Real-time Autonomous Cruise Control of Connected Plug-in Hybrid Electric Vehicles under Uncertainty

by

Bijan Sakhdari

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2018

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Ya-Jun Pan Professor, Dept. of Mechanical Engineering, Dalhousie University
Supervisor:	Nasser L. Azad Associate Professor, Dept. of Systems Design Engineering, University of Waterloo
Internal Member:	John McPhee Professor, Dept. of Systems Design Engineering, University of Waterloo
Internal Member:	Baris Fidan Associate Professor, Dept. of Mechanical Engineering, University of Waterloo
Internal-External Member:	Soo Jeon Associate Professor, Dept. of Mechanical Engineering, University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

I declare that all the designs and evaluations of the controllers presented in this thesis are my work and were developed and performed by me.

This thesis contains some material that was taken from multi-author papers including:

- B. Sakhdari and N. L. Azad, "Adaptive tube-based nonlinear MPC for autonomous cruise control of plug-in hybrid electric vehicles", IEEE Transactions on Vehicular Technology (In press), 2018.
- B. Sakhdari and N. L. Azad, "A distributed reference governor approach to ecological cooperative adaptive cruise control," in IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 5, pp. 1496-1507, May 2018.
- B. Sakhdari and E. Moradi Shahrivar and N. L. Azad, " Robust tube-based MPC for automotive adaptive cruise control design,"2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSc), Yokohama, Japan, 2017.

  Contribution of E. Moradi Shahrivar: collaboration in developing the MPC controller
- B. Sakhdari and M. Vajedi and N. L. Azad, "Ecological adaptive cruise control of a plug-in hybrid electric vehicle for urban driving," 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSc), Rio de Janeiro, 2016.

  Contribution of M. Vajedi: collaboration in developing the NMPC controller

Abstract

Advances in embedded digital computing and communication networks have enabled the development of automated driving systems. Autonomous cruise control (ACC) and cooperative ACC (CACC) systems are two popular types of these technologies, which can be implemented to enhance safety, traffic flow, driving comfort and energy economy. This PhD thesis develops robust and adaptive controllers for plug-in hybrid electric vehicles (PHEVs), with the Toyota Plug-in Prius as the baseline vehicle, in order to enable them to perform safe and robust car-following and platooning with improved vehicle performance.

Three controllers are designed here to achieve three main goals. The first goal of this thesis is the development of a real-time Ecological ACC (Eco-ACC) system for PHEVs, that is robust to uncertainties. A novel adaptive tube-based nonlinear model predictive control (AT-NMPC) approach to the design of Eco-ACC systems is proposed. Through utilizing two separate models to define the constrained optimal control problem, this method takes into account uncertainties, modeling errors and delayed data in the design of the controller and guaranties robust constraint handling for the assumed uncertainty bounds. In addition, it adapts to changes in order to improve the control performance when possible. Furthermore, a Newton/GMRES fast solver is employed to implement the designed AT-NMPC in real-time. The second goal is the development of a real-time Ecological CACC (Eco-CACC) system that can simultaneously satisfy the frequency-domain and time-domain platooning criteria. A novel distributed reference governor (RG) approach to the constraint handling of vehicle platoons equipped with CACC is presented. RG sits behind the controlled string stable system and keeps the output inside the defined constraints. Furthermore, to improve the platoon's energy economy, a controller is presented for the leader's control using NMPC method, assuming it is a PHEV. The third objective of this thesis is the control of heterogeneous platoons using an adaptive control approach. A direct model reference adaptive controller (MRAC) is designed that enforces a string stable behavior on the vehicle platoon despite different dynamical models of the platoon members and the external disturbances acting on the systems. The proposed method estimates the controller coefficients on-line to adapt to the disturbances such as wind, changing road grade and also to different vehicle dynamic behaviors. The main purpose of all three controllers is to maintain the driving safety of connected vehicles in car-following and platooning while being real-time implementable. In addition, when there is a possibility for performance enhancement without sacrificing safety, ecological improvement is also considered.

For each designed controller, Model-in-the-Loop (MIL) simulations and Hardware-in-the-Loop (HIL) experiments are performed using high-fidelity vehicle models in order to validate controllers' performance and ensure their real-time implementation capability.

Acknowledgements

This thesis would not have been possible without the generous support, motivation and great encouragement of my supervisor Professor Nasser L. Azad. His guidance helped me in all the time of research and also writing of this thesis. I have been lucky to have had him as my supervisor and I truly could not have imagined having a better advisor and mentor for my PhD study.

My special thanks to my PhD examination committee, Professors John McPhee, Baris Fidan, Soo Jeon and Ya-jun Pan, for their valuable time, attention, and also their insightful comments and hard questions that helped me widen my research from previous perspectives.

I thank my fellow labmates in Smart Hybrid and Electric Vehicles (SHEVS) Laboratory, with special thanks to Dr. Mahyar Vajedi who helped me push my research ahead whenever I was having difficulties.

I thank Dr. Ken Butts and Toyota Technical Center for their suggestions which greatly assisted the research.

I acknowledge Toyota Technical Centre, Natural Sciences and Engineering Research Council (NSERC) of Canada and the Ontario Centres of Excellence for financial support of this research.

I would like to thank my family: my parents Hossein Sakhdari and Fatemeh Sakhdari, whose love and guidance are with me in whatever I pursue. I also want to thank my brother, Saeed Sakhdari, my sister, Hanieh Sakhdari and also Ali, Jana, and Artin F. Rad, for supporting me spiritually throughout writing this thesis and my life in general.

Last but definitely not least, I must express my very profound gratitude and love to Kiana Amini, for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of research and writing this thesis.

Dedication

This is dedicated to the one I love.

Table of Contents

List of Tables

List of Figures

Nomenclature

η_g	Electric generation efficiency
η_i	Actuation time constant
η_m	Electric actuation efficiency
μ_{r_0}	Static rolling resistance coefficient
μ_{r_v}	Dynamic rolling resistance coefficient
ω_e	Angular velocity of engine
ω_g	Angular velocity of generator
ω_m	Angular velocity of motor
ρ_a	Air density
τ_a	Actuation delay
A_a	Vehicle's frontal area
a_h	Host vehicle's acceleration
a_p	Preceding vehicle's acceleration
C_d	Drag coefficient
C_e	Electricity cost
C_f	Fuel cost

e_p	Position error
e_v	Velocity error
E_{max}	Maximum battery energy
F_r	Resistance forces
h	Headway time constant
$I_m, I_r, I_g, I_s, I_e, I_c$	Inertial coefficient
K_i	Actuation gain
m_f	Fuel mass
P_g	Generator power
P_m	Eletric motor
P_{bat}	Battery power
PR	Power ratio
R_g	Gear ratio
r_w	Wheel radius
R_{bat}	Battery internal resistance
SOC	State of charge
T_e	Engine torque
T_g	Electric generator torque
T_m	Electric motors torque
T_w	Wheel torque
V_0	Open cicuit voltage
v_h	Host vehicle's velocity
v_p	Preceding vehicle's velocity

Chapter 1

Introduction

1.1 Background

The ever-increasing need for fast and safe transportation for the growing world population, air pollution from the transportation sector and also limited available sources of fossil fuel are major concerns worldwide. The increased number of vehicles has resulted in more congested traffic, accompanied by a higher risk of accidents. Earth's current population is nearly 7 billion and continuing to grow at a high rate. The high population growth rate and developments in economies demand more efficient and safe transportation systems. Currently, there are about 1 billion vehicles in the world, a number that with the current demand for personal transportation, is expected to double in the next few decades [1]. This rapid growth in the number of vehicles has led to high congested traffic which increases driving risk and demands a higher effort from the driver to drive safely. Each year, around 1.3 million people die in car accidents and, by the current growing rate of vehicle ownership, this number is expected to increase to about 1.9 million per year [2]. Moreover, transportation is currently responsible for 22% of the total anthropogenic global greenhouse emissions [3]. These emissions have doubled since 1970 and road vehicles are responsible for 80% of it. If this growth continues at the current rate, emissions will be doubled by 2050 which makes the transportation sector the largest producer of greenhouse gases [1, 4]. These issues have encouraged governments, industry, and academia to come up with different ways to achieve more efficient, green and safe transportation systems.

Consequences of air pollutants from the transportation sector have forced governments to apply harsh emission standards such as the American Corporate Average Fuel Economy

(CAFE), whose purpose is to reduce energy consumption by increasing fuel economy. Each manufacturer must pay a penalty fee if fail to meet the standard of the given model year [5]. These laws force car manufacturers to investigate different techniques to increase their fleet's fuel economy.

Electrification was introduced as a way to achieve sustainable transportation with high fuel efficiency and minimal emissions [6]. Battery Electric Vehicles (BEVs) produce zero emission as they use a battery as their only energy source. Also, with 80% efficiency, they offer a lower energy cost compared to the conventional vehicles with about 30% efficiency. However, despite the very opportunities that lie in BEVs, there is an unwillingness towards purchasing them due to their smaller size, shorter range, higher price and longer refueling time [1].

Hybrid Electric Vehicles (HEVs) are another solution for improving fuel efficiency and lowering emissions. They combine a main engine with electric motors and batteries to improve their efficiency and regenerate wasted energy from braking or in down-hill slopes. PHEVs are another version of HEVs with larger batteries that can be fully charged from the grid before starting the vehicle. The larger batteries enable them to perform as an electric vehicle, with an engine reserved to extend their range when needed. They have much better fuel economy and also produce lower emissions by getting energy from their two different sources.

In parallel, advances in embedded digital computing and communication networks, along with the development of lower cost on-board vehicle sensors, have enabled the development of automated driving systems. Advanced Driver Assistance Systems (ADAS) is one of the most important outcomes of these developments. The purpose of ADAS technologies is to sense the vehicle's environment and assist the driver according to the changes in the environment [7]. ADAS technologies can improve safety by reducing the effect of error in human judgment in emergency situations and enhance the driving performance by making optimal decisions in autonomous interaction with the environment. Autonomous cruise control (ACC) and Cooperative ACC (CACC) are two of the most promising ADAS technologies. In these systems, radar and or Vehicle to Vehicle (V2V) communications give the inter-vehicular distance and speed of the preceding vehicle(s) to the controller of the autonomous system. The controller will adjust the vehicle's speed to keep a minimum safe distance ahead in ACC and to enable a string of vehicles to follow each other in very close inter-vehicular distances, in CACC. These technologies are considered key components for any future intelligent autonomous vehicle [7]. The two-way wireless communication environment provided by the V2V and Vehicle to Infrastructure

(V2I) communications enables the development of a connected vehicles environment. To improve its performance, a connected vehicle uses wireless technologies to communicate with other vehicles, infrastructure, and other on-board sensors. A connected vehicle is able to collect previously unobtainable and accurate traffic data such as other vehicles' maneuvers, trajectories, destinations and road data [8]. With more information about its environment, the connected vehicles control system can make optimal decisions to improve its performance.

As explained above, ACC allows close car-following, which is the ability to follow a preceding vehicle safely with no input from the driver. CACC goes one step further by providing vehicle platooning, which is the ability to create strings of vehicles that move cooperatively and safely with short inter-vehicular distances. These technologies will enhance driving safety and performance. Close car-following and platooning would result in less drag and better fuel economy for the follower vehicles. Authors in [8] showed that platooning can decrease fuel consumption of heavy-duty vehicles up to 7.7%. Also, platooning results in reduced necessary action and judgment from the driver, which result in safer travel with more comfort for the driver. Moreover, close driving increases the capacity of the roads by minimizing necessary gaps between vehicles. These benefits will result in higher safety and reduced fuel consumption, emissions, traffic congestions, travel time and also less demand for the construction of more roads.

The benefits of vehicle electrification, ADAS, and connected vehicles technologies have captured the attention of many researchers from academia and also industry. These technologies can enable vehicle manufacturers to improve the safety of their vehicles, enhance driving performance and reduce fuel consumption and pollutions caused by their fleet to maintain their annual average fuel consumption allowed by the government standards. To be able to take advantage of these technologies, intelligent vehicle control systems should be developed that can work with such new and advanced technologies.

