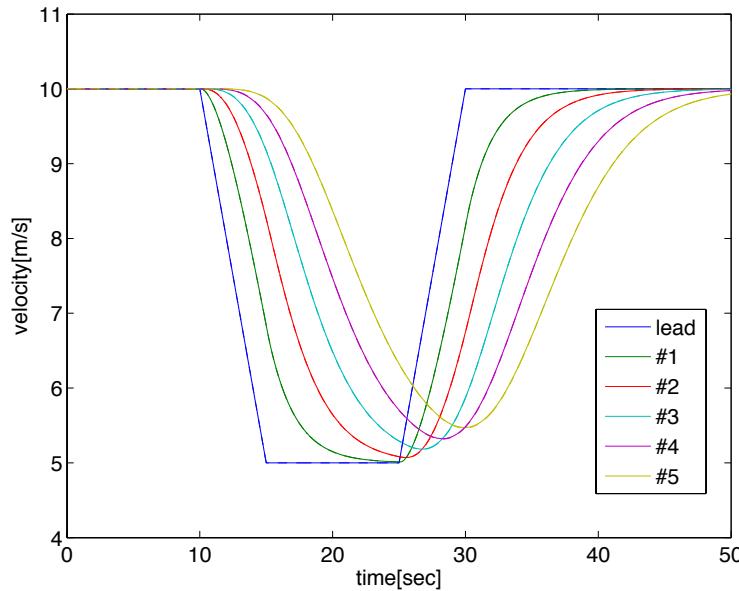


Figure 5.1: Simulation results with LQ controllers: velocity (solid lines: with CACC, dashed lines: with ACC)

velocities with the ACC and CACC are very similar, however the errors are different as shown in Figures 5.2. As shown in Figure 5.2, the peak error of the vehicle #1 is reduced with the CACC compared to the result with the ACC. As described before, the CACC additionally considered the velocities and the relative distances of other vehicles in the platoon. That makes the vehicle react faster and the errors of the remaining vehicles are more negative when the lead vehicle decelerates and more positive when the lead vehicle accelerates while the errors with the ACC system is the opposite. Negative error means that the relative distance to the front vehicle is longer than the desired distance and hence is more desirable and safer than a positive error especially when the lead vehicle is decelerating. When the lead vehicle is accelerating, safety is not a big issue and positive error is acceptable and also desirable since it means faster reaction. In other words, it is better for the sign of the error to be the same with the sign of the lead vehicle acceleration. Figure 5.3 shows the plot of the product of the error and the lead vehicle acceleration. As shown in the figure, with the CACC the values are positive in most of time while the values are mostly negative with the ACC.

5.3 Cut In/Out of a Vehicle in a Platoon

As studied in the previous section, the feedback gain of a CACC system depends on the number of vehicles in the platoon. When a vehicle cuts into the platoon, or cuts out from

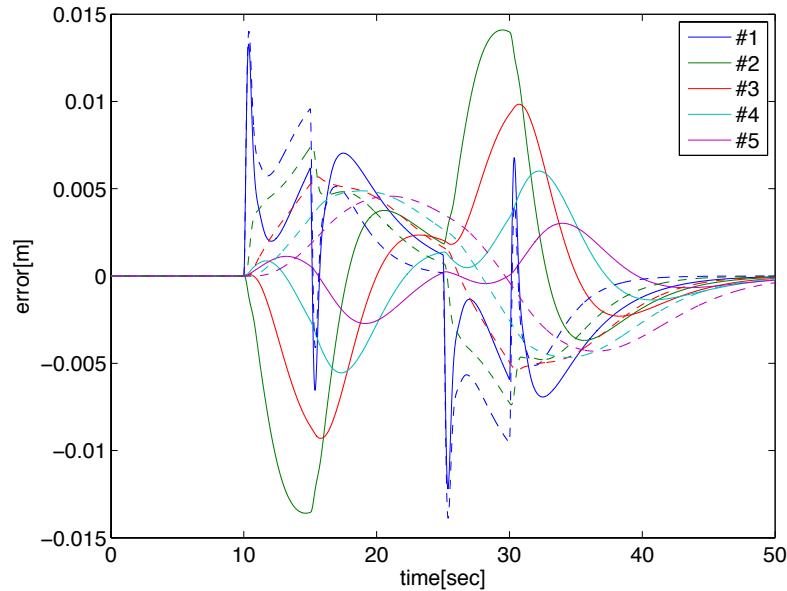


Figure 5.2: Simulation results with LQ controllers: error (solid lines: with CACC, dashed lines: with ACC)

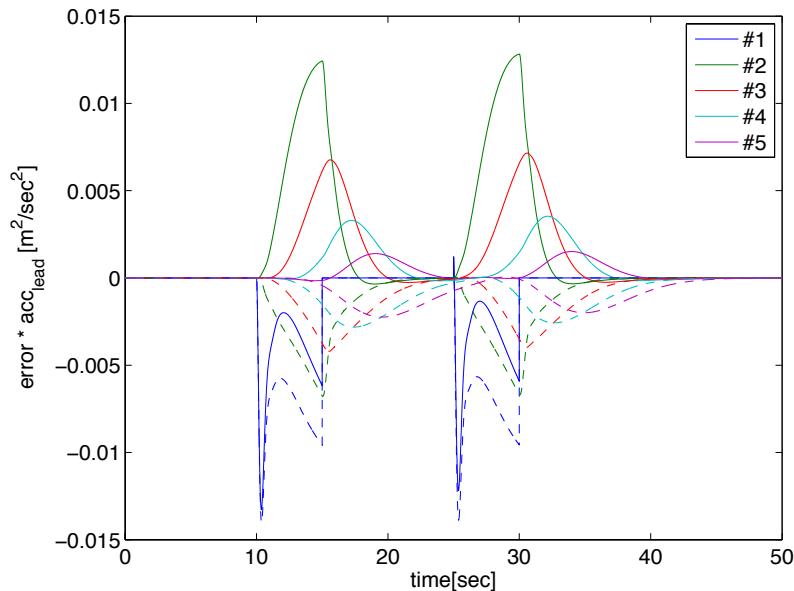


Figure 5.3: Simulation results with LQ controllers: product of the error and the lead vehicle acceleration (solid lines: with CACC, dashed lines: with ACC)

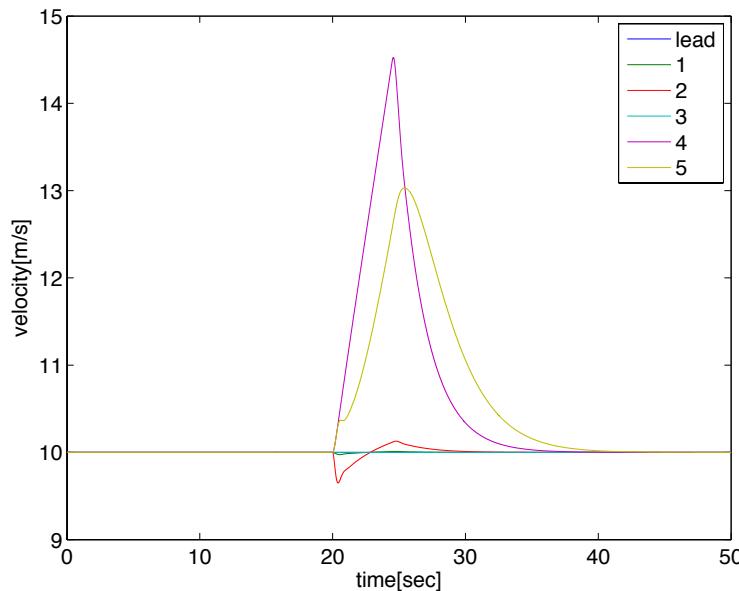


Figure 5.4: Simulation results when a vehicle cuts out from the platoon

the platoon, there is a sudden change in the states of the platoon and that can cause some undesirable acceleration or deceleration if there is no proper transient control scheme. As shown in Section 3, the virtual lead vehicle scheme is useful for a single vehicle ACC system when the lead vehicle cuts out or another vehicle cuts in between the host and the lead vehicles. In this section, the virtual lead vehicle scheme is utilized for smooth transient motion control when a vehicle cuts in to or out from the platoon.

5.3.1 Problem Description

Figure 5.4 shows the simulation result for the case where a vehicle cuts out from the platoon. As described in Figure 5.5, a vehicle platoon of 6 vehicles is considered in the simulation where the lead vehicle is moving with a constant speed. At 20 second, the vehicle #3 cuts out from the platoon. Without any transient controller, there is a sudden large step change in the error for the vehicle #4. That makes the vehicle #4 accelerate suddenly and also the vehicle #5 since it follows the vehicle #4. Also, when the vehicle #3 cuts out the relative distance between the vehicles #2 and #4 becomes very larger than it should be. That makes the vehicle #2 decelerate to compensate the error. The deceleration of the vehicle #2 is very undesirable and unnecessary since it does not improve safety while worsens the fuel efficiency. Also it can make the passengers feel very uncomfortable since the passengers may not know the cutting out of the vehicle behind.