1.2 Motivation and challenges

Although these technologies offer significantly improved performances, issues like communication network imperfections and also the model and measured data uncertainty have prevented their practical applications. These shortcomings become more concerning when dealing with safety, especially in close car-following and platooning that rely heavily on the data received from the connected vehicles environment. Network imperfections and data uncertainties can result in low performance, unsafe driving and even accidents.

To guarantee the safety of car-following and platooning of connected vehicles, controllers that can handle network imperfections, data uncertainties, and modeling errors must be designed. To be able to achieve a higher performance and maintain passenger's safety and comfort, the designed controller must have robust stability and performance against the mentioned flaws.

A platoon of vehicles must satisfy many requirements to be safe for practical implementations. String stability is the most discussed issue in platooning which is defined in the frequency-domain. Other than string stability, safety, comfort and actuation constraints must also be satisfied. Since constraints are defined in the time-domain, satisfying both string stability, and constraints handling in a single controller is not possible. Therefore, a method is required that can handle both of these conditions in a vehicle platoon.

PHEVs are increasingly considered a solution for sustainable transportation. They demonstrate higher performance compared to other HEVs with both better fuel economy and lower emissions. To integrate PHEVs with the connected vehicles environment and enable them to do platooning and car-following, designed controllers must be able to interact with the local vehicle controllers, especially with the energy management system. Unique and complex design of the PHEVs brings many challenges to the controller design.

Control systems that are able to handle connected vehicles objectives are very complex, which makes their real-time implementation a big challenge. The PHEVs complicated control architecture compared to conventional vehicles and necessary robustness considerations of the designed controllers will definitely increase this burden. This PhD research addresses these challenges by developing robust and adaptive controllers for enabling ACC and CACC for connected PHEVs with real-time implementation capability. These controllers take advantage of radar and V2V communications to improve the performance of the vehicles and enable them to perform safe, close car-following and platooning while enhancing vehicle's energy economy.

1.3 Objective and methods

The goal of this PhD research project is to design and evaluate real-time robust ACC and CACC for connected PHEVs that can maintain safety in short inter-vehicular distances and in addition, when possible and without sacrificing safety, optimize energy cost based on short horizon future trip information. To guarantee system safety, robust control methods have been employed, and in particular robust model predictive control, to consider

modeling errors and data uncertainties in control design. The designed cruise controllers are for real-time implementation, therefore fast optimization solvers have been used to improve the computational cost of the designed controllers. To evaluate the potential of practical implementation, the controllers are fine-tuned for Plug-in Toyota Prius for evaluations and validation purposes, specifically HIL testing. This research project has three main contributions in control design for real-time implementation on connected PHEVs:

1. Design and evaluation of an adaptive tube-based nonlinear model predictive controller (AT-NMPC) for robust and adaptive Eco-ACC;

2. Design and evaluation of a distributed reference governor (RG) for Eco-CACC of connected vehicles; and

3. Design and evaluation of a direct model reference adaptive controller (MRAC) for platooning of heterogeneous vehicle platoons.

1.3.1 Modeling

Modeling is a necessary step for both control design and control evaluation. To design model-based controllers, a control-oriented model with low complexity that can capture the general behavior of the system is required. A high-fidelity model is essential to evaluate the performance of our system on a close to the real model of the vehicle. Especially, high-fidelity models are useful in evaluating fuel consumption. Different models have been used in this thesis for the design and evaluation of controllers which include:

1. **High-fidelity models:** We have used a high-fidelity model for the Prius PHEV, developed in Autonomie, which is a MATLAB-based automotive modeling software. Autonomie is developed by Argonne National Laboratory and it is very popular in the industry for Model-in-the-Loop (MIL) and Hardware-in-the-Loop (HIL) simulations. The energy management and some other sub-models of the Autonomie model have been modified or replaced by controllers that have been previously generated in our research group [9]. In the last chapter, similar high-fidelity models from the Autonomie software were generated for different vehicles to construct a heterogeneous vehicle platoon.

2. **Control-oriented model:** This model has low complexity compared to the high-fidelity model but it captures the general behavior of the considered dynamic system. This simplification allows us to use it at the heart of our model-based controllers

which need simple models with low computational cost. Depending on the application and the type of our problem, the created control-oriented model might be linear or nonlinear. These models have been validated based on the available high-fidelity model so that they will be descriptive enough to capture the general behavior of the system but simple so that they will be appropriate for real-time control.

3. **Reduced model**: Parametric or reduced models are used in parameter estimation steps. The real parameters of the model get lumped into single parameters to make a simpler model for adaption. The input and output of this model are available through measurements and the parameters get estimated online using parameter adaptation methods.

1.3.2 Control design

The main objectives we seek in control design are enabling safe close car-following and platooning and improving energy cost economy while maintaining stability, string stability, robustness, driving comfort, and ensuring real-time implementation capability. Effects of uncertain and delayed data, and modeling errors have been considered in the design of the controllers to guarantee robust stability and performance of the system. We have taken advantage of the AT-NMPC method to design our Eco-ACC. This method takes into account uncertainties to achieve robustness while adapting to them to improve the performance of the system. Although, AT-NMPC can significantly improve energy economy and also robustly maintain safety of the system, it can not enforce frequency-domain requirements. Therefore, this controller can be used for control of a vehicle platoon's leader but it can not be used for vehicle platooning to achieve string stability and constraint handling at the same time.

For constraint handling in platooning, we have designed a distributed predictive RG to enforce constraints on a string stable platoon. This method maintains the behavior of the whole system and only intervenes when there is a possibility of constraint violation. This way frequency-domain and time-domain requirements can be achieved simultaneously. This controller design is based on homogeneity assumption.

To extend the given design in the previous steps to heterogeneous cases, control of heterogeneous platoons is considered by employing MRAC approach. This method takes advantage of an online estimation method to estimate control parameters such that a reference string stable model is enforced on the system. Moreover, this method adapts

to the changes in the system to maintain string stability despite dynamic difference and changes in the environment.

1.3.3 Control evaluation

The goal of control evaluation is to determine the performance of designed controllers on complex and accurate models of the system. This step will validate the proposed designs in different scenarios and also on practical control hardware. The following strategies have been followed to evaluate the designed controllers:

- Performance of the designed controllers have been compared against currently available designs in the literature;
- MIL simulations have been performed to evaluate the newly designed controllers on the high-fidelity model and in different traffic scenarios;
- HIL experiments have been performed to evaluate the real-time implementation capability of the devised controllers on an embedded vehicle system with limited computational power.

1.4 List of contributions

The research contributions can be summarized as follows:

- Developed a real-time, adaptive and robust tube-based nonlinear model predictive controller for ecological autonomous cruise control of PHEVs.
- Developed a real-time, distributed reference governor for constraint handling of connected PHEVs.
- Developed a real-time direct MRAC for heterogeneous platooning.

1.5 Thesis layout

In this chapter, we presented a background on the existing opportunities in electrifications, ADAS, and connected vehicle technologies. Then, the motivation and challenges of the presented PhD research were explained along with a brief summary of objectives and methods. In the end, the contributions of this PhD research were summarized. The rest of this thesis is organized as follows:

Chapter 2 will review state-of-the-art connected vehicles control strategies, especially for robust Eco-ACC, Eco-CACC and heterogeneous platooning. This chapter will address current researches and successful attempts in design and implementation of ACC for close car-following and platooning particularly by considering the robustness issues. Moreover, HIL experimenting is explained in this chapter along with references from the literature which have been performed similar experiments successfully.

Chapters 3 explains an adaptive and robust model-based approach to the controller design of Eco-ACC systems. First, control-oriented and high-fidelity models are explained, which later are used for control design and evaluation. Then the design of a novel AT-NMPC is explained, which can maintain robustness while adapting to changes in the system, through utilizing two different models. The validation of the controllers though MIL and HIL evaluation is also discussed.

Chapter 4 presents a distributed reference governor (RG) approach to the constraint handling of vehicle platoons equipped with CACC. This method and similar approaches have never been used in this context before. In this chapter, first modeling steps are explained. Then, controller design for achieving string stability and constraint handling is presented. The performance of the controller is demonstrated through MIL simulations. HIL experiments have also been carried out to show the real-time capability of this controller.

Chapter 5 presents an adaptive control approach to the platooning of heterogeneous vehicles. First, a nonlinear dynamical model is presented that explains a heterogeneous platoon subjected to external disturbances. Next, a direct model reference adaptive controller (MRAC) is designed that enforces a string stable behavior on the vehicle platoon despite different dynamical models of the platoon members and the external disturbances acting on the systems. The proposed method estimates the controller coefficients online to adapt to the disturbances such as wind, changing road grade and also to different vehicle dynamic behaviors. MIL simulations and HIL experiments show the performance and real-time capability of this controller.

Finally, Chapter 6 discusses conclusions and future work and outlines the contributions of this research project.

Chapter 2

Literature review

As mentioned in the previous chapter, in recent years, the increased amount of harmful emissions, climate change and limited sources of fossil fuel have become a major problem worldwide. Greenhouse gases are considered as the main cause for the global warming and climate changes. There are different human activities and sectors that are responsible for the increased amount of the global emissions, such as: residential, commercial, industrial and most importantly transportation sectors. Transportation is currently responsible for 33% of greenhouse gas emissions in the U.S., which makes it the biggest producer of these gases [10]. Emissions from the transportation sector has increased by more than 20% from 1990 to 2009 while emissions from other sectors have reduced [2]. In parallel, due to the increasing demand for personal transportation, traffic intensity is escalating around the world, which has made traffic congestion a growing issue worldwide . The escalated traffic intensity has resulted in an increased risk of accidents in different parts of the world. The annual cost of road traffic injuries is estimated at $518 billion with human perception error as the most significant factor in 90% of road traffic accidents [11].

A major effort has been done in the past few years by automotive control engineers to use the newly developed technologies and low-cost vehicle sensors to alleviate the above mentioned issues. ADAS are a major outcome of these efforts. These advaced technologies assist drivers on the road in order to improve road transportation safety and comfort for the driver [12]. ACC and CACC are two of the most promising ADAS technologies that have attracted a vast interest from automotive engineers. ACC is an advanced version of cruise controller which, in addition to maintaining a speed set-point, uses an onboard vehicle radar to maintain a safe distance from the preceding vehicle. CACC goes a step further from ACC by utilizing V2I and V2V communications to gather information about

the preceding traffic through these wireless connections. V2V communications can provide more useful data about the immediate preceding vehicle and also other vehicles beyond it with a much lower delay compared to radars [13]. These communications enable vehicles to drive closely with short inter-vehicular distances to construct a vehicle platoon. Platooning and close car-following increase the amount of road throughput and reduce the need for developing more road network.

In this chapter, we will review the current literature on connected vehicles control strategies with an emphasis on the robust Eco-ACC of PHEVs, Eco-CACC for PHEV platooning and heterogeneous vehicles platooning. Furthermore, a background on the methods for control of connected vehicles will be presented. This chapter concludes necessary research for the development of ACC and CACC for connected PHEVs and also reviews significant contributions from the successful design of the proposed control systems.