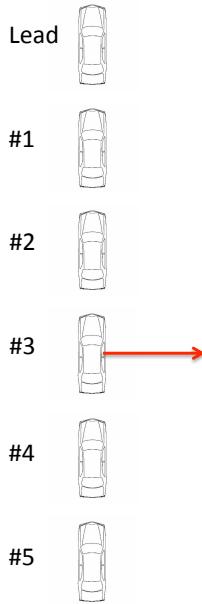


Figure 5.5: Sketch of the cutting out scenario

5.3.2 Virtual Lead Vehicle Scheme

For the case described in Figure 5.5, the state of the LQ formulation before the cutting out is given as:

$$X(t) = \begin{bmatrix} x_l(t) - x_1(t) \\ x_1(t) - x_2(t) \\ x_2(t) - x_3(t) \\ x_3(t) - x_4(t) \\ x_4(t) - x_5(t) \\ v_l(t) \\ v_1(t) \\ v_2(t) \\ v_3(t) \\ v_4(t) \\ v_5(t) \end{bmatrix} \quad (5.15)$$

When the vehicle #3 cuts out, the new state is:

$$X(t) = \begin{bmatrix} x_l(t) - x_1(t) \\ x_1(t) - x_2(t) \\ x_2(t) - x_4(t) \\ x_4(t) - x_5(t) \\ v_l(t) \\ v_1(t) \\ v_2(t) \\ v_4(t) \\ v_5(t) \end{bmatrix} \quad (5.16)$$

Since the relative distance $x_2(t) - x_4(t)$ is very large when the cutting out happens, the vehicle #4 accelerates and #2 decelerates to reduce the relative distance. To prevent this, the virtual lead vehicle scheme is introduced. Let $x_{v,i}(t)$ the position of the virtual lead vehicle followed by the vehicle $\#i$. Assuming that the initial positions and the initial velocities of the virtual lead vehicles are the same with those of the actual lead vehicles, (5.15) can be represented as:

$$X(t) = \begin{bmatrix} x_l(t) - x_1(t) \\ x_1(t) - x_2(t) \\ x_2(t) - x_3(t) \\ x_3(t) - x_4(t) \\ x_4(t) - x_5(t) \\ v_l(t) \\ v_1(t) \\ v_2(t) \\ v_3(t) \\ v_4(t) \\ v_5(t) \end{bmatrix} = \begin{bmatrix} x_{v,1}(t) - x_1(t) \\ x_{v,2}(t) - x_2(t) \\ x_{v,3}(t) - x_3(t) \\ x_{v,4}(t) - x_4(t) \\ x_{v,5}(t) - x_5(t) \\ v_l(t) \\ v_1(t) \\ v_2(t) \\ v_3(t) \\ v_4(t) \\ v_5(t) \end{bmatrix} \quad (5.17)$$

Similarly, the new state (5.16) is also represented as:

$$X(t) = \begin{bmatrix} x_{v,1}(t) - x_1(t) \\ x_{v,2}(t) - x_2(t) \\ x_{v,4}(t) - x_4(t) \\ x_{v,5}(t) - x_5(t) \\ v_l(t) \\ v_1(t) \\ v_2(t) \\ v_4(t) \\ v_5(t) \end{bmatrix} \quad (5.18)$$

Comparing (5.16) and (5.18), with the virtual lead vehicle scheme only the relative distances between the virtual lead vehicles and the host vehicles are utilized. That prevents

the sudden change of the relative distance is the state and makes the state to be continuous if the virtual lead vehicle is properly controlled to merge to the actual lead vehicle smoothly when a vehicle cuts in or out.

5.3.3 Simulation Results

Simulations are performed to show the performance of the virtual lead vehicle scheme. In the simulations, each vehicle has its own virtual lead vehicle. At steady state, the position and the velocity of the virtual lead vehicle are the same with the position and the velocity of the actual lead vehicle. When a vehicle cuts in or cuts out, the virtual lead vehicle merges to the new lead vehicle by the controller designed in Chapter 3. The state feedback control law is utilized with the control gains found in Section 5.2.

Figure 5.6 shows the simulation result with the virtual lead vehicle scheme. As shown in the figure, the vehicle #4 does not accelerates much when the cutting out happens. Instead, it follows the virtual lead vehicle which smoothly merges into the vehicle #2. The peak velocity and the convergence time depends on the virtual lead vehicle whose controller gains can be adjusted based on the preference of the driver. Also, the vehicle #2 does not decelerate unnecessarily. Figures 5.7 and 5.8 show the simulation result when two vehicles cut out from the platoon at the same time. As shown in the figures, with the virtual lead vehicle scheme there is no excessive acceleration and the velocities and the positions of the vehicles converge smoothly to their desired values.

Figures 5.10 ~ 5.12 show the simulation results when a vehicle cuts into the platoon. As described in Figure 5.9, initially a vehicle platoon of 5 vehicles is considered where the lead vehicle is moving with a constant speed. At 20 second, vehicle #3 cuts into the platoon from the side lane. Figure 5.10 shows the result without any transient controller. As shown in the figure, without any transient control scheme, vehicles #3 ~ #5 decelerates rapidly. Also, the vehicle #2 accelerates to increase the relative distance between the vehicles #2 and #3. It is very undesirable since it can make the passengers feel uncomfortable since they may not know that the vehicle cut in.

As shown in Figures 5.11 and 5.12, the vehicles #3 ~ #5 does not excessively decelerate with the virtual lead vehicle scheme. Instead they decelerate smoothly to increase the relative distances. Once the relative distances are increased enough, the vehicle accelerates and maintains the speed same with the lead vehicle. Also, it can be observed that the vehicle #2 does not unnecessarily accelerate. In this simulation, it is assumed that the speed of the vehicle #3 is the same with other vehicles in the platoon. Also the CACC system of the vehicle #3 is activated right after the cutting in. If the speed of the vehicle that cuts in is slower than the speed of the platoon, it can be a dangerous situation and a separate safety feature such as the anti collision system should be utilized.

Comparing (5.17) and (5.18), the length of the state vector is changed after the cutting out. It is the same for the cutting in case. This can make the feedback control command be discontinuous when the cutting in or out occurs. To prevent the discontinuities, some transient control scheme can be introduced. For example, a fuzzy controller can be used whose

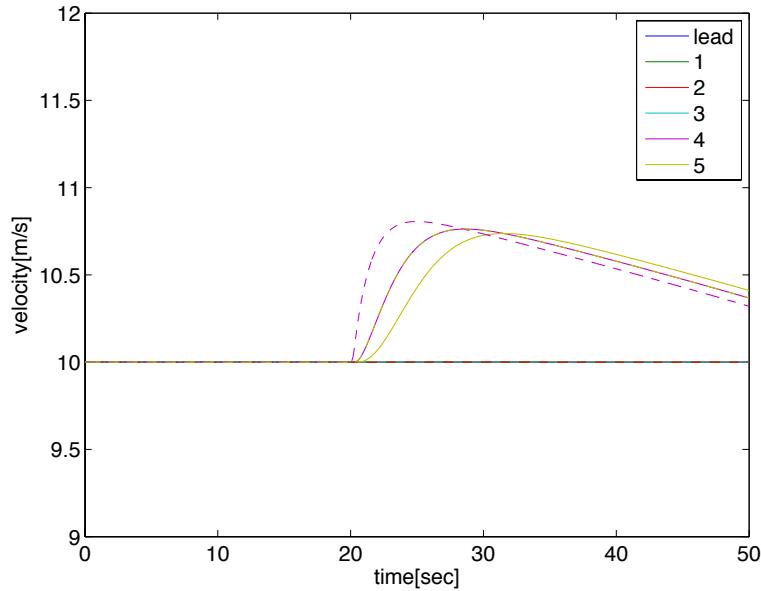


Figure 5.6: Simulation results when a vehicle cuts out from the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)

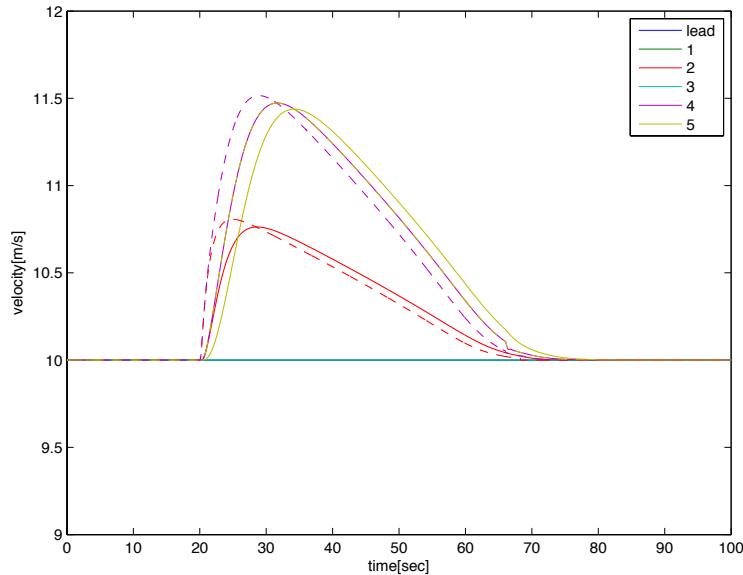


Figure 5.7: Simulation results when two vehicles cut out from the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)

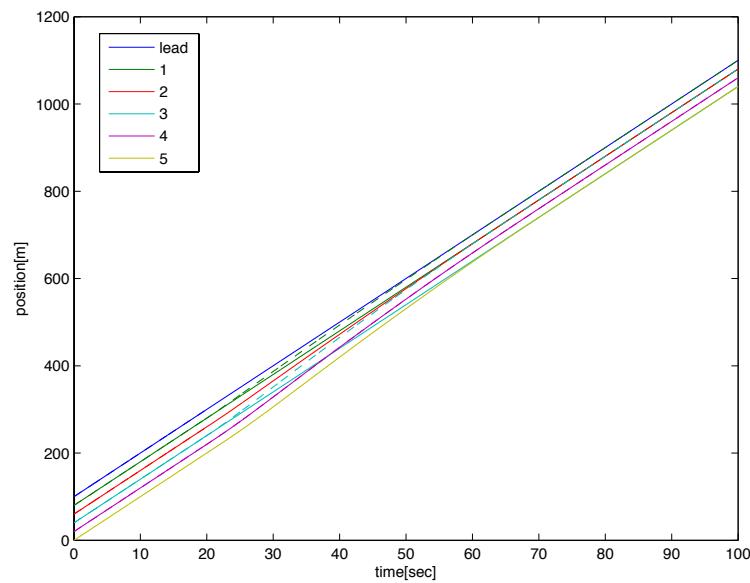


Figure 5.8: Simulation results when two vehicles cut out from the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)