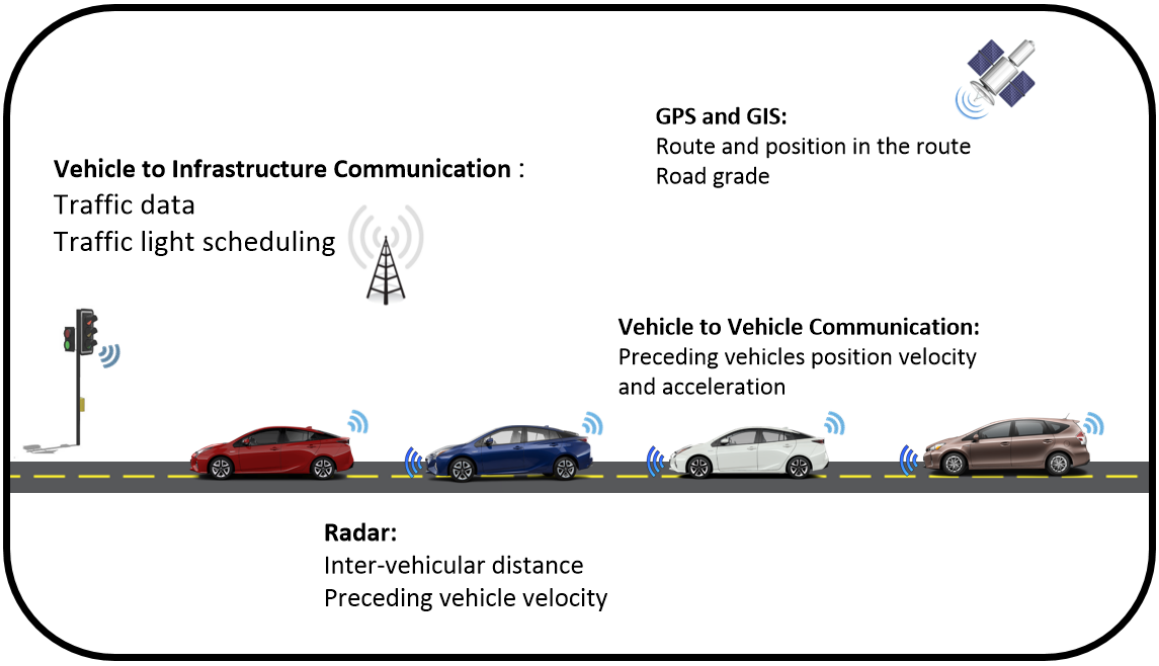


Figure 2.1: Connected vehicle environment and available data sources

2.1 Background on connected vehicles

Each vehicle is part of a larger transportation system that includes other vehicles and their environment. To achieve a higher performance, the vehicle's control system must be provided with information about the larger system to generate optimal maneuvers and make appropriate control decisions. Connected vehicles can take advantage of wireless communications and onboard vehicle sensors to obtain this information. Figure 2.1 shows a connected vehicle with its available data sources. Data from GPS and GIS give the route map, road characteristics and position on the route. These data provide an image of the whole trip and control algorithms can use them for long-term speed trajectory building and energy management. V2I communications provide the route's traffic data and light scheduling, or intersection control algorithm, through connection to the traffic light or the intersection manager. The last and the most important part of a vehicle's environment is its surrounding traffic. A vehicle in motion is most of the times constrained by its surrounding vehicles. Radar and laser sensors (Lidar) provide the distance to the objects ahead and also the inter-vehicular distance and velocity. Current ACCs use radars to avoid accidents by adapting the vehicle's speed when reaching a preceding vehicle. This technology can also enable close car-following. To enhance close following performance, V2V wireless communications could be useful by transmitting acceleration set-points, vehicle specifications and trip data from the preceding vehicle. V2V communications can also provide data from a number of other cars beyond the immediate preceding vehicle, in front and behind of the controlled host vehicle. These technologies enable vehicles to communicate in a network and make a platoon of vehicles that share data and move in very short inter-vehicular distances with high safety and improved performance.

In this PhD research, we are mainly concerned with the development of robust controllers that can keep their performance in the presence of network deficiencies especially delay and jitter and also other uncertainties and disturbances that can affect the performance of the system such as modeling errors, changing wind speed and inaccurate road grade data.

Another concern of this PhD research project is ecological driving. Pollution and the increasing cost of fossil fuels have encouraged researchers to investigate different possible ways to reduce vehicle emission and fuel cost. In the United States, traffic congestion wastes 3.1 billion gallons of fuel, 6.9 billion hours of extra time in traffic, and costed 160 billion dollars in 2014 [14]. Lowering trip energy cost can have a huge impact on the economy and general health of today's populated countries. The main consumer of energy in a vehicle

is the powertrain whose power demand is directly affected by speed trajectory. Looking ahead, speed adaptation and speed trajectory optimization by using information about the upcoming driving condition can significantly reduce energy consumption and traffic congestion. This improvement will also result in less fuel cost and tailpipe emissions. Using data from the oncoming traffic could decrease the time wasted in traffic which also results in less fuel consumption and is beneficial to the passengers. Here, we will review the current literature on ecological cruise controlling in the connected vehicle environment.

We will start by reviewing current researches on robust Eco-ACC for car-following and Eco-CACC for platooning and then CACC for heterogeneous platoons. Then, we will investigate a preproduction evaluation test for developing automotive control systems. Finally, we will conclude the necessary research for achieving a reliable system of connected vehicles based on the mentioned studies.

2.2 Adaptive cruise control

Advances in embedded digital computing and communication networks, along with the development of lower cost onboard vehicle sensors, have enabled the development of automated driving systems. While fully autonomous vehicles that can operate freely on roads are still very far from reality, semi-autonomous driving systems are around right now. These systems are mostly driver assistance that, instead of replacing drivers, help them to enhance their driving performance. The purpose of these assistance systems, which are called ADAS, is to sense the vehicle's environment and assist the driver according to the changes in the environment [7]. ADAS can reduce the effect of error in human judgment in emergency situations to improve driving safety and also improve driving performance by making optimal decisions to enhance the autonomous interaction with the environment. A major topic of interest among various types of ADAS is Adaptive Cruise Control (ACC), which has been the subject of many studies in this field. ACC is an advanced version of cruise control systems that can automatically decrease speed when approaching a preceding vehicle and increase it when the preceding vehicle starts to accelerate without any additional input from the driver. ACC uses a low-cost onboard sensor to measure the inter-vehicular distance and velocity compared to the preceding vehicle and then generates the appropriate control inputs to control the vehicle. This functionality can increase traffic flow and reduce accidents while improving comfort for the driver by providing a semi-autonomous driving experience [15].

The most common type of ACC system uses linear controllers to maintain a safe inter-vehicular distance. In [16] the authors developed a single-lane ACC with intelligent ramp metering to increase the capacity of highways, by enabling vehicles to move in short inter-vehicular distances. Their design was able to keep a one-second time headway under different traffic conditions. In [17], the authors created a more complicated ACC system that could adapt itself to different driving conditions. In their design, an algorithm automatically detects traffic situation based on local information and changes the parameters of the ACC system with respect to each traffic situation. In [18], a PID-based ACC was developed with the aim of performing the same as a human driver. Many other authors used linear methods in their design [19, 20]. These methods could provide a satisfactory tracking performance with low complexity. Also, PID controllers offer tuning parameters that can be easily adjusted to match different situations and systems. However, these methods only consider the current state of the system and cannot take into account the available information about the future driving condition. Another problem is that these methods are not able to handle constraints. Therefore, the safety constraints cannot be considered explicitly in control design, which may make them undesirable for practical use.

To improve the benefits of ACC systems, it is possible to consider fuel efficiency in its design. Considering the future prediction of traffic motion and environment of the vehicle could be very useful in this direction [21, 22]. This information can help the vehicle control system to make optimal decisions to prevent unnecessary accelerations and decelerations, which in turn will result in enhanced energy consumption and comfort for the passengers. Utilizing ACC to improve the fuel efficiency of vehicles has been widely investigated in previous research. In [23], Model Predictive Control (MPC) was incorporated in real-time to find the optimum speed trajectory, which then was fed to the cruise control system of a heavy diesel truck. Their results showed that incorporating look-ahead cruise control can significantly improve fuel economy, especially on hilly roads with available elevation profile. Reference [24] proposed a multi-objective predictive controller, which could take into account both speed tracking and fuel consumption at the same time. In [25], the authors developed an Ecological ACC (Eco-ACC) system based on MPC. They assumed a stationary condition (zero acceleration) on surrounding vehicles in order to perform their finite horizon optimization. Their result showed that utilizing future prediction of the preceding vehicle's trajectory yields a better fuel economy. MPC is a powerful method that can take into account future prediction and therefore has been used by many studies in this regard. In [21], a smooth acceleration degradation in the prediction horizon was employed to predict the preceding vehicle's trajectory and also presented a jamming wave prediction to prevent jamming waves while maintaining a safe inter-vehicular distance. The

authors used MPC to address this problem and showed that their method improves traffic flow and driver comfort as well as fuel efficiency. In [11],[22] a higher energy efficiency has been achieved via exploiting nonlinear MPC in designing Eco-ACC for PHEV. Both studies used a communication to the V2V and V2I communications to improve their prediction of the preceding vehicle.

Many of the existing studies in this area consider radar measurements to be reliable with no imperfection or uncertainty. This assumption may not be correct in real practice, for example, the performance of the radar and Lidar are highly dependent on weather conditions and it has been shown that fog intensity, rain, snowfall and snow storm can significantly decrease their accuracy [26, 27]. These sensors can also have different levels of accuracy depending on the number of objects in range and the type of background [28]. A reliable ACC must be robust to data uncertainty and modeling error to be able to keep its stability and performance. In [29], the authors considered wind and road disturbances and designed a model-based controller based on the H_2/H_∞ control method. Their simulations showed improvements in tracking performance. In [30], the authors assumed a linear model for their vehicle with disturbances on the states to develop a robust ACC system. They achieved a better tracking performance at the cost of higher computational effort. They used the min-max robust MPC method in their design, which has a high computational burden and therefore is not appropriate for real-time applications. Tube-based MPC (T-MPC) is another version of robust MPC that works based on a tube resulted from bounded uncertainties in the system [31,32]. T-MPC keeps the nominal system inside a tighter region to ensure boundedness of the real states inside the defined constraints. Since the required tube can be calculated offline, the computational demand of T-MPC is not much higher than regular MPC, which makes it appropriate for real-time applications. In [33] and [34], the authors used T-MPC to design a semi-autonomous ground vehicle. They considered the uncertainties and nonlinearities as an additive disturbance and then calculated a tube for the disturbed states. Their MPC used the resulted tube to gain robustness against system uncertainties. In [35], the authors developed a robust ACC controller using linear T-MPC. Their simulations on a high-fidelity vehicle model showed that this method can ensure robustness against delayed data, uncertainty and modeling errors in a car-following scenario. The T-MPC method in [35] uses a linear model which means the nonlinear part of the system must be translated as a disturbance. Therefore, this methods will result in a large disturbance tube that can increase the conservativeness of the robust controller.

Robust control methods can guarantee safety and stability, however, they are usually conservative and can deteriorate the performance of the controlled system. Therefore, many researchers prefer adaptive control approaches that estimate changes in parameters and

respectively adapt to them to maintain performance and stability of the system. Moreover, due to changing conditions and aging of the car, the actual parameters of a car might change and therefore an online optimization algorithm with a fixed model might not be able to find the actual optimal control decisions. To consider this matter, in [36], the authors designed a hierarchical cruise control for connected vehicles and used a gradient-based parameter estimation to estimate changes in vehicle parameters. They fed the estimated parameters to a low-level sliding-mode controller to regulate axle torque so that the desired states are followed. In [37], the authors used a recursive least square parameter estimator and adaptive nonlinear MPC to design a cruise controller with fuel optimization. They used parameter estimation to improve the control-oriented model of their MPC and, by performing vehicle experiments, showed that their method can achieve a 2.4% improvement in fuel economy compared to a production cruise controller. Similar adaptive control approaches can be found in [38],[39],[40],[41]. Although these methods can capture the changes in the model and act according to them, in the event of a sudden change in parameters or wrong estimations, they may lose performance and stability [42]. Especially, for close car-following, the controller must be able to guarantee safety of the system while improving the control performance. Therefore, a method is needed that can adapt to changes while being robust to uncertainties, disturbances and model errors. To combine robustness and model adaptation, in [43] and [44] the authors used a type of adaptive MPC that they called learning-based model predictive control. In their method, a linear controller generates optimal inputs based on a learned linear model and a separate model checks if the constraints will be satisfied. Nonlinear learning based MPC was used in [45] and [46] for path tracking control of a mobile robot in the outdoor and off-road environment. They used a simple known model and a Gaussian process disturbance model that can be learned based on trial experience. Their experimental results on different robot platforms show that their controller is able to reduce path tracking error by learning and improving the disturbance model through experience.