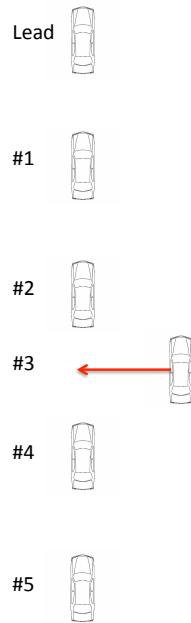


Figure 5.9: Sketch of the cutting in scenario

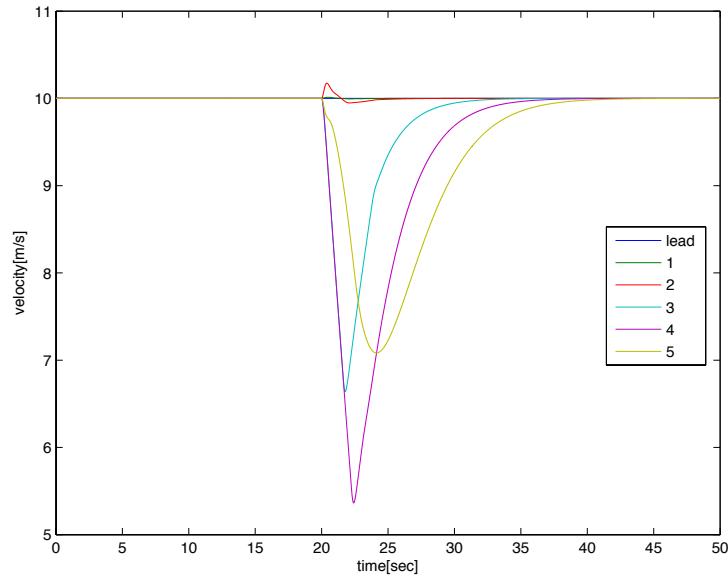


Figure 5.10: Simulation results without the virtual lead vehicle when a vehicle cuts into the platoon

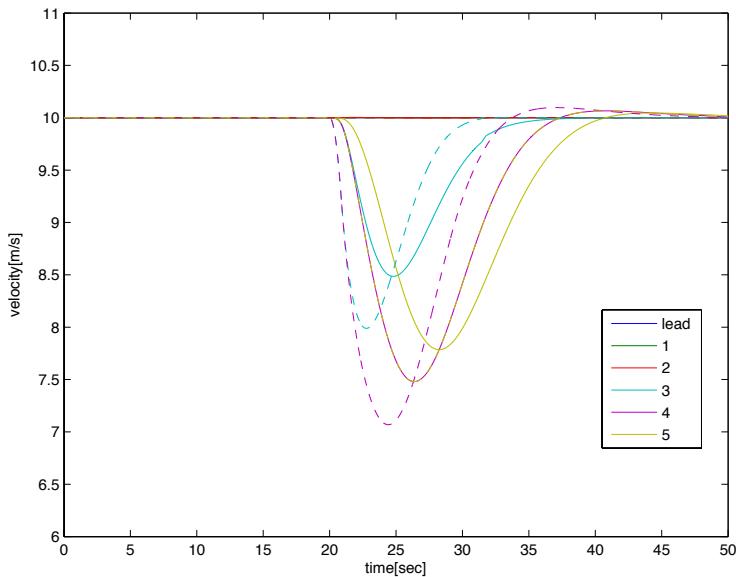


Figure 5.11: Simulation results when a vehicle cuts into the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)

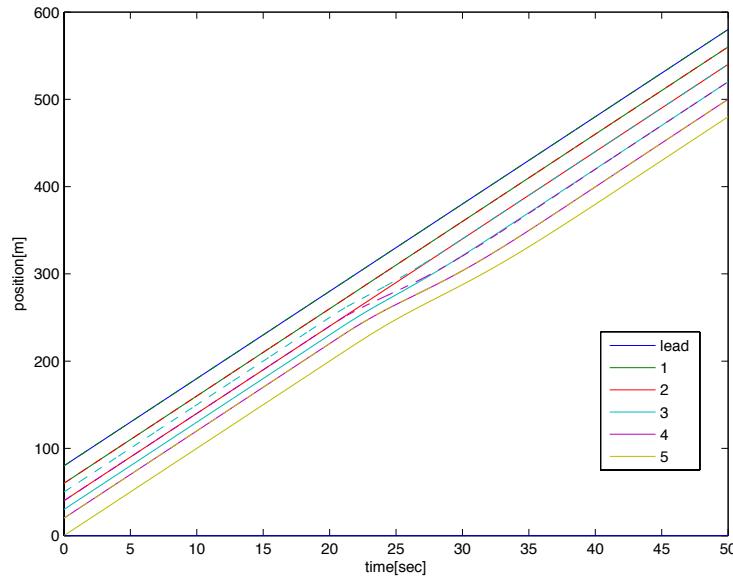


Figure 5.12: Simulation results when a vehicle cuts into the platoon (Solid line: Actual vehicle, Dashed line: Virtual lead vehicle)

membership functions depend on the error of the virtual lead vehicles. However, this scheme makes the control structure complicated while showing little performance improvement. As shown in the previous simulation results, the discontinuities are hardly shown when cutting in or out occurs and will be ignored in this thesis.

5.4 CACC in Lossy Network Condition

Since the CACC system utilizes wireless communication, there is a possibility that some data are lost during the transmission through the wireless network. Recently, with the increase of wireless applications, various research have been done in the area of system control over lossy networks. Some of the research are related to designing state estimators with the data packet loss [46, 14, 62, 49] or the time delay [36, 35] in wireless networks. Also, feedback controllers were designed considering the packet loss between the controller and the actuator [46, 35, 15, 26, 66].

In case of the CACC, the data are transmitted between vehicles in wireless networks. However, the ACC controller in each vehicle is connected to the actuator (or lower level controllers) in a wired manner. So it is reasonable to assume that there is no dropped control command from the controller to the actuator. Also, sensor measurements are transferred to the controller in wired manner inside each vehicle. So some of the data are transferred to the controller without any data loss. In most cases, those data are related to the states of

the lead and the host vehicles and are usually important. Since the data which can be lost in the wireless network are far less important, it can be ignored or be given a minimum level of attention to compensate the data loss. In this section, various control and estimation schemes are suggested and compared by simulations.

5.4.1 Problem Description

In this section, it is assumed that sensors in each vehicle measure the relative distance to the lead vehicle, the velocity of the lead vehicle, and the velocity of the host vehicle itself. The measurements are transmitted to other vehicles in the platoon via wireless network. It is assumed that the data can be lost during the transmission over the wireless network with a fixed probability and it can be represented as:

$$X_{ij}^r(k) = \begin{cases} X_i^s(k) & \text{if } \gamma_{i,j} = 1 \\ n/a & \text{if } \gamma_{i,j} = 0 \end{cases} \quad (5.19)$$

where $X_{ij}^r(k)$ is the data from the i^{th} vehicle received by the j^{th} vehicle, $X_i^s(k)$ is the data sent from the i^{th} vehicle, and $\gamma_{i,j}$ is a Bernoulli random variable with $P(\gamma_{i,j} = 1) = \bar{\gamma}$. It is also assumed that $\gamma_{i,j}$'s are independent for all i 's and j 's. Note that the lost data (n/a) can be distinguished from zero values. For simplicity, a discrete time system is considered and it is assumed that there is no time delay through the communication.

In the simulations, no measurement and process noises are considered.

5.4.2 Control Schemes for Dropped Data

For i^{th} vehicle, the feedback control law is given by:

$$a_{des,i}(k) = -K_i X(k) \quad (5.20)$$

where K_i is the i^{th} row of the feedback gain matrix from the LQ optimal control problem, and $X(k)$ is the state vector given in (5.3). In $X(k)$, $x_{i-1}(k) - x_i(k)$, $v_{i-1}(k)$, and $v_i(k)$ are measured by sensors and other data are transmitted by wireless networks and can be lost. If some of the data are lost, there are various ways to respond.

Case 1 Set the lost data to be zeros.

Case 2 $a_{des,i}(k) = 0$ if $\prod_j \gamma_{i,j} = 0$.

Case 3 $a_{des,i}(k) = \begin{cases} -K_{i,CACC} X(k) & \text{if } \prod_j \gamma_{i,j} = 1 \\ -K_{i,ACC} X(k) & \text{if } \prod_j \gamma_{i,j} = 0 \end{cases}$.

Case 4 Estimate the lost data.

For Case 1, $v_j(k)$ can be transmitted from the j^{th} and $j + 1^{th}$ vehicles to the i^{th} vehicle. $v_j(k)$ is set to be zero only if both $\gamma_{i,j}$ and $\gamma_{i,j+1}$ are zeros. For Case 2, the desired acceleration is set to be zero when any of the data are lost. For Case 3, if any of the data are lost during

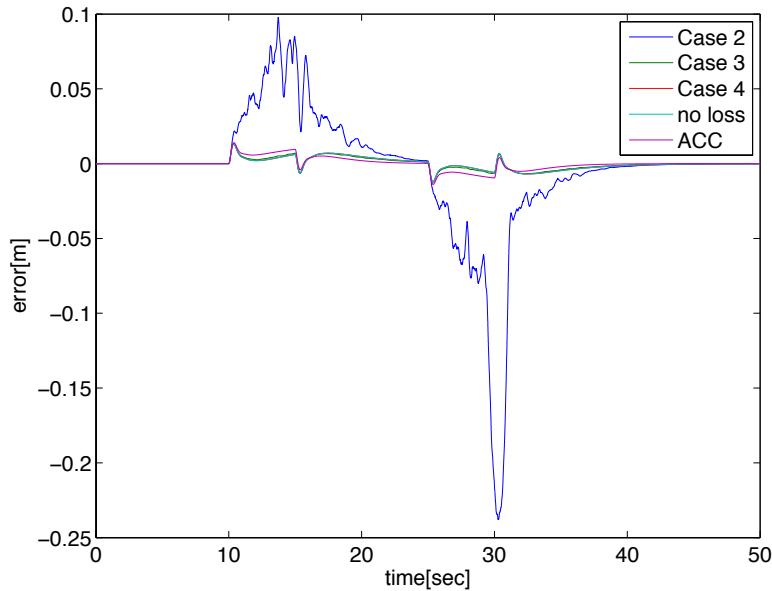


Figure 5.13: Simulation results over lossy network: error for vehicle #1

the transmission, the ACC control law is utilized instead of the CACC. It is possible since all the data used in the ACC control law are measurable with the sensors in the i^{th} vehicle. For Case 4, the state is estimated by $\hat{X}^o(k+1) = (A - BK)\hat{X}(k)$ and $\hat{X}(k+1)$ is updated by replacing the elements of a-priori estimate $\hat{X}^o(k+1)$ with the data measured by sensors or transmitted successfully through the network.

5.4.3 Simulation Results

The simulation set-up is the same with the simulation in Section 5.2. The data loss rate is assumed to be 5%, i.e. $\bar{\gamma} = 0.95$. 4 cases suggested in the previous section are simulated and compared with the CACC without any data loss and with the case where all the vehicles utilizes identical ACC controllers. Case 1 shows the worst performance which is a lot worse than the other cases and the result with Case 1 will not be shown in the figures for better comparison.

Figures 5.13 ~ 5.17 show the simulation results. As shown in the result, Case 2 shows the worst performance. It shows the largest error and also the benefit of the CACC, reacting faster and making the sign of the error to be the opposite of the lead vehicle acceleration, is lost. Case 4, in this case, shows almost the same performance with the CACC system without data loss.

As shown in the figures, Case 3 shows the best performance. The error with Case 3 is even smaller than the case without data loss and also has opposite sign with the lead vehicle

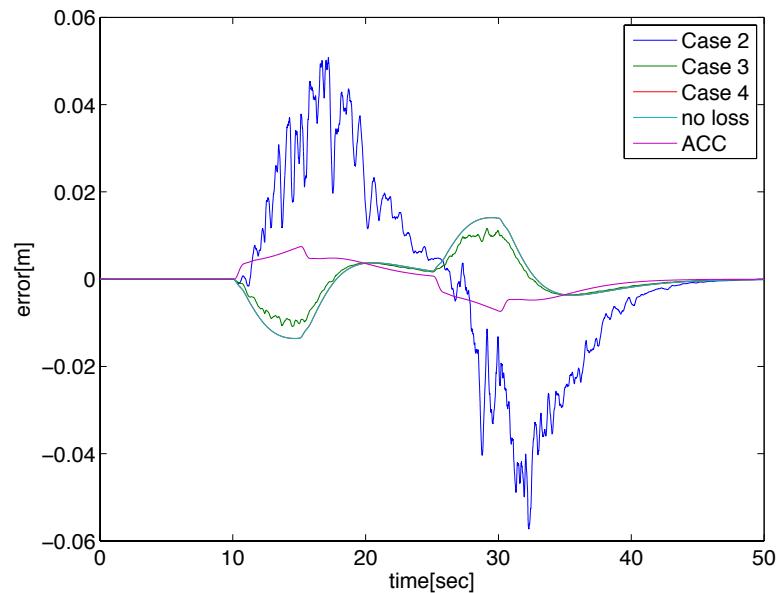


Figure 5.14: Simulation results over lossy network: error for vehicle #2

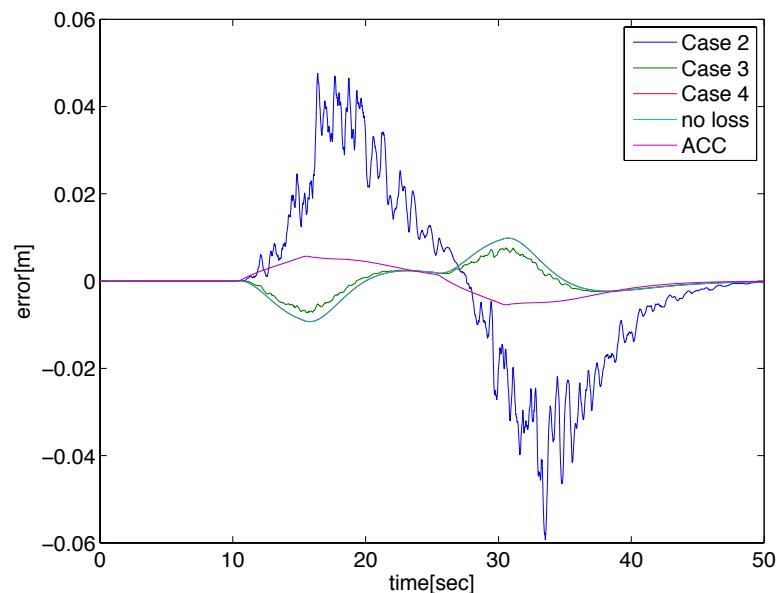


Figure 5.15: Simulation results over lossy network: error for vehicle #3

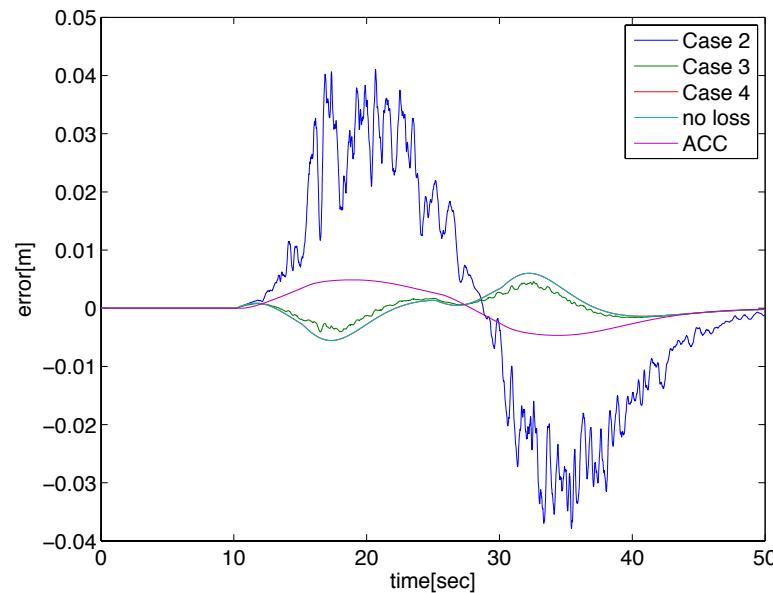


Figure 5.16: Simulation results over lossy network: error for vehicle #4

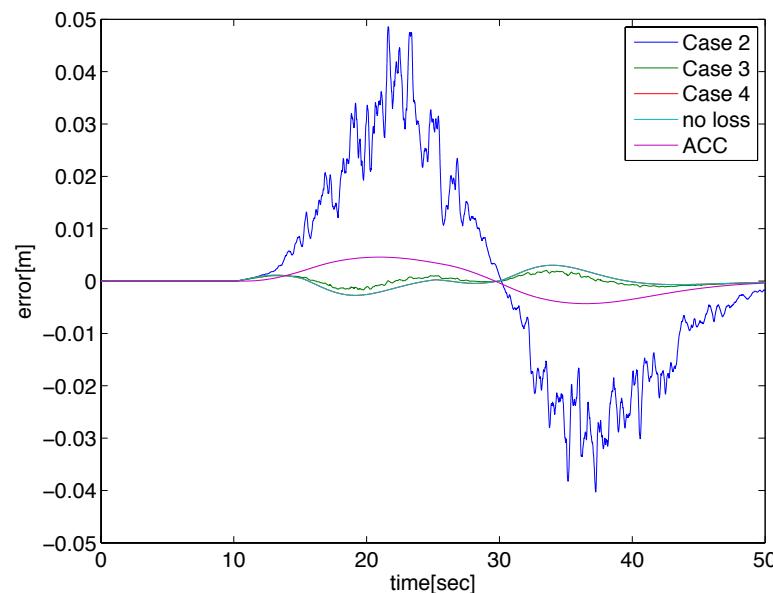


Figure 5.17: Simulation results over lossy network: error for vehicle #5

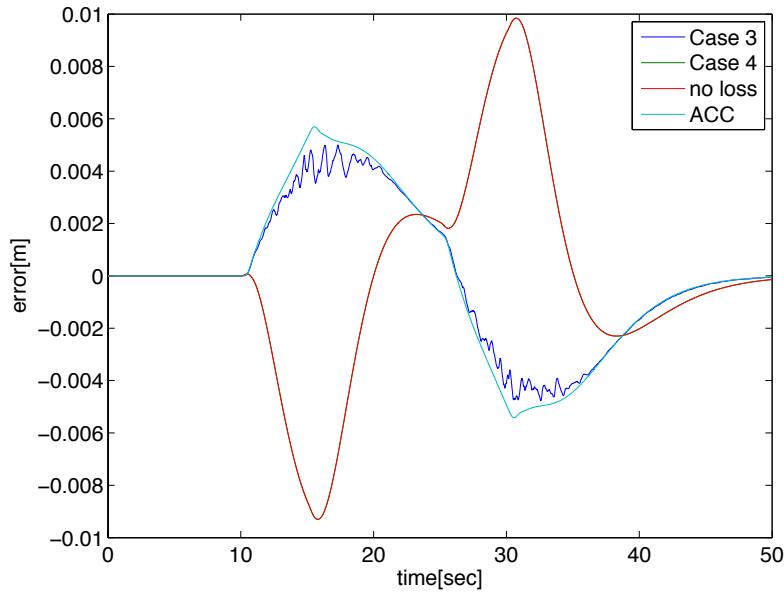


Figure 5.18: Simulation results over lossy network with $\bar{\gamma} = 0.50$: error for vehicle #3

acceleration. As described before, the error with the opposite sign with the lead vehicle acceleration is the benefit of the CACC which considers the errors of the front vehicles in the platoon. In Case 3, the vehicle only utilizes the measurement and does not consider the error in front when the data are lost. This reduces the effect of CACC and fortunately reduces the magnitude of the error.