In this PhD research project, an adaptive tube-based nonlinear MPC (AT-NMPC) controller is presented that can improve the performance of Eco-ACC by adapting to the changes in the system while maintaining robustness against uncertainties, disturbances, and modeling errors. This method decouples performance from robustness and therefore is able to maintain stability and safety while adapting to changes in the system and the environment. First, the nonlinear T-MPC method is used to design a robust controller that can handle uncertainties. These uncertainties include the uncertainty in the estimation of the drag coefficient, uncertainty in the estimation of gravitational forces, due to uncertainty in road's grade estimation, uncertainty in the preceding vehicle's acceleration and also delay

in the data gathered from the onboard vehicle radar. The designed controller optimizes the vehicle's motion in finite horizon to improve the consumed energy cost of the vehicle while handling the defined constraints in the presence of uncertainty and disturbances. Then, to capture changes in the system and enhancing the control-oriented model, an online parameter estimation algorithm is used that estimates new parameter values based on minimizing the error between the estimated and actual output of the system. This way the online optimization will find the actual optimal point based on the adapted model while constraints are handled based on the original nominal model.

2.3 Cooperative adaptive cruise control

Vehicle platooning has attracted a vast attention from academia and industry in the recent years. Cooperative Adaptive Cruise Control (CACC) is the enabling technology for vehicle platooning. CACC extends the current version of ACC by utilizing wireless V2V and V2I communications. V2V communications provide more useful information about the preceding vehicle and also other vehicles in range, with lower delay compared to radars [47]. Using V2V communications and radar, CACC can enable a string of vehicles to safely, efficiently and cooperatively move in short, inter-vehicular distances to construct a platoon of vehicles.

As mentioned above, ACC can enable autonomous car-following, which is the ability to follow a preceding vehicle with no additional inputs from the driver. It has been shown that employing effective ACCs can significantly enhance safety [48], driver comfort [49], traffic flow [50], vehicle emissions performance [51] and fuel economy [52]. Improved fuel economy can be achieved by utilizing future predictions of the preceding vehicle's trajectory. NMPC is a convenient way to achieve this improvement in ACC [53]; however, when the number of vehicles increases to form a platoon, it might not be as useful as in regular ACC systems.

Platooning is a step forward from ACC that provides close car-following for a string of vehicles. Its vast benefits have encouraged many researchers to further investigate platooning of connected vehicles. It has been shown that platooning improves traffic flow and road throughput [47] and enhances fuel economy by reducing aerodynamic drag [54]. ACC can enable two vehicles to closely follow each other, but if the number of vehicles increases to make a platoon of vehicles, instability may occur. This instability occurs due to the existing delay in the radar's performance that could cause instability in upstream vehicles. Therefore, in addition to a radar, V2V communications are also necessary to

enable platooning. Otherwise, an ACC based platoon will require a bigger inter-vehicular distance to become stable, which undermines the whole benefits of platooning.

One of the main objectives of platooning is achieving string stability. String stability is a term defined for interconnected systems that, besides their own internal states, have interconnection states. It is a term used in many applications, such as economy [55], irrigation systems [56] and supply chain management [57]. From 1974 onwards, researchers have found string stability to be one of the major issues in vehicle platooning [58]. Swaroop and Hedrick [58] defined string stability as a uniform boundedness of states of all the systems. A more practical definition for our application is: an attenuation of disturbance signals along the vehicles' string. Different mathematical descriptions can be used to define a platoon's string stability. For example, [59] used the H_∞ norm and [58] used the L_∞ norm to define string stability and stated that exponentially stable connected systems are string stable if they have a weak coupling, meaning that a signal's energy will decrease as it propagates through the string. Other types of string stability definitions also exist in the research literature; for example, [60] used L_2 norm and [61] defined L_p norm for the definition of string stability. String stability can be defined compared to the immediate preceding vehicle or only the lead vehicle. String stability compared to leader requires a direct communication to the leader, which will put a maximum length for the platoon because of the limited range of V2V communications. String stability compared to the immediate proceeding vehicle only requires a communication to that preceding car, which improves the scalability of the CACC design. However, this type of string stability is more conservative and puts a tighter constraint on the dynamic of connected systems [62].

String stability is an important factor in platooning, but it is certainly not enough for ensuring the platoon's safety and performance. A platooning controller must be able to maintain a maximum and minimum inter-vehicular distance and absolute velocity while considering actuation limits and comfort constraints. Other than string stability, for a platoon, the safety and performance requirements must be satisfied within the allowable route's limits and available actuation efforts. Otherwise, the safety and performance of the platoon cannot be guaranteed. To find necessary safety constraints, Kianfar et al. [63] used reachability analysis to come up with a maximal asymptotic safe set for a vehicle in a platoon, defined as a set of states that a given controller is guaranteed to control the desired speed and distance while fulfilling the defined constraints. They applied a PD controller to their platoon and assumed the preceding vehicle's acceleration as a disturbance affecting the system and used backward reachable sets to come up with the safe set in the case of preceding vehicle's emergency braking. Alam et al. [64] used a similar approach but defined a pursuer and evader game. They came up with a safe set for platooning without

considering a specific controller. In their controller synthesis, they considered the effect of delay and showed that a greater minimum inter-vehicular distance is required for bigger communication delays.

The main issue in platoon constraint handling is that it has a time-domain definition while the string stability is defined in frequency-domain. MPC is a powerful controller that is able to handle constraints and its formulation is appropriate for online applications. However, since it is a time-domain approach, it cannot enforce string stability in decentralized platoon controllers. In [60], to combine constraint handling and string stability, the authors designed a string stable linear platooning controller and optimized the weightings of their MPC objective function to get the same output as the linear controller. The result is a controller that is string stable, since it gives the same output as the linear string stable controller and, only when it is necessary, changes the control action to maintain constraints. Their experiment on a three-vehicle platoon showed that MPC comes into work when a constraint is violated and moves the state inside the defined constraint by harsh braking. In [62] and [65], enforcing string stability was studied by translating it into an inequality constraint. This method requires all vehicles to broadcast future trajectories to followers, which may be impractical.

One major contribution of this PhD research project is the design of a platooning controller that can simultaneously achieve both string stability and constraint handling. First, we will design a linear controller to guarantee the string stability of the system. This controller can be designed based on any definition of the string stability. However, here we will use the induced L_2 norm criterion to define string stability. To enforce the defined constraint on the string stable platoon, we propose the design of a predictive RG. An RG is a nonlinear controller that sits behind a controlled stable system and keeps the system inside the defined output constraints by modifying the original reference [66]. It separates the design of the main controller from constraint handling and therefore it is very useful in alleviating potential concerns from computational time, robustness, stability and tuning complexity [67]. We will design a reference management controller, which is a type of RG, to enforce the platooning constraints on the previously designed string stable platoon. Unlike classic RG, this method can work with alternating references and also void admissible sets and therefore it is suitable for platooning control design.

Ecological aspects of platooning have been discussed before in many previous publications in the literature. short inter-vehicular distance in a platoon can decrease the drag force on the follower vehicle, which in turn will result in lower energy consumption for the whole platoon [68, 69]. However, reduced drag requires an extremely short inter-vehicular

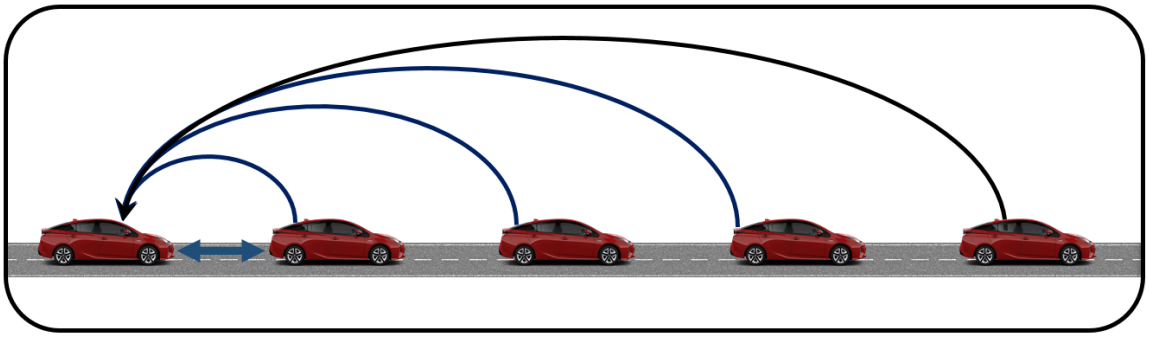
distance, which may not be possible due to safety requirements. For this reason, energy economy due to drag reduction is usually considered only in the case of heavy-duty vehicles platooning that have much higher drag resistance compared to passenger cars [54]. Aside from the reduced drag, a proper CACC can reduce the energy cost by minimizing unnecessary accelerations and decelerations. In [70], the authors developed a two-layered control architecture to improve the safety and fuel efficiency of their platoon. They used dynamic programming in the high-level to find an optimal speed trajectory based on the road topology and MPC in the low-level for real-time control of the vehicle. Their results showed increased fuel efficiency of up to 12 %. A platoon's fuel economy in urban roads was studied in [71] using MPC and in [72] using fast MPC. Their controller received traffic light scheduling through V2I communications and generated a target velocity to minimize the idling time behind the traffic light. Their results showed significant improvements in the fuel consumption and reduced idling times. In these investigations, a decentralized MPC offered a convenient way to develop environmentally-friendly platoons by considering an ecological control for each vehicle individually, using available data from the connected vehicles' environment. The problem is that these controllers cannot guarantee the string stability of the platoon due to their time-domain nature. Therefore, the real-world implementation of these methods will be in question. To have a practical platooning system, improved energy economy must be considered using an approach that is able to enhance energy consumption while maintaining string stability and passenger's safety.

In this PhD research, predictive platooning controllers are proposed to achieve Eco-CACC for a string of vehicles. We develop a predictive RG that enforces the constraints of the string stable platoon. Also, a predictive controller based on NMPC is presented to control the platoon's leader to improve the energy economy of the whole platoon. The proposed controllers are fine-tuned for a Toyota Prius PHEV, which is our baseline vehicle and will be evaluated using an Autonomie-based, high-fidelity model of this vehicle. Although the control evaluations in this research are performed on the Toyota Prius PHEV, the proposed approach is equally valid for most other passenger and commercial vehicles. One main contribution of this PhD research is the design of a platooning controller that can simultaneously achieve string stability in the frequency-domain and constraint handling in the time-domain. The RG approach is proposed to separate platooning time-domain requirements from frequency-domain requirements.