However it is not guaranteed that the performance of Case 3 is better than CACC without any data loss when the data loss rate is larger. Figure 5.18 shows the simulation result with a worse loss rate is 50% in the simulation. When the loss rate is higher, Case 3 loses the benefit of the CACC system and the error has the same sign with the lead vehicle acceleration while Case 4 still shows similar performance with the CACC without data loss.

5.4.4 Estimation with Kalman Filter

In the previous section, the estimation scheme showed the best performance. However, no measurement noises were considered during the estimation process. Actually, the measured signals are contaminated by noises. Also, the covariance of each noise are different. For example, the velocity of the j^{th} vehicle was assumed to be measured by the j^{th} vehicle with encoders and by the $j+1^{th}$ vehicle with the radar. However, actually the radar measures the relative velocity between the j^{th} and $j+1^{th}$ vehicles and the velocity of the $j+1^{th}$ vehicle should be added to it. By doing that, the velocity measurement from the $j+1^{th}$ vehicle has larger measurement noise than the measurement from the j^{th} vehicle. Kalman filter is one

of the most popular estimation scheme when the data are contaminated by noise. It can consider the different noise covariances and find the optimal state estimation.

Problem Description

The system model is basically the same with the model used in Section 5.2. However, process noises and measurement noises are introduced. Also, as described before, it is assumed that the velocity of the j^{th} vehicle is measured by the j^{th} vehicle with encoders and by the $j+1^{th}$ vehicle with radar. Considering the data loss, the discrete time state space equations are considered as:

$$X(k+1) = AX(k) + BU(k) + B_wW(k) \quad (5.21)$$

$$Y(k) = CX(k) + V(k) \quad (5.22)$$

$$X(k) = \begin{bmatrix} x_l(k) - x_1(k) \\ x_1(k) - x_2(k) \\ \vdots \\ x_{N-2}(k) - x_{N-1}(k) \\ v_l(k) \\ v_1(k) \\ \vdots \\ v_{N-1}(k) \end{bmatrix} \quad (5.23)$$

$$U(k) = [a_l(k) \ a_1(k) \ \cdots \ a_{N-1}(k)]^T \quad (5.24)$$

$$Y(k) = \begin{bmatrix} x_l(k) - x_1(k) \\ \vdots \\ x_{N-2}(k) - x_{N-1}(k) \\ v_1(k) \\ \vdots \\ v_{N-1}(k) \\ v_l(k) \\ \vdots \\ v_{N-2}(k) \end{bmatrix}^T \quad (5.25)$$

$$W(k) = [w_l(k) \ w_1(k) \ \cdots \ w_{N-1}(k)]^T \quad (5.26)$$

$$V(k) = \begin{bmatrix} v_{r,d,1}(k) \\ \vdots \\ v_{r,d,N-1}(k) \\ v_{e,1}(k) \\ \vdots \\ v_{e,N-1}(k) \\ v_{r,v,1}(k) \\ \vdots \\ v_{r,v,N-1}(k) \end{bmatrix} \quad (5.27)$$

$$(5.28)$$

where w_i is the process noises for the i^{th} vehicle. $v_{r,d,i}$, $v_{r,v,i}$, and $v_{e,i}$ are the measurement noises of the distance measurement by the radar, the front vehicle velocity measurement by the radar, and the host vehicle velocity measurement by the encoder in i^{th} vehicle, respectively.

The state space matrices are given as:

$$A = \begin{bmatrix} \Delta t & -\Delta t & 0 & \cdots & 0 \\ I_{(N-1) \times (N-1)} & 0 & \Delta t & -\Delta t & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \Delta t & -\Delta t \\ 0_{N \times (N-1)} & & I_{N \times N} \end{bmatrix} \quad (5.29)$$

$$B = B_w = \begin{bmatrix} \frac{1}{2}\Delta t^2 & -\frac{1}{2}\Delta t^2 & 0 & \cdots & 0 \\ 0 & \frac{1}{2}\Delta t^2 & -\frac{1}{2}\Delta t^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \frac{1}{2}\Delta t^2 & -\frac{1}{2}\Delta t^2 \\ \Delta t I_{N \times N} \end{bmatrix} \quad (5.30)$$

$$C = \begin{bmatrix} I_{(N-1) \times (N-1)} & 0_{(N-1) \times N} \\ 0_{2(N-1) \times (N-1)} & \begin{bmatrix} 0_{(N-1) \times 1} & I_{(N-1) \times (N-1)} \\ I_{(N-1) \times (N-1)} & 0_{(N-1) \times 1} \end{bmatrix} \end{bmatrix} \quad (5.31)$$

In this thesis, the noises are assumed to be stationary zero mean white and independent to each other. Also, the covariances of radar measurement noises are assumed to be larger than the encoder measurement noises.

Kalman Filter

The standard a-posteriori Kalman Filter equations are:

$$\hat{X}(k) = \hat{X}^o(k) + F(k)\tilde{Y}^o(k) \quad (5.32)$$

$$\hat{X}^o(k+1) = A\hat{X}(k) + BU(k) \quad (5.33)$$

$$\tilde{Y}^o(k) = Y(k) - C\hat{X}^o(k) \quad (5.34)$$

$$F(k) = M(k)C^T(CM(k)C^T + S_{vv}(k))^{-1} \quad (5.35)$$

$$\begin{aligned} M(k+1) &= AM(K)A^T + B_wS_{ww}(k)B_w^T \\ &\quad - AM(k)C^T(CM(k)C^T + S_{vv}(k))^{-1}CM(k)A^T \end{aligned} \quad (5.36)$$

where $S_{ww}(k)$ and $S_{vv}(k)$ are the noise covariance matrices for the process and the measurement noises. Since the lead vehicle acceleration is assumed to be zero, the actual lead vehicle acceleration is considered as a process noise. So, $S_{ww}(k)$ is set as a diagonal matrix. The diagonal elements are all the same except the $(1, 1)$ element which is larger than others.

The measurement noise covariance matrix $S_{vv}(k)$ is stationary. However, when some of the data are lost through the wireless communication, the lost data are set to be zeros and the error covariances are set to be very large numbers. For example, $Y_j(k)$, $Y_{N-1+j}(k)$, and $Y_{2(N-1)+j}(k)$ are from the j^{th} vehicle. If the data are lost, the data are set to be zeros and (j, j) , $(N-1+j, N-1+j)$, and $(2(N-1)+j, 2(N-1)+j)$ elements of $S_{vv}(k)$ are set to be very large so that the lost data are not considered in the Kalman Filter.

5.4.5 Simulation Result

Simulations are performed to verify the performance of the Kalman Filter. Simulation results with 4 different methods are compared: (1) KF w/o loss Kalman Filter designed in the previous section is utilized while no data are lost. (2) ACC ACC system is utilized for each single vehicle. (3) CACC w/KF Kalman Filter designed in the previous section is utilized with 5% data loss rate. (4) Case 3 Kalman Filter is utilized when no data are lost. However, ACC system is utilized when data are lost. The loss rate is 5%.

For (2) and (4), when the ACC system is utilized, the relative distance to the lead vehicle and the velocities of the lead and the host vehicles are measured by each vehicle. However, a Kalman Filter with a single lead and a single host vehicle model is utilized to compensate the noises.

Figures 5.19 ~ 5.23 show the simulation results with 5% loss rate. As shown in the result, with the Kalman Filter the performance is very similar to the case when no data are lost. The overall results are similar to the case without noises and for some cases Case 3 shows better performance than the CACC with Kalman Filter. However, similar to the result without noises, it is not guaranteed that the performance of Case 3 is better than CACC with Kalman Filter when the data loss rate is larger. Figure 5.24 shows the simulation result with a worse loss rate. The loss rate is 50% in the simulation. When the loss rate is higher, Case 3 loses the benefit of the CACC system and the performance is even worse than the ACC system.

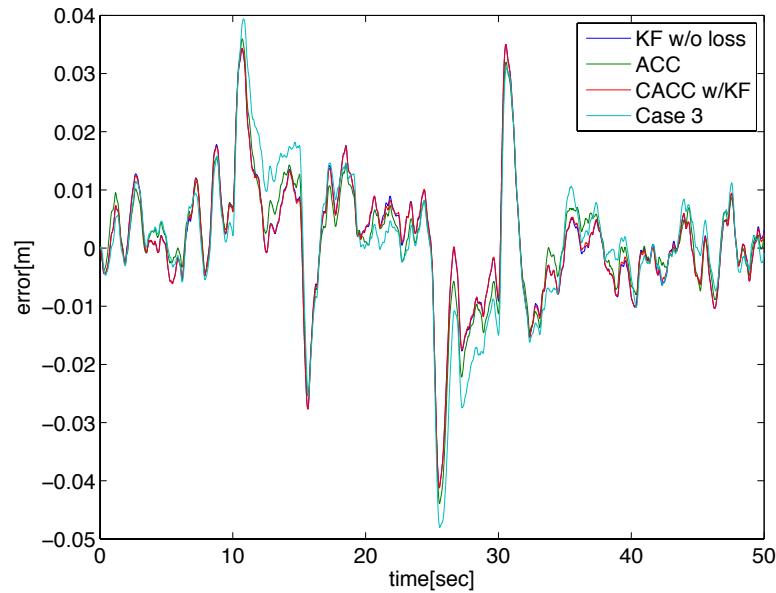


Figure 5.19: Simulation results with Kalman Filter over lossy network: error for vehicle #1

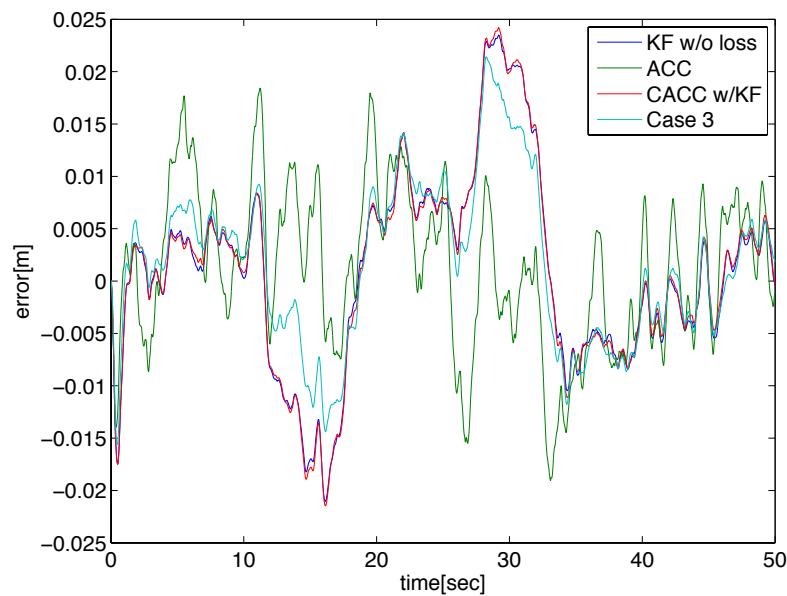


Figure 5.20: Simulation results with Kalman Filter over lossy network: error for vehicle #2

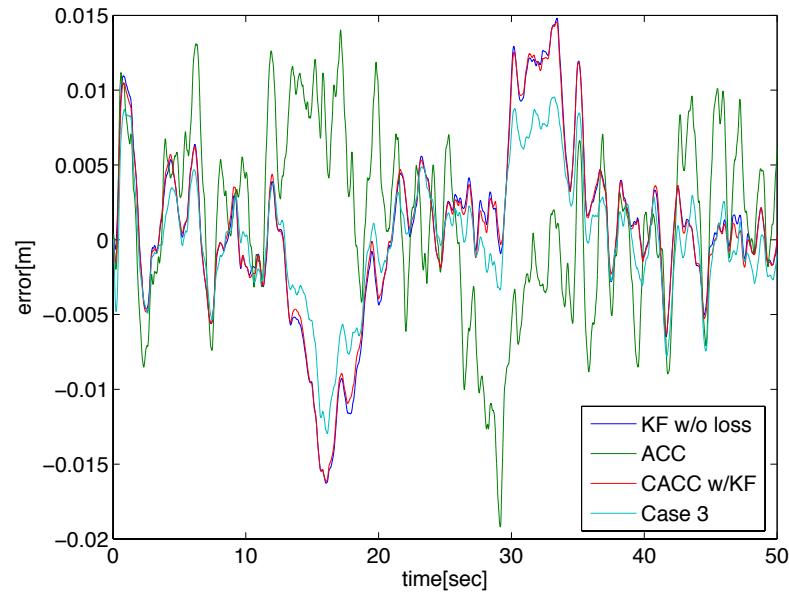


Figure 5.21: Simulation results with Kalman Filter over lossy network: error for vehicle #3

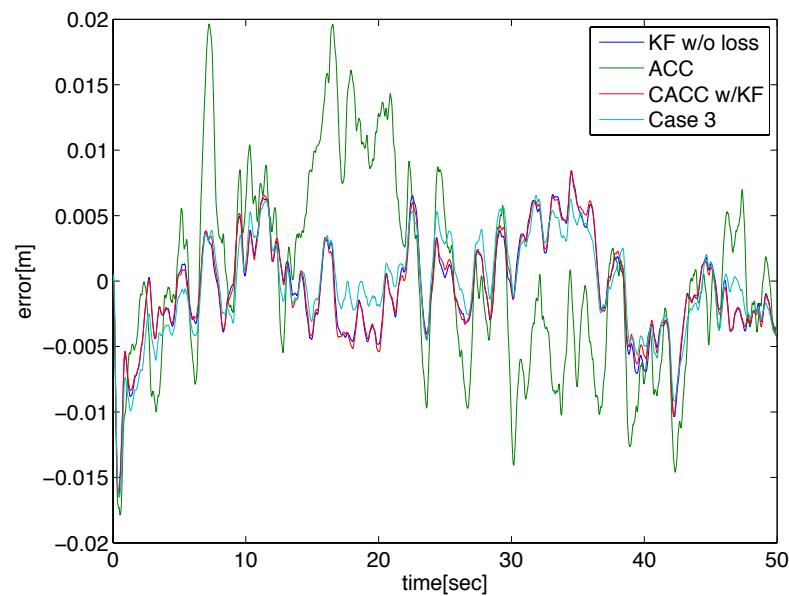


Figure 5.22: Simulation results with Kalman Filter over lossy network: error for vehicle #4

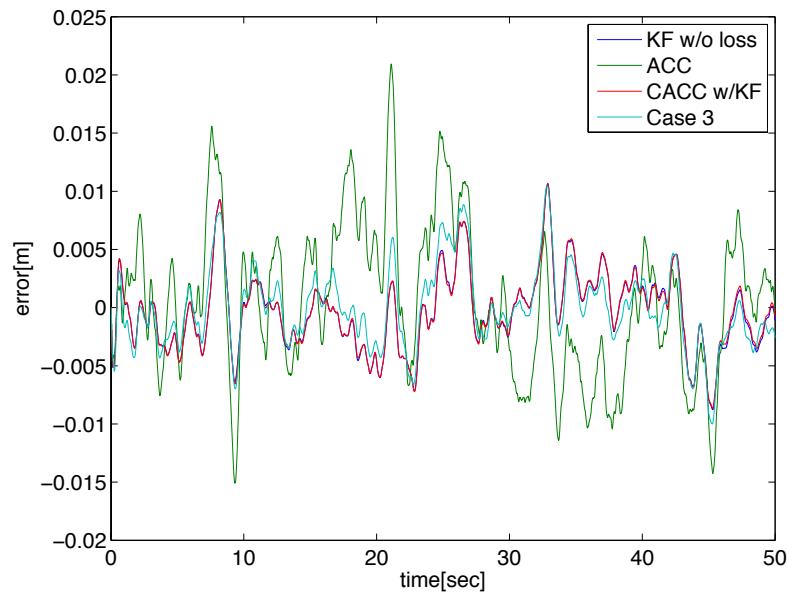


Figure 5.23: Simulation results with Kalman Filter over lossy network: error for vehicle #5

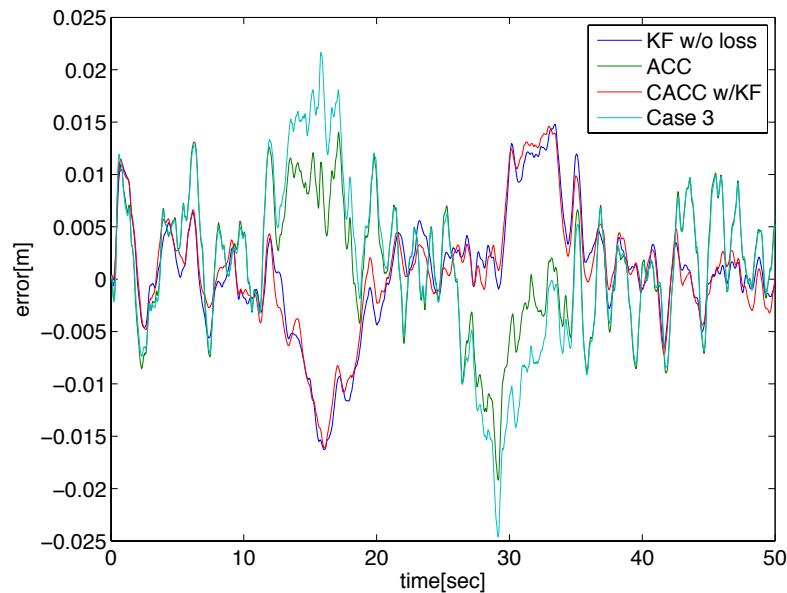


Figure 5.24: Simulation results with Kalman Filter over lossy network with $\bar{\gamma} = 0.50$: error for vehicle #3

5.5 Limitation of the Centralized Controller

The LQ controller designed in the previous sections is a centralized controller. The performance of the LQ controller was good, but the controller may not be practical for implementations. In the LQ controller, it is assumed that the controller have the information about all the vehicles in the platoon. However, the CACC control law is embedded in each vehicle and there is no central control system which can control all the vehicles. Thus, synchronization of the controllers distributed among the vehicles in a platoon is an issue. To utilized the LQ control law, the controllers in each vehicle should have the information such as the length of the platoon and the location of itself in the platoon. Failure of an embedded controller in one vehicle may cause an unacceptable situation. If the length of the platoon is too long, the wireless communication signal may not reach the entire platoon.