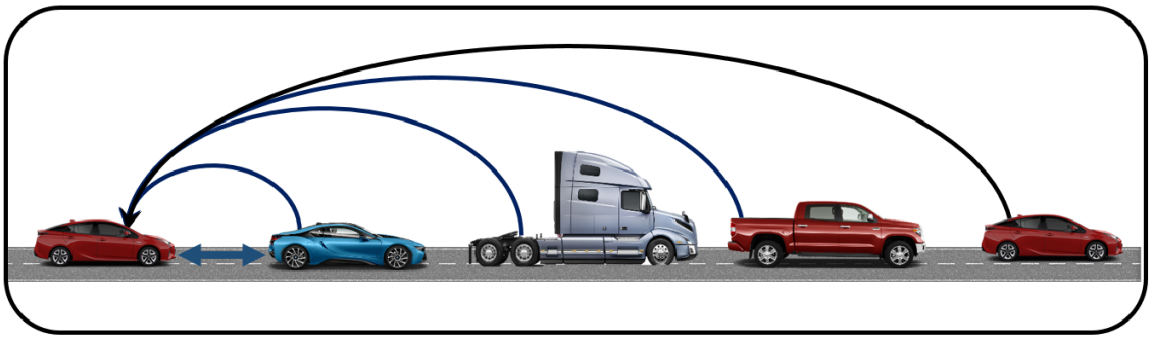
2.4 Heterogeneous platooning

As mentioned above, a platoon consists of a number of vehicles that are connected through a wireless network and follow each other in short inter-vehicular distances. Depending on the number of vehicles in the platoon, and the shape of the connection network between them, a platoon control problem can become extremely complicated. Therefore, to be able to solve such a complicated problem, many simplifying assumptions are usually required. For example, many studies assume the agents (vehicles) as point masses to simplify the dynamic of the whole system [73, 74] or the force drag reduction caused by short inter-vehicular distances is usually ignored due to its complicated uncertain behavior [75]. Another very common simplifying assumption in most studies is homogeneity [36, 54, 60, 72]. A homogeneous platoon consists of identical vehicles with the same dynamic models. This assumption is useful in the study of platooning for a special kind of car, however, in practice, a vehicle might have to join a platoon with non-identical vehicles. Figure 2.2 illustrates a homogeneous and heterogeneous platoon. In both cases, a follower vehicle receives information about all preceding vehicles through wireless communications. Based on this figure, the leader's motion affects a follower in two ways: firstly by the signal that it sends to the follower through V2V communications and secondly by affecting the motion of other followers and therefore affecting the inter-vehicular distances of all follower vehicles. To be able to use the received signals through V2V communications, platooning controller must be able to match it to the motion of its preceding vehicle. Otherwise, string instability can occur due to improper use of V2V communications signals. This matter will be discussed further in chapter 5 with mathematical explanations. In chapter 4, platooning of PHEVs is considered. However, in reality, a PHEV might have to join a platoon constructed by different types of vehicles. Therefore, to develop more practical platooning controllers for PHEVs, we need to also consider the case of heterogeneous platoons.

There can be different reasons for causing a heterogeneity in a platoon. Several types of network topologies exist in the literature that each can be used to form a platoon. For instance, some studies consider communication only between a vehicle and its immediate predecessor [75, 76], many studies consider communications to the immediate preceding vehicle and leader [77], some studies consider communications to a predecessor and a follower which forms the bidirectional platoon by using a forward and a backward radar [38, 78, 79] and also many other forms of network topologies. These various topologies result in a heterogeneity of the vehicle platoons. Another source of heterogeneity is different spacing policies that each vehicle might be using to perform car-following. There are several types of spacing policies such as constant spacing, constant time headway, and nonlinear



(a)



(b)

Figure 2.2: (a) a homogeneous vehicle platoon, and (b) a heterogeneous vehicle platoon with all preceding vehicles communications

spacing policies. An important cause of heterogeneity is the different types of vehicles in a platoon with different dynamics. Each vehicle has different actuation delay, vehicle lag, maximum and minimum acceleration and brake forces and also different low-level controllers. Therefore, a practical platoon controller must be able to take these difference into account.

In [80], the authors reviewed the causes and effects of heterogeneity in string stable platooning. They suggested a platooning control architecture by feed-forwarding the acceleration set-point from V2V communications and using a linear controller to adjust the inter-vehicular distance. Then they studied the effect of delay and different constant time headway values on string stability. To improve the string stability of their heterogeneous platoon, they assumed a known dynamic model for the preceding vehicle and adjusted their control signals based on that. In [81] the authors developed a distributed adaptive sliding mode controller for a heterogeneous platoon with constant spacing policy. They proved the string stability of their controller by using an explicitly constructed Lyapunov function. Moreover, they used an adaptive algorithm to reduce the effect of external acceleration disturbances. Robust control of heterogeneous highway platoons was considered in [82] based on the H-infinity control method. They considered parametric uncertainties in the model of their platoon and based on that analyzed the robustness. Their simulations showed that their platoon is able to achieve string stability for fixed but different vehicle models. A platoon with heterogeneous communication time-varying delays was considered in [83]. They used a distributed consensus strategy and derived a controller composed of two terms: a local action dependent on the vehicle states and another term dependent on the network signals from neighboring vehicles. They proved stability of their system by using Lytapunox-Razumikhin theorem. A multi-layer consensus seeking approach was proposed in [84] for heterogeneous platooning. They designed a two-layered framework to separate trajectory planning from controlling toward those trajectories and showed that their method could handle a platoon of vehicles with heterogeneous linear dynamic models. An ACC based heterogeneous platoon was considered in [85, 86] with no wireless communication. They proposed a decentralized controller and proved string stability based on their heterogeneous platoon model. In [87], authors used an adaptive control approach to heterogeneous platooning and came up with an adaptive law to achieve a string stable platoon with switching to ACC in the time of connection loss.

To address the heterogeneity problem, this research proposes a direct model reference adaptive control (MRAC) approach heterogeneous platoon. Direct MRAC uses a reference model and estimates the controller coefficients such that the controlled system would behave as the reference system [88]. This way a string stable behavior can be enforced onto

the system despite the dynamical differences and also alternating external disturbances. Therefore, the same controller can be used on different vehicles with separate dynamical models.

2.5 Hardware-in-the-loop experiment

As mentioned before, a major concern with automotive controllers is their computational burden. The developed controllers must be executed in real-time on a vehicle ECU and work with other subsystems to be able to be used in practice. Therefore, a crucial task in controller development is testing its performance on a real ECU. An effective way to do this task is to connect the developed ECU to a real plant and examine its performance in different situations. However, this method could be extremely time-consuming, costly, unsafe, inefficient and may also be too difficult to do on a real car. HIL experiments, however, are efficient tools for control development that can be used ahead of vehicle production [89]. In HIL experiment, instead of the real vehicle, a virtual model is used and the controller can be tested by connecting the developed ECU to the virtual model. HIL tests provide a more efficient way for early software verification and improving software quality by performing various kind of experiments in different virtual scenarios. These test can significantly reduce the vehicle development process time and cost by replacing expensive field tests by laboratory experiments with less hardware compared to physical prototyping. Moreover, HIL tests are safer than field experiments especially for tests in extreme conditions such as winter driving tests, collision avoidance tests, cold-start tests and also close car-following experiments [90].

Reference [91] used HIL experiments to evaluate a real-time explicit MPC design for energy management control of the powertrain of a PHEV. Their results showed an improved energy consumption as well as a real-time implementation performance. Reference [90] developed a power management strategy and an ACC system for PHEVs and used HIL tests to demonstrate the real-time implementation capability of these controllers. In [92], a dSPACE HIL experiment setup with MATLAB/Simulink was used to perform HIL tests for an electric vehicle powertrain which included vehicle energy management system, battery management system and motor control unit. Reference [93] developed a model-free optimal controller for ACC systems using reinforcement learning. They validated their controller through HIL tests using a dSPACE HIL experiment setup and by comparing to PID and LQR controllers. Several other research works have also used HIL experiments to validate their design for real-time implementations [94–96].

Model-in-the-loop (MIL) simulations are a convenient way to do these tests as well. In these simulations, ECU and its communications are also part of the virtual model and the combined model is executed on a desktop computer. However, such off-line simulations cannot validate the real-time performance of the controllers on an embedded system with a constrained sample time and with real-world I/O interface. Therefore, HIL experiments must be performed to measure the turnaround time, which is the time that the controller requires to be executed in each sampling time step, of the controller on a vehicle ECU to make sure that it can be executed in real-time. HIL tests take into account the computational limits and communication issues and therefore their results are considered more practical than MIL simulations. Because physical prototyping in the early stages of development of vehicle control systems will be very expensive, HIL experiments, which are less expensive and also faster and safer, are usually carried out before manufacturing the prototype vehicle [97].

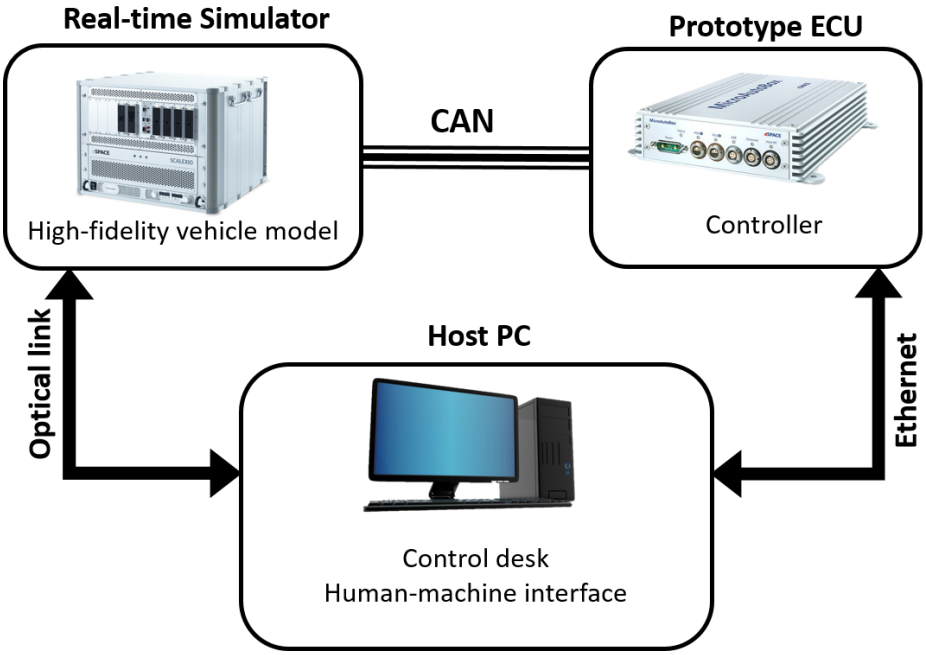


Figure 2.3: Schematic of an HIL experiment setup

In this research, a dSPACE Micro-Autobox II control prototyping hardware is used for HIL tests. dSPACE GmbH (Digital Signal Processing and Control Engineering) is a well-known provider of these tools especially for automotive applications and many vehicle manufactures, such as Toyota, General Motors, Honda, Ford, BMW, and Nissan are cur-

Table 2.1: Specification of the dSPACE HIL experiment platform

Component	Parts	Specifications
Real-time Simulator	Hardware Processor Memory I/O	DS-1006 Processor board DS1006 Quad-core AMD, 2.8 GHz 1GB local, 4x128 MB global DS-2202
Prototype ECU	Hardware Processor Memory I/O	MicroAutoBox II DS-1401 PowerPC 750GL 900 MHz 16 MB main, 16 MB non-volatile DS-1511
Host PC (Interface)	Hardware Processor Memory	Dell Intel Core i7, 3.4 GHz 16 GB

rently using their instruments. This specific setup is one of the widely used instruments for calibration and testing of ECUs especially for automotive applications. As shown in Figure 2.3, the HIL experimental setup has three main components: a prototype ECU (MicroAutoBox II), which is an independent processing module that runs the uploaded control algorithm; a real-time simulator (DS1006 processor board) which is responsible for running the complex high-fidelity model of the vehicle in real-time fashion; and a personal computer (PC) that serves as the human-machine interface and it is used for programming the real-time machine and prototype ECU, as well as for recording the desired test signals. All communications between the prototype ECU and the real-time simulator is performed through a Controller Area Network (CAN) bus.

Table 2.1 shows the specifications of the HIL setup. To test a developed controller, an optimized C code must be generated for the targeted ECU platform, which can make ECU development a challenging task. dSPACE provides libraries in MATLAB and Simulink to automatically generate C-codes for the both MicroAutoBox II and DS1006 processor board. These libraries support all features of these devices with blocks for communications through CAN bus. The generated C code from the developed controller must be uploaded to the prototype ECU and the high-fidelity plant model gets uploaded to the real-time simulator using the interface computer.

2.6 Summary

In this chapter, we reviewed the existing literature on robust Eco-ACC, Eco-CACC, heterogeneous platooning and HIL experiments. Although there are numerous studies investigating the design and evaluation of the mentioned controller, there remain knowledge gaps that need to be addressed in order to achieve reliable high-performance connected vehicles control systems.

Most of the existing works in this area have ignored the effect of network imperfections, data uncertainty, and modeling errors on their system. To guarantee the stability and improved performance of the designed controllers, considering robust controller design methods is crucial. In the proposed research, we will develop robust controllers to guarantee the stability and performance of our controllers in the presence of uncertainties, external disturbances, modeling errors and data imperfections.

String stability is the most important factor in platooning and achieving it has been the focus of many studies in the literature. However, simultaneous satisfaction of string stability and constraint handling is an important issue that is missing from the literature. The frequency-domain definition of string stability and time-domain definition of constraint handling make combining them in a single controller very complicated. In this PhD research, we will develop CACC controllers for constraint handling of a string stable platoon.

Although, this research is mainly concerned with developing ACC and CACC for PHEVs, in practice, a PHEV might also have to join a platoon of non-identical vehicles. Therefore, to have more practical PHEV platooning controllers, it is also needed to study PHEV platooning controllers in interaction with heterogeneous platoons. Here, we will develop adaptive platooning controllers for heterogeneous vehicle platoons based on the model reference adaptive control method.

MPC has been very popular in the design of ACC and CACC systems for connected vehicles since it is able to incorporate future trip data in the control procedure and can handle multi-objective optimization problems with constraints. T-MPC technique is a robust version of MPC that has lower computational demand than its other competitors. In this research, we will take advantage of T-MPC method to design robust Eco-ACC systems. Moreover, we will extend the T-MPC to nonlinear adaptive T-MPC (AT-NMPC) to be able to use nonlinear vehicle models in the design of our controller and also maintain high performance by online model adaptation.

Real-time implementation capability is an important factor for the practical application of automotive controllers. Proper and fast solvers will be used to solve the presented optimal control problems in real-time. Moreover, hardware-in-the-loop (HIL) experiments will be performed to evaluate the performance of the designed controllers on a test ECU.

Chapter 3

Robust Ecological Autonomous Cruise Control

This chapter is dedicated to the design of a robust and adaptive ACC for ecological improvement of the baseline PHEV. An adaptive tube-based nonlinear MPC (AT-NMPC) controller is presented that can improve the performance of Eco-ACC by adapting to changes in the system while maintaining robustness against uncertainties, disturbances and modeling errors. This method separates the robust satisfaction of constraints from performance and therefore is able to maintain stability and system's safety while simultaneously adapting to changes in the system and also the environment. To design this controller, first, the nonlinear T-MPC method is used to design a robust predictive controller that can handle the uncertainties in estimation of the resistance drag forces, gravitational forces due to uncertainty in road's grade estimation, uncertainty in the preceding vehicle's acceleration and also delay in the data gathered from the onboard vehicle radar. The designed controller optimizes the vehicle's motion in finite horizon to improve the consumed energy cost of the vehicle while handling the defined constraints in the presence of uncertainty and disturbances. Then, to capture changes in the system and enhance the control-oriented model, an online parameter estimation algorithm is used that estimates new parameter values based on minimizing the error between the estimated and actual outputs of the system. This way the online optimization will find the actual optimal point based on the adapted model while constraints are handled based on the original nominal model.

The main contribution of this chapter is in combining robustness against uncertainties with parameter adaptation in the design of the Eco-ACC controller. To achieve this performance, the AT-NMPC approach is proposed that has a nonlinear formulation, which

means that it can include a more detailed model of the vehicle and also uses two separates models for robust constraint handling and cost function definition. Moreover, to be able to execute the designed optimal controller in real-time, a Newton/GMRES fast solver is adapted to solve the AT-NMPC optimal control problem.

The rest of this chapter is structured, as follows: Section 3.1 presents preliminary definitions. In Section 3.2, modeling procedure is explained and a control-oriented model for car-following is presented that can be used for the design of ACC controllers, also uncertainty bounds are defined based on the presented model as an additive disturbance. In Section 3.3, the controller design is explained by taking advantage of the AT-NMPC method to achieve robustness and high performance in ACC design. Section 3.4 is devoted to controller evaluations. In this section, the proposed method has been simulated on a high-fidelity vehicle model in a car-following scenario and in a simulation environment with injected uncertainties. Moreover, HIL experiments are presented that show the real-time implementation capability of the devised controller. Finally, some concluding remarks are presented in Section 3.5.

3.1 Preliminaries

It is required later in this chapter to perform some set calculations for the design of the AT-NMPC controller, in order to find the necessary control sets. Therefore, the following definitions are useful throughout this chapter.

The sign $\oplus$ indicates the Minkowski's sum, which is the sum of two sets and results in a set calculated by adding each point of each set to all points of the other set. Therefore, if X and Y are sets, then:

$$Y \oplus X = \{x + y : x \in X, y \in Y\}.$$

Erosion of a set Y with respect to X is shown by $\ominus$, which is usually referred to as the Pontryagin's set difference. This set difference can be written as:

$$Y \ominus X = \{v \in R^n : v \oplus X \subseteq Y\}.$$

In the same way, Minkowksi's sum of three sets can be defined as:

$$X \oplus (Y \oplus V) = \{x + y + v : x \in X, y \in Y, v \in V\}.$$

The scaling of a set can be defined as:

$$AX = \{Ax : x \in X\},$$

and therefore based on the previous definitions, we can conclude that:

$$AX \oplus BY = \{Ax + By : x \in X, y \in Y\}.$$

Also, the summation of multiple sets is defined as:

$$\oplus_{i=n}^{m} Y_i = Y_n \oplus Y_{n+1} + \cdots \oplus Y_m.$$

3.2 Modeling

This section explains the models that have been used for the design and evaluation of the proposed controller. Control evaluations have been done with a high-fidelity model of the baseline vehicle, which is the Toyota Prius PHEV. It consists of detailed high-fidelity models and mappings of all the components in the vehicle that can affect longitudinal motion and energy consumption. The high accuracy of this model makes it a reliable tool for the evaluation of the designed controllers. For the control design, however, a simple model is needed that has low computational demand but is descriptive enough to capture the general behavior of the system. Here, different control-oriented models are presented that represent longitudinal motion and energy consumption of the vehicle.

3.2.1 High-fidelity powertrain model

Figure 3.1 shows the schematic of a power-split PHEV. This system divides the power of the engine along to different paths. One path is to the generator to produce electricity and the other goes through a mechanical gear system to power the wheels. The main components of this system are the engine, power split gearbox, electric motor/generators, transmission shafts, final drive, and battery. Combining these parts will make a HEV. The high-fidelity model, we are using the Toyota Prius PHEV model developed in Autonomie, which is a famous commercial MATLAB-based software. This model has been previously developed and validated in our research group [9, 90, 98].

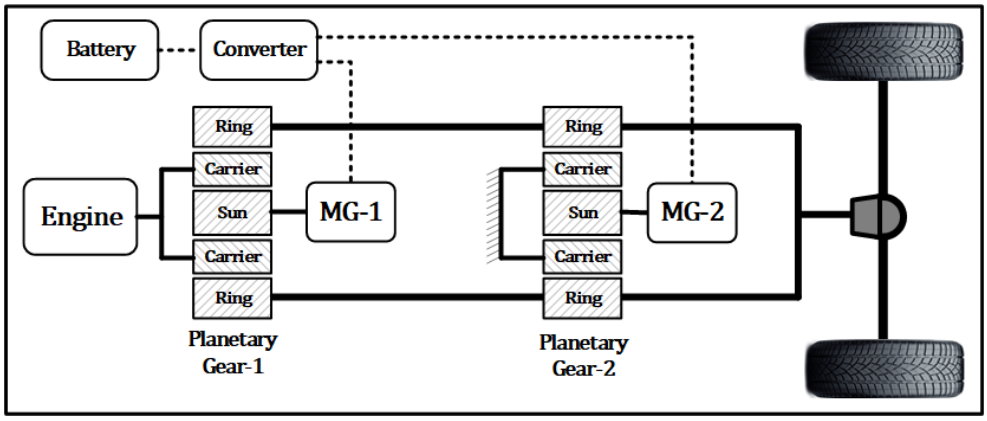


Figure 3.1: Schematic of Toyota power-split powertrain

The engine is responsible for producing torque from the combustion of gasoline. An engine controller gets the input from the driver and, based on the current speed of the engine, changes the fuel rate to get the desired torque. The output torque of the engine is dependent on many factors like fuel and air ratio, engine speed, temperature, and internal friction from the chamber walls, etc. To model these relations, Autonomie uses mappings that were developed by performing tests on the real engine. Also, the engine model produces fuel consumption and emissions based on the working trajectory of the engine. Plug-in Prius has a 1.8L internal combustion engine with maximum power of 73 kW and torque of 142 N.m.

The electric motors of the Prius powertrain can work as a generator and electric motor. In Figure 3.1, the electric motor/generators are shown as MG-1 and MG2. A high-fidelity model of these motors is provided by mappings between speed and motor commands to output torque. The power split drive decouples the engine from the wheel and, combined with electric motors, adjusts the engine's operation to get the highest fuel efficiency. The transmission system consists of two planetary gears. The engine is connected to the sun of the first planetary gear and MG-1 is connected to its carrier. MG-2 is connected to the sun of the second planetary gear and its carrier is connected to the chassis. The rings of both planetary gears are connected directly to the wheels.

The final drive is responsible for splitting the power between the left and right wheels. Since we are considering longitudinal dynamics and straight line maneuvers, the final drive model is a simple speed conversion with an efficiency and a constant inertia.

Drive shafts are responsible for connecting the final drive to the wheels. In reality,

these shafts have flexibility and their input angle is different from their output. Here, we do not consider this flexibility and our model considers a solid connection.

The **battery** is modeled as a simple RC circuit. The relation between battery power and electric power is as follows:

$$P_{bat} = \frac{P_m}{\eta_m} - P_g \eta_g, \tag{3.1}$$

where P_m and η_m are actuation power and efficiency of electric motors, P_{bat} is the battery power, and P_g and η_g are generator power and efficiency when the electric motors generate energy. The battery's consumed energy can be calculated as:

$$S\dot{O}C = \frac{I_{bat}}{Q_{max}} = \frac{-V_0 + \sqrt{V_0^2 - 4P_{bat}R_{bat}}}{2R_{bat}Q_{bat}} = -\eta_b \frac{P_{bat}}{E_{max}} \tag{3.2}$$

where SOC is the battery state of charge, I_{bat} is the battery current, Q_{max} is the maximum battery electric charge, V_0 is the open circuit voltage, R_{bat} is the internal resistance, E_{max} is the maximum battery energy, and Q_{bat} is the battery charge.

External Longitudinal Forces affect the longitudinal motion of the vehicle. The main external forces are drag force, gravitational force and rolling resistance force.

Drag Forces are produced because of the resistance of air against the motion of the vehicle. Aerodynamic forces have a significant impact on the vehicle, especially in higher speeds. The drag force can be estimated at:

$$F_{drag} = \frac{1}{2}\rho_a A_a C_d v^2, \tag{3.3}$$

where ρ_a is the air density, A_a is the maximum frontal area, C_d is the drag coefficient and v is the speed of the vehicle.

Gravitational Forces are acting on the system when the vehicle is moving on a slope. The gravitational force can have a significant impact on the vehicles fuel consumption. If the road's grade angle (θ_g) is available, it can be calculated by the following equation:

$$F_{gravity} = mg\, sin\theta_g. \tag{3.4}$$

Rolling Resistance is the resistance against the tires rotation on the road. It is produced because of the eccentric vertical force from the ground to the wheel. It is mostly

modeled as a friction force acting on the tire.

$$F_{rr} = mg(\mu_{r_0} + \mu_{r_v} v)\, cos\theta_g, \tag{3.5}$$

where μ_{r_0} and μ_{r_v} are rolling resistance coefficients and θ_g is the grade angle.

Combining the powertrain model and external forces acting on the system, we can come up with the longitudinal dynamics equations of the power-split PHEV [99]:

$$\begin{aligned}\left(\frac{I'_v(S+R)^2}{RI'_eK} + \frac{I'_vS^2}{RI'_gK} + R\right)\dot{\omega}_m = \left(\frac{(S+R)^2}{RI'_e} + \frac{S^2}{RI'_g}\right) T_m \\ + \frac{S+R}{I'_e}T_e + \frac{S}{I'_g}T_g - \left(\frac{(S+R)^2}{RI'_eK} + \frac{S^2}{RI'_gK}\right)T_d,\end{aligned} \tag{3.6}$$

where T_m, T_e, T_g are the torques of the MG-2, engine, and MG-1, respectively. Also:

$$\begin{aligned} I'_v &= \frac{mr_w^2}{K} + I_mK + I_rK, \\ I'_g &= I_g + I_s, \\ I'_e &= I_e + I_c, \\ T_d &= mgr_w\big((\mu_{r_0} + \mu_{r_v})cos\theta + sin\theta\big) + \frac{1}{2}\rho_a A_a C_d\,(\frac{\omega_m}{K})^2\, r_w^3, \\ \omega_m R + \omega_g S &= \omega_e(R+S), \end{aligned} \tag{3.7}$$

where ω_m, ω_e, ω_g angular velocity of the MG-2, engine, and MG-1, respectively, and I_m, I_r, I_g, I_s, I_e and I_c are the inertia for the motor, ring gear, generator, sun gear, and engine; the carrier K is the final drive ratio, and; R and S are the numbers of teeth on the ring and sun gears.

We will use the high-fidelity model for the controller evaluation purposes and for validating our control-oriented models. This model has been developed by other members in the author's research group who are continuing to improve its accuracy by performing identification tests on the Toyota Prius PHEV components [9,90,100]. Table 3.1 shows the parameter values of the high-fidelity model.

Table 3.1: The characteristics of Toyota Prius Plug-in Hybrid.

Parameters	Symbol	Unit	Value
Vehicle mass	m	kg	1703
Drag coefficient	C_d	$-$	0.25
Frontal area	A_a	m^2	2.19
Static rolling resistance	μ_{r_0}	$-$	0.012
Dynamic rolling resistance	μ_{r_v}	$\frac{s}{m}$	0.0001
Engine power	P_e	kW	73
Motor power	P_m	kW	50
Generator power	P_g	kW	30
Wheel radius	r_w	m	0.31
Number of battery cells	N_b	$-$	56
Battery cells nominal voltage	V	v	3.7
Battery nominal capacity	Q	Ah	21

3.2.2 Car-following control-oriented model

To develop a model for car-following problem shown in Figure 3.2, it is necessary to define a safe car-following rule. Among different spacing policies in literature, a constant time headway rule was chosen as:

$$d = d_0 + hv_h,$$

where d is the desired distance, d_0 is the minimum distance at standstill, v_h is the host vehicle's velocity and h is the constant headway time [15]. This spacing policy requires increasing distance with respect to velocity so it takes a specific constant amount of time for the host vehicle to reach to its preceding. This behavior is similar to a human driver and it is a natural assumption for increasing safety. Based on the chosen gap policy, the

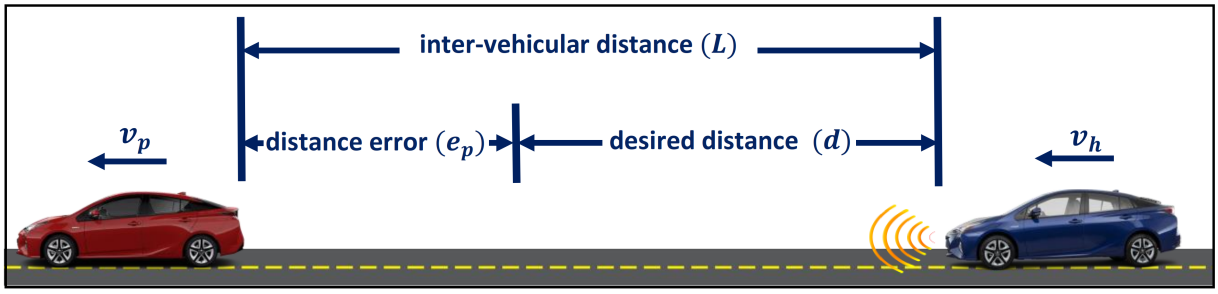


Figure 3.2: Car-following with autonomous cruise control

state equations of the system can be written as follows:

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu(t-\tau_a) + B_p a_p + B_g F_r$$

$$\mathbf{x} = \begin{bmatrix} e_p \\ e_v \\ v_h \\ T_w \end{bmatrix}, \; A = \begin{bmatrix} 0 & 0 & 1 & -\frac{h}{mr_w} \\ 0 & 0 & 0 & -\frac{1}{mr_w} \\ 0 & 0 & 0 & \frac{1}{mr_w} \\ 0 & 0 & 0 & -\frac{1}{\eta} \end{bmatrix}, \; B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{K_a}{\eta} \end{bmatrix}, \; B_p = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \; B_g = \begin{bmatrix} -h \\ -1 \\ 1 \\ 0 \end{bmatrix}, \tag{3.8}$$

$$F_r = -\frac{1}{2}\rho_a A_c C_d (v_h + v_w)^2 - (\mu_{r_0} + \mu_{r_v} v_h) mg\cos(\phi_r) - mg\sin(\phi_r),$$

where $\mathbf{x}$ shows the state of the system in continuous time domain, $e_p = p_p - p_h - hv_h$ and $e_v = v_p - v_h$ are the position and velocity errors, a_h, v_h, and p_h are the acceleration, velocity and position of the host vehicle, p_p, v_p and a_p are the velocity, position and acceleration of the preceding vehicle, τ_a is the actuation delay, F_r represents the sum of all the resistance forces, ρ_a is the air density, A_c is the frontal area of the car, C_d is the drag coefficient, μ_{r_0} and μ_{r_v} are the rolling resistance coefficient, m is the vehicle's mass, v_w is the headwind speed and ϕ_r is the road grade. T_w is the wheel torque and r_w is the wheel radius. Equation (3.8) represents a nonlinear control-oriented model that is used in the design of the controller and calculating the disturbance sets. This model includes drag forces by vehicle motion and wind, rolling resistance force and gravity force due to grade changes. To define a cost function based on energy economy improvement, a control-oriented model for the consumed energy is needed. Because our baseline vehicle is a PHEV, we have to consider both fuel and electricity costs. Therefore, instead of fuel rate and electrical current, we define the cost function based on the combined energy cost of the two sources.

$$E_{cost} = -C_f \frac{\dot{m}_f}{v_h} - C_e \frac{\dot{SOC}}{v_h}, \tag{3.9}$$

where E_{cost} is the costs of energy, C_f and C_e are the cost of gasoline and electricity, respectively, and SOC is the state of charge of the battery. The energy cost has been divided by the host vehicle's velocity to eliminate the effect of traveled distance. This way, the consumed energy per distance unit is considered in the cost function. At lower velocities, the energy cost will be assumed constant to avoid singularities. An energy management algorithm decides the distribution of energy between the energy sources while the vehicle is running to keep the powertrain near its optimal working point. Therefore, it can be assumed that the engine is always working in its optimum working point and approximate the fuel consumption with the following equation [75]:

$$\dot{m}_{f_i} = \alpha_1 + \alpha_2 P_e + \alpha_3 P_e^2 + \alpha_4 v_h, \tag{3.10}$$

where $\dot{m}_{f_i}$ is the fuel rate, P_e is the engine power and α_1, α_2, α_3 and α_4 are constant coefficients. To estimate the electricity rate, we used the following equation:

$$\dot{SOC} = \gamma_1 + \gamma_2 P_m + \gamma_3 P_m^2, \tag{3.11}$$

where P_m is the motor's or generator's power and γ_1, γ_2, and γ_3 are constant coefficients. The squared electric power has been included in the model to represent ohmic losses. As mentioned previously, the energy management controller decides the power ratio between electricity and gasoline. Therefore, based on power ratio, the energy cost can alternate for different total power demands. Based on power ratio, the power demand from each source can be calculated:

$$\begin{aligned} PR &= \frac{P_e}{P_{total}}, \\ P_e &= PR \cdot v_h \cdot u \cdot m, \\ P_m &= (1 - PR) \cdot v_h \cdot u \cdot m, \end{aligned} \tag{3.12}$$

where PR represents the power ratio and P_{total} is the total power demand. Using this equation, the energy cost of the trip can be calculated from the total power demand and power ratio of the energy management controller.

3.2.3 Reduced model

The control-oriented model needs to be updated based on online measurements in the system. Here, adaptation has been done based on a recursive least-square method, which requires a parametric model. The following formulation has been used for the longitudinal dynamic's parametric estimation model:

$$a_h = \frac{R_g \eta_p}{r_w m} T_{com} - \frac{\rho_a A_c C_d}{2m}(v_h + v_w)^2 - g(\mu_{r_0} + \mu_{r_v})cos(\phi_r) - gsin(\phi_r), \tag{3.13}$$

where R_g is gear ratio, η_p is powertrain efficiency, T_{com} is the commanded torque and other parameters are as defined before. This formulation can be rearranged, as follows:

$$\begin{aligned} sv_h = \frac{R_g \eta_p}{r_w m} T_{com} - \frac{\rho_a A_c C_d}{2m}(v_h)^2 - (\frac{\rho_a A_c C_d}{m} v_w + g\mu_{r_v})(v_h) \\ - g(\phi_r) - \frac{\rho_a A_c C_d}{2m} {v_w}^2 - g\mu_{r_0}. \end{aligned} \tag{3.14}$$

Finally, by using a stable filtering, this model can be reduced to the following model:

$$\frac{s}{s+\lambda} v_h = \theta_1 \frac{1}{s+\lambda} T_{com} - \theta_2 \frac{1}{s+\lambda} {v_h}^2 - \theta_3 \frac{1}{s+\lambda} v_h - \theta_4 \frac{1}{s+\lambda} \phi_r - \theta_5 \frac{1}{s+\lambda}[\mathbf{1}], \tag{3.15}$$

where λ is the stable filter's time constant and $\mathbf{1}$ is the unit input. Therefore, the estimation model is:

$$\hat{a}_h = \hat{\Theta}^T \Phi_d, \tag{3.16}$$

where $\hat{a}_h$ is the estimated acceleration and:

$$\hat{\Theta} = \begin{bmatrix} \hat{\theta}_1 & \hat{\theta}_2 & \hat{\theta}_3 & \hat{\theta}_4 & \hat{\theta}_5 \end{bmatrix}^T, \tag{3.17}$$

which $\hat{\Theta}$ is the vector of the estimated parameters and:

$$\Phi_d = \frac{1}{s+\lambda} \begin{bmatrix} T_{com} & -{v_h}^2 & -v_h & \phi_r & 1 \end{bmatrix}^T, \tag{3.18}$$

where Φ_d is the regressors' vector for the longitudinal dynamic estimator. Equations (3.10) and (3.11) are already in the parametric form of $\dot{m}_f = \hat{A}\Phi_e$ and $\dot{SOC} = \hat{\Gamma}\Phi_m$ with:

$$\begin{aligned} \Phi_e &= \begin{bmatrix} 1 & P_e & {P_e}^2 & v_h \end{bmatrix}^T, \quad \Phi_m &= \begin{bmatrix} 1 & P_m & {P_m}^2 \end{bmatrix}^T, \\ \hat{A} &= \begin{bmatrix} \hat{\alpha}_1 & \hat{\alpha}_2 & \hat{\alpha}_3 & \hat{\alpha}_4 \end{bmatrix}^T, \quad \hat{\Gamma} &= \begin{bmatrix} \hat{\gamma}_1 & \hat{\gamma}_2 & \hat{\gamma}_3 \end{bmatrix}^T. \end{aligned} \tag{3.19}$$

The hat shows the estimated value of a parameter. The presented models in this section will be used for control designs and evaluations in the following sections.