For implementations, a decentralized controller is more practical than the centralized control design. The decentralized controller is embedded in each vehicle and may not have all the information of the entire vehicle platoon. However it can utilize the sensor measurements and receive some information from the vehicles inside its communication range. Unlike the centralized LQ controller, the control law of the decentralized design should be independent of the location in the platoon, and it may depend on the parameters set by the driver.

When the decentralized controller is designed, the errors of other vehicles in the platoon are not considered anymore and the overall performance of the vehicle platoon may be worsened with the decentralized controller. String stability must be maintained in decentralized designs. The performance of the centralized control design presented in this section can be utilized as one of the performance standards for the design of a CACC system.

5.6 Summary

In this chapter, the Cooperative Adaptive Cruise Control(CACC) system which utilizes the vehicle-to-vehicle wireless communication was studied. A Linear Quadratic (LQ) optimal control problem was formulated for a platoon of vehicles and solved to find the optimal state feedback control law. The performance of the CACC system were compared to the ACC system via simulations. The CACC system additionally utilized the error of the front vehicles in the platoon and reacted faster to the lead vehicle acceleration. As a result, the error with the CACC system had the same sign with the lead vehicle acceleration which means safer performance when the lead vehicle decelerates and faster when the lead vehicle accelerates.

The virtual lead vehicle scheme was also utilized in case of the CACC system. Without the virtual lead vehicle scheme, the following vehicles accelerated suddenly and also the vehicle in front decelerated to compensate the sudden large step change of the error when a vehicle cuts out from the platoon. Similar result was shown when a vehicle cuts into the platoon. With the virtual lead vehicle scheme, the acceleration or deceleration of the following vehicles were decreased and the front vehicle did not decelerate or accelerate unnecessarily.

Since the wireless network, which is lossy, is utilized in the CACC, control schemes when the data are lost during the transmission were studied. 4 different control schemes were compared via simulations. The scheme in which the ACC control law was utilized when the data were lost and the CACC law was utilized when no data were lost showed the best performance with a low data loss rate. However the performance worsened as the loss rate was increased. When the lost data were estimated from the linear closed loop model, the performance was similar to the case when no data were lost. Also the estimation scheme showed more robust performance when the loss rate was increased. A Kalman Filter was introduced for the case when the measurement data were contaminated by noise and showed similar performance with the case when no data were lost.

Chapter 6

Concluding Remarks and Open Issues

6.1 Concluding Remarks

This thesis suggested and studied several control schemes for the Adaptive Cruise Control systems, including (1) smooth reaction of the host vehicle to the cutting in and out of lead vehicles, (2) real-time optimal profile generation for stop-and-go motions, (3) optimal feedback controller design, and (4) Cooperative Adaptive Cruise Control (CACC) system.

Among the suggested schemes, original contributions of this thesis are (1) virtual lead vehicle scheme, (2) LQ optimal controller design with variable weights, (3) multi-resolution formulation of a Quadratic Programs for real-time applications, and (4) design of a CACC controller considering upstream vehicles. Details are discussed separately below.

6.1.1 Smooth Reaction to Cutting In and Out

The virtual lead vehicle scheme was studied in Chapter 3. It was originally introduced as a method to prevent mode switching between the distance control and the speed control. With a smooth motion control of the virtual vehicle, the scheme could provide smooth transient motion control of the host vehicle when a lead vehicle cuts out or a new lead vehicle cuts in. It was also utilized in Chapter 5 for the CACC system. With the virtual lead vehicle scheme, some undesirable motions were prevented when a vehicle cuts into or out from the platoon of vehicles equipped with identical CACC systems.

6.1.2 Real-time Optimal Profile Generation for Stop-and-Go

Optimal profiles for stopping and starting motions were studied in Chapter 4. The optimization problems were formulated into constrained Quadratic Programs. Interior Point method and Lemke algorithm were utilized to solve the optimization problems. To solve the problems and find the solutions in real time, a multi-resolution formulation was suggested. With the multi-resolution formulation, the size of the problem was decreased and the problem could be solved very fast with the Lemke algorithm. The solutions of the optimization problems

were utilized in the sliding control for the stopping motion and in the virtual lead vehicle scheme for the starting motion.

6.1.3 Optimal Feedback Controller Design

The optimal feedback controllers were designed and used as feedback controllers for both the host and the virtual vehicles. In Chapter 2, a Linear Quadratic (LQ) optimal controller was designed for a single host and a single lead vehicle. In Chapter 3, variable weights were utilized for the LQ optimal controller for the virtual lead vehicle. With the variable weights, the motion of the virtual lead vehicle was controlled to be smooth when there were no safety threats while the motion was still responsive and fast when a dangerous situation occurs. The LQ optimal controller was also applied for a platoon of vehicles equipped with identical CACC systems in Chapter 5.

6.1.4 Cooperative Adaptive Cruise Control

The Cooperative Adaptive Cruise Control system was studied in Chapter 5. An LQ optimal controller was designed. Compared to the general ACC system, the CACC system showed faster and safer performance. The virtual lead vehicle scheme was applied for the case when vehicles cut into or out from the platoon. Since the CACC system utilized wireless communications, lossy network conditions were considered. In lossy network, the estimation scheme showed good and also robust performance compared to the cases where zero output or ACC control law were utilized when the data were lost through the wireless communication. A Kalman Filter was introduced for the case when the measurement signals were contaminated by noise. The noise covariance was set to be very large when the data were lost. The designed Kalman Filter showed similar performance with the case when no data were lost.

6.2 Open Issues

In this thesis, several control schemes have been discussed regarding ACC systems. However, there are some issues that remain and is an area to explore for future work.

6.2.1 Lower Level Control and Brake/Engine Switching Logic

This thesis focused on designing and analyzing the upper level controller. With better lower level controllers, the acceleration of the vehicle body will better follow the desired acceleration from the upper level controller and the performance of the overall system can be improved. Also, if the engine/brake dynamics are considered in the design of the upper level controller, the desired acceleration signal can be determined by the upper level controller so that it can be better followed by the actual engine and brake.

When considering the engine/brake dynamics, an appropriate scheme for switching between the engine and the brake should be studied. Since the ACC system is highly related to the comfort of the passengers, frequent switchings between the engine and the brake are undesirable. A fuzzy controller which can imitate the driver's habit can be a good candidate for the switching logic.

6.2.2 Mixed Traffic CACC

In Chapter 5, the CACC system was studied. However it was assumed that all the vehicles are equipped with identical CACC systems. If the platoon is mixed with vehicle equipped with the CACC system and vehicles driven manually, the performance of the CACC system may decrease or CACC may be not appropriate. However, if the number of manually driven vehicles are relatively smaller compared to the number of vehicles equipped with the CACC systems, there may be enough data to estimate the behaviors of the manually driven vehicles. Then the manually driven vehicles can be modeled and included in the CACC system to enhance the overall performance.

6.2.3 Implementation and Verification

In this thesis, all the performance verifications and evaluations have been done via simulations. Even though the simulator includes realistic models of the vehicle body, the engine, the brake, and the environments, it cannot perfectly emulate the behavior of actual vehicles. To better evaluate the suggested control schemes and to find potential problems that were not considered in this thesis, implementation of the control algorithms and experiments with actual vehicle systems are necessary.

Bibliography

- [1] T. Acarman, Y. Liu, and U. Ozguner. “Intelligent cruise control stop and go with and without communication”. In: *Proceedings of the 2006 American Control Conference* (2006).
- [2] K. Ahn et al. “Microscopic Fuel Consumption and Emission Models”. In: *Proceedings of the 78th Annual Meeting of the Transportation Research Board* (1999).
- [3] B. Arem, C. Driel, and R. Visser. “The impact of cooperative adaptive cruise control on traffic-flow characteristics”. In: *IEEE Transactions on Intelligent Transportation Systems* 7.4 (2006).
- [4] V. Bageshwar, W. Garrard, and R. Rajamani. “Model Predictive Control of Transitional Maneuvers for Adaptive Cruise Control Vehicles”. In: *IEEE Transactions on Vehicle Technology* 53.5 (2004), pp. 1573–1585.
- [5] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.
- [6] Y. Bin et al. “Longitudinal acceleration tracking control of vehicular stop-and-go cruise control system”. In: *Proceedings of the 2004 IEEE International Conference on Networking, Sensing & Control* (2004).
- [7] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge, 2004.
- [8] D. Bruin et al. “Design and test of a cooperative adaptive cruise control system”. In: *IEEE Intelligent Vehicles Symposium* (2004).
- [9] D. Cho and J.K. Hedrick. “Automotive powertrain modeling for control”. In: *ASME Journal of Dynamic Systems, Measurement and Control* 111 (1989), pp. 568–576.
- [10] S.C. Davis, S.W. Diegel, and R.G. Boundy. *Transportation Energy Data Book: Edition 30*. Oak Ridge National Laboratory, 2011.
- [11] Z. Eizad and L. Vlacic. “A control algorithm and vehicle model for stop & go cruise control”. In: *IEEE Intelligent Vehicles Symposium* (2004).
- [12] R. Fletcher. *Practical Methods of Optimization*. Wiley, 1987.
- [13] J. Gerdes. “Decoupled design of robust controllers for nonlinear systems: as motivated by and applied to coordinated throttle and brake control for automated highways”. In: *PhD Dissertation, University of California, Berkeley* (1996).