3.3 Control design

This section is devoted to the control design procedure. Figure 3.3 illustrates the proposed Eco-ACC architecture. To design this controller, first, the effective disturbances and uncertainties are analyzed and an additive disturbance term is presented that can capture them. A linear feedback controller (K_c) is designed that stabilizes the system and bounds the effect of additive disturbances on the system's states. Then, the nonlinear T-MPC design procedure is explained that is able to handle the defined constraints in the presence of bounded uncertainties and disturbances. The final control input to the system is generated by combining the designed linear controller with the output of NMPC. Finally, an online least square parameter estimator is presented that estimates the uncertain parameters of the system in real-time. The estimated parameters are used inside the NMPC controller to improve its performance in case of a change in the parameters' values. This way, the final system will be robust to changes in the uncertain parameters and also can adapt to them to improve the control performance. Therefore, the robustness and performance will be separated and can be achieved simultaneously.

3.3.1 Disturbance set

To design a robust MPC, a bound must be established on the states' error caused by disturbances or a robust positive invariant (RPI) set defined below.

Definition 3.1. *For an autonomous system $x[k+1] = Ax[k] + Bw[k]$ with bounded disturbance $w[k] \in \mathbb{W}$, robust positive invariant set Φ is the set of all $x[k] \in \Phi$ such that for all $w[k] \in \mathbb{W}$ and $i > 0$, $x[k+i] \in \Phi$ [101].*

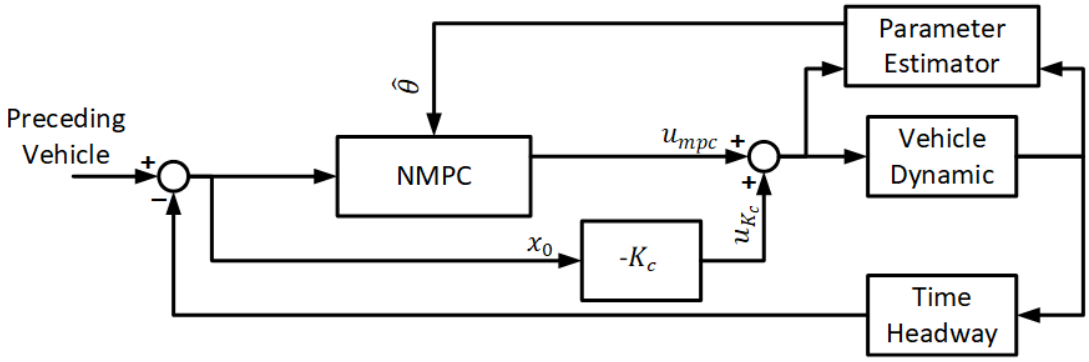


Figure 3.3: AT-NMPC controller architecture

Suppose a nonlinear system in the following format:

$$\begin{aligned} &x[k+1] = Ax[k] + Bu[k] + g(x[k]) + w[k], \\ &\text{subject to} \\ &\quad x[k] \in \mathbb{X}, \\ &\quad u[k] \in \mathbb{U}, \\ &\quad w[k] \in \mathbb{W}_m, \end{aligned} \tag{3.20}$$

where x is the state of the system in discrete time domain, u is input, w is an additive disturbance, $\mathbb{X}$, $\mathbb{U}$, and $\mathbb{W}_m$ are bounds on the state, input and disturbance in the system and $g(x)$ is the nonlinear part of the system. Now suppose that the input to the system has the following form:

$$u_k = -Kx[k - \tau_d] + c_0, \tag{3.21}$$

where K is a linear stabilizing controller, c_0 is the input generated by the model predictive controller and τ_d is the total delay in radar and actuation. We ignore τ_d in the rest of the calculations and model it as part of uncertainty in Proposition 1. By considering this input, Equation (3.20) can be rewritten as

$$x[k+1] = A_c x[k] + Bc_0 + g(x[k]) + w[k], \tag{3.22}$$

where $A_c = A - BK$. On the other hand, without considering the disturbance term, the nominal system can be written as

$$\bar{x}[k+1] = A_c \bar{x}[k] + Bc_0 + g(\bar{x}[k]), \tag{3.23}$$

where $\bar{x}$ is the nominal state. By reducing (3.23) from (3.22) and considering state error as $e = x - \bar{x}$, error dynamics can be defined:

$$e[k+1] = A_c e_k + (g(x[k]) - g(\bar{x}[k])) + w[k] \tag{3.24}$$

An RPI set of this system is equivalent to the maximum error caused by the additive disturbance. To be able to find RPI set of this system, we need to handle the error in the nonlinear term. Authors of [34] showed that if the nonlinear term $g(x)$ is Lipschitz continuous, $\|g(x) - g(\bar{x})\|_2$ will be bounded. If $g(x)$ is Lipschitz in the region $x \in \mathbb{X}$ then:

$$\|g(x) - g(\bar{x})\|_2 \leq L \|x - \bar{x}\|_2 , \quad \forall x_1, x_2 \in \mathbb{X}, \tag{3.25}$$

where the smallest L satisfying this condition is the Lipschitz constant. Now if $L(\mathbb{X})$ is the Lipschitz constant over $\mathbb{X}$ then it can be obtained from (3.25) that

$$\begin{aligned} \forall x, \bar{x} \in \mathbb{X} \ \& \ e \in \mathbb{E}, \ \|g(x) - g(\bar{x})\|_\infty \\ \leq L(\mathbb{X}) \max_{e \in \mathbb{E}} \ \|e\|_2 , \end{aligned} \tag{3.26}$$

where $\mathbb{E}$ is a subset of $\mathbb{X}$ which includes the origin. This inequality defines a boxed shaped set that bounds the error in the nonlinear term.

$$\mathbb{W}_g = \{\zeta \in \mathbb{R}^n | \ \|\zeta\|_\infty \leq L(X) \max_{e \in \mathbb{E}} \ \|e\|_2\}, \tag{3.27}$$

which can be added to $\mathbb{W}_m$ to make $\mathbb{W} = \mathbb{W}_g \oplus \mathbb{W}_m$. Basically, if a bound can be defined on the nonlinear term in the constraints region, then we can consider it as part of the additive disturbance. The next step is to find a bound for $\mathbb{W}_m$. To be able to use this method, all sources of uncertainty must be combined into a single additive disturbance. One major cause of uncertainty on this system is the delay in feedback loop due to τ_d. This uncertainty can be bounded by finding the maximum state change that can happen in the maximum delay time. The following proposition explains the calculation of this bound.

Proposition 3.1. *Let $x[k] \in X, u[k] \in U, a_p[k] \in A_p, w[k] \in W$ where all of the sets X, U, A_p, W are bounded. Furthermore, assume that the radar and actuator delay is upper-bounded by a sufficiently small T_d, i.e., $0 \leqslant \tau_d \leqslant T_d$. Then w_τ, the uncertainty caused by delay, will be bounded by the set:*

$$W_\tau = T_d B_d K_c \times \{AX \oplus (BU \oplus (EA_p \oplus W))\} .$$

Proof. In order to prove this proposition, we use the fact that the difference between $x[k]$ and $x[k-\tau_d]$ is given by the rate of changes of x in τ_d-duration multiplied by τ_d (assuming that τ_d is small). Rigorously

$$x[k-\tau_d] = x[k] - \tau_d \dot{x}[k].$$

Next, note that according to (3.8), the set of all possible state change rates ΔX can be given by:

$$\begin{aligned}\Delta X = \{\dot{\mathbf{x}} \mid \dot{\mathbf{x}} = A\mathbf{x} + Bu + B_p a_p + w,\\ \forall \mathbf{x} \in X, \forall u \in U, \forall a_p \in A_p, \forall w \in W\},\end{aligned}$$

which, looking back at the preliminary definitions, is equivalent to the following Minkowski sum:

$$\Delta X = \{AX \oplus (BU \oplus (B_p A_p \oplus W))\}.$$

Therefore, by Equation (3.14), the total amount of uncertainty that delay produces in the system is given by

$$W_\tau = T_d B_d K_c \times \{AX \oplus (BU \oplus (B_p A_p \oplus W))\},$$

which is what we aimed to show. □

Another source of uncertainty is the acceleration of the preceding vehicle. In (3.8) the preceding vehicle's acceleration a_p has been modeled as an additive disturbance. Therefore, w_{a_p} which is the uncertainty caused by a_p can be bounded by knowing a bound for maximum possible acceleration for the preceding vehicle.

$$w_{a_p} \in \mathbb{W}_a = B_p A_p. \tag{3.28}$$

The uncertain parameters can also increase the model uncertainty. Uncertainty in vehicle mass, tire radius, drag coefficient, rolling resistance coefficients, road grade, wind speed and powertrain efficiency must be considered in the vehicle control design. If a bound for each of these uncertain parameters is available, Equation (3.13) can be used to find the maximum model error that the uncertain parameters can cause. Suppose $\Upsilon = \begin{bmatrix} m & r_w & C_d & \mu_{r_v} & \mu_{r_0} & \phi_r & v_w & \eta_p \end{bmatrix}$ as the vector of uncertain parameters in Equation (3.13) with $\overline{\Upsilon}$ as the vector of their nominal value and Υ_{max} and Υ_{min} as the vector of their maximum and minimum values. Then, the following optimization problem will find

the maximum model error.

$$e_{a_{min}} = \min_{\Upsilon, v_h, T_{com}} a_h(\Upsilon, v_h, T_{com}) - a_h(\overline{\Upsilon}, v_h, T_{com}),$$
$$e_{a_{max}} = \max_{\Upsilon, v_h, T_{com}} a_h(\Upsilon, v_h, T_{com}) - a_h(\overline{\Upsilon}, v_h, T_{com}),$$

subject to

$$\Upsilon_{min} \leq \Upsilon \leq \Upsilon_{max},$$
$$0 \leq v_h \leq v_{h_{max}},$$
$$T_{min} \leq T_{com} \leq T_{max}.$$

where $e_{a_{min}}$ and $e_{a_{max}}$ are the minimum and maximum errors caused by the parameter uncertainty which based on them the set of all possible acceleration errors due to the parameter uncertainty can be defined as: $e_a \in E_a$, and also bounded additive disturbance due to the parameter uncertainty can be calculated, as follows:

$$w_{a_h} \in \mathbb{W}_h = BgE_a, \tag{3.29}$$

where B_g is as defined in (3.8). The combination of all the uncertainty sources will be the bounded additive disturbance term.

$$\mathbb{W} = \mathbb{W}_g \oplus \mathbb{W}_\tau \oplus \mathbb{W}_a \oplus \mathbb{W}_h. \tag{3.30}$$

This disturbance set will be used for the design of the tube-based controller. Equation (3.24) can be rewritten as

$$e[k+1] = A_c e[k] + w[k] \qquad w \in \mathbb{W}. \tag{3.31}$$

Using this stable model with a bounded additive disturbance and Minkovski sum, the finite reachable set for the error can be calculated as given below:

$$\Phi_n = \oplus_{i=0}^{n} {A_c}^i \mathbb{W}, \tag{3.32}$$

where Φ_n is the finite reachable error set and its infinity limit Φ_∞ is called the robust positive invariant set [102]. In this research, our T-MPC is similar to [103] which uses a finite invariant set instead of infinity RPI set with a fixed current state.

3.3.2 Model adaptation

A model adaptation method is used to adapt to changes in the system and environment to maintain the performance of designed controllers. Here, we use a least square parameter adaption method with a forgetting factor similar to [88] with the same notation. This method uses previously presented parametric models to estimate the value of each effective parameter. It works based on minimizing the squared error between the estimated and measured output of the system by minimizing the following cost function.

$$J(\hat{\theta}) = \frac{1}{2}\int_0^t \frac{e^{-\beta(t-\tau)}(z(\tau) - \hat{\theta}^T(t)\phi(\tau))^2}{m_s^2(\tau)}d\tau \; + \