- [14] V. Gupta et al. “Optimal LQG control across a packet-dropping link”. In: *Systems Control Lett.* (2005).
- [15] C.N. Hadjicostis and R. Touri. “Feedback control utilizing packet dropping network links”. In: *Proc. IEEE Conf. Decision and Control* 2 (2002), pp. 1205–1210.
- [16] S. Ibaraki and M. Tomizuka. “H optimization of fixed structure controller”. In: *Proc. Of the 2000 International Mechanical Engineering Congress and Exposition* (2001).
- [17] S. Ibaraki and M. Tomizuka. “Tuning of a hard disk drive servo controller using fixed-structure H controller optimization”. In: *Journal of Dynamic Systems, Measurement, and Control* 123 (2001), pp. 31–55.
- [18] P. Ioannou and C.C. Chien. “Intelligent Cruise Control”. In: *IEEE Trans. on Vehicular Technology* 42.4 (1993), pp. 657–672.
- [19] S. Kata et al. “Vehicle control algorithms for cooperative driving with automated vehicles and intervehicle communications”. In: *IEEE Transactions on Intelligent Transportation Systems* 3.3 (2002).
- [20] S.G. Kim, M. Tomizuka, and K.H. Cheng. “Mode Switching and Smooth Motion Generation of the Adaptive Cruise Control System by a Virtual Lead Vehicle”. In: *IFAC Symposium on Control in Transportation Systems* (2009).
- [21] S.G. Kim, M. Tomizuka, and K.H. Cheng. “Smooth Motion Control of the Adaptive Cruise Control System with Linear Quadratic Control with Variable Weights”. In: *ASME Dynamic Systems and Control Conference (DSCC)* (2010).
- [22] S.L. Koo, H.S. Tan, and M. Tomizuka. “Impact of Tire Compliance Behavior to Vehicle Longitudinal Dynamics and Control”. In: *American Control Conference* (2007).
- [23] H. Kwakernaak and R. Sivan. *Linear Optimal Control Systems*. New York: Wiley-Interscience, 1972.
- [24] YD. Landau. *System identification and control design*. Englewood Cliffs, NJ: Prentice Hall, 1990.
- [25] J. Laumonier, C. Desjardins, and B. Chaib-draa. “Cooperative adaptive cruise control: a reinforcement learning approach.” In: *in Proceedings of 4th Workshop on Agents in Traffic And Transportation, AAMAS’06* (2006).
- [26] Q. Ling and M. Lemmon. “Optimal dropout compensation in networked control systems”. In: *IEEE Conf. Decision and Control* (2003).
- [27] P. Lingman and B. Schmidtbauer. “Road slope and vehicle mass estimation using Kalman filtering”. In: *Vehicle System Dynamics* 37 supp. (2003), pp. 12–23.
- [28] J.J. Martinez and C. Canudas-de Wit. “A safe longitudinal control for adaptive cruise control and stop-and-go scenarios”. In: *IEEE Transactions on Control Systems Technology* 15.2 (2007), pp. 246–258.

- [29] J.P. Maurice and H.B. Pacejka. "Relaxation length behavior of tyres". In: *Vehicle system dynamics* 27 supplement (1997).
- [30] T. Nagata and M. Tomizuka. "Engine torque control based on discrete event model and disturbance observer". In: *Proceedings of the ASME Int. Mech. Eng. Congr. Expo.* (2007).
- [31] T. Nagata and M. Tomizuka. "Robust engine torque control by discrete event disturbance observer". In: *Proceedings of the 17th IFAC World Congress* (2008).
- [32] J.E. Naranjo et al. "ACC+Stop&Go maneuvers with throttle and brake fuzzy control". In: *IEEE Transactions on Intelligent Transportation Systems* 7.2 (2006).
- [33] G. Naus et al. "Cooperative adaptive cruise control, design and experiments". In: *American Control Conference* (2010).
- [34] G. Naus et al. "Explicit MPC design and performance evaluation of an ACC Stop-&-Go". In: *Proceedings of the 2008 American Control Conference* (2008).
- [35] J. Nilsson. "Real-time control systems with delays". In: *Ph.D. dissertation, Dept. Automatic Control, Lund Inst. Technol.* (1998).
- [36] J. Nilsson, B. Bernhardsson, and B. Wittenmark. "Stochastic analysis and control of real-time systems with random time delays". In: *Automatica* 34.1 (1998), pp. 57–64.
- [37] K. Ohnishi. "A new servo method in mechatronics". In: *Transaction of Japanese Society of Electrical Engineering* 107-D (1987), pp. 83–86.
- [38] H. Ohtsuka and L. Vlacic. "Stop & Go vehicle longitudinal model". In: *IEEE 5th International Conference on Intelligent Transportation Systems* (2002).
- [39] H.B. Pacejka. *Tire and vehicle dynamics*. Oxford: Society of Automotive Engineers / Butterworth-Hienemann, 2002.
- [40] Z. Papp, C. Brown, and C. Bartels. "World modeling for cooperative intelligent vehicles". In: *IEEE Intelligent Vehicles Symposium* (2008).
- [41] M. Persson et al. "Stop & Go controller for adaptive cruise control". In: *Proceedings of the 2004 IEEE Internation Conference on Control Applications* (1999).
- [42] R. Rajamani. *Vehicle Dynamics and Control*. Springer, 2006.
- [43] R. Rajamani and S.E. Shladover. "An experimental comparative study of autonomous and co-operative vehicle-follower control systems". In: *Journal of Transportation Research, Part C Emerging Technologies* 9.1 (2001), pp. 15–31.
- [44] A. Said and W.A. Pearlman. "An image multiresolution representation for lossless and lossy compression". In: *IEEE Transaction on Image Processing* 5.9 (1996).
- [45] J. Schakel, B. Arem, and B. Netten. "Effects of cooperative adaptive cruise control on traffic flow stability". In: *13th International IEEE Annual Conference on Intelligent Transportation Systems* (2010).

- [46] L. Schenato et al. "Foundations of Control and Estimation Over Lossy Networks". In: *Proceedings of IEEE* 95.1 (2007), pp. 163–187.
- [47] S. Sheikholeslam and C.A. Desoer. "Longitudinal control of a platoon of vehicles". In: *Proc. of 1990 American Control Conference* (1990), pp. 291–296.
- [48] J.-J. E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1991.
- [49] S. Smith and P. Seiler. "Estimation with lossy measurements: Jump estimators for jump systems". In: *IEEE Trans. Autom. Control* 48.12 (2003), pp. 1453–1464.
- [50] H. Stearns, S. Mishra, and M. Tomizuka. "Iterative tuning of feedforward controller with force ripple compensation for wafer stage". In: *Advanced Motion Control, 2008. 10th IEEE International Workshop* (2008).
- [51] Y. Takae et al. "A study of drivers' trust in a low-speed following system". In: *SAE Technical Paper* doi:10.4271/2005-01-0430 (2005).
- [52] T. Takizawa et al. "Load measuring device for rolling bearing unit and load measuring rolling bearing unit". In: *U.S. Patent 7 320 257* (2008).
- [53] C.K.W. Tam and K.A. Kurbatskii. "Multi-size-meshmulti-time-step dispersion-relation-preserving scheme for multiple-scales aeroacoustics problems". In: *International Journal of Computational Fluid Dynamics* 17 (2003), pp. 119–132.
- [54] H.S. Tan, R. Rajamani, and W.B. Zhang. "Demonstration of an automated highway platoon system". In: *Proc. of 1998 American Control Conference* 3 (1998), pp. 1823–1827.
- [55] T. Umeno and Y. Hori. "Robust speed control of DC servomotors using modern two degrees of freedom controller design". In: *IEEE Transactions on Industrial electronics* 38.5 (1991), pp. 363–368.
- [56] A. Vahidi and A. Eskandarianm. "Research Advances in Intelligent Collision Avoidance and Adaptive Cruise Control". In: *IEEE Transactions on Intelligent Transportation Systems* 4.3 (2003), pp. 143–153.
- [57] A. Vahidi, A. Stefanopoulou, and H. Peng. "Recursive least squares with forgetting for online estimation of vehicle mass and road grade: theory and experiments". In: *Vehicle System Dynamics* 43.1 (2005), pp. 31–55.
- [58] P. Venhovens, K. Naab, and B. Adiprasito. "Stop and Go Cruise Control". In: *International Journal of Automotive Technology* 1.2 (2000), pp. 61–69.
- [59] V. Winstead and V. Kolmanovsky. "Estimation of road grade and vehicle mass via model predictive control". In: *Proceedings of the 2005 IEEE Conference on Control Applications* (2005).
- [60] J.Y. Wong. *Theory of ground vehicles*. 2nd. John Wiley and Sons, Inc., 1993.
- [61] Q. Xu et al. "Effects of vehicle-vehicle/ roadside-vehicle communication on adaptive cruise controlled highway systems". In: *IEEE 56th In Vehicular Technology Conference, 2002. Proceedings. VTC 2002-Fall* 2 (2002), pp. 1249–1253.

- [62] Y. Xu and J. Hespanha. “Estimation under controlled and uncontrolled communications in networked control systems”. In: *Proceedings of IEEE Conference Decision and Control* (2005), pp. 842–847.
- [63] K. Yi, J. Hong, and Y.D. Kwon. “A Vehicle Control Algorithm for Stop-and-Go Cruise Control”. In: *Journal of Automobile Engineering, Proceedings of the Institution of Mechanical Engineers* 1.215 (2001), pp. 1099–1115.
- [64] K. Yi et al. “Implementation and vehicle tests of a vehicle stop-and-go cruise control system”. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 216.7 (2002), pp. 537–544.
- [65] N. Yoshitani and A. Hasegawa. “Model-based control of strip temperature for the heating furnace in continuous annealing”. In: *IEEE Transactions on Control Systems Technology* 6 (1998), pp. 146–156.
- [66] N. Yu et al. “Stabilization of networked control system approach”. In: *IEEE Int. Symp. Computer Aided Control Systems Design* (2004), pp. 362–367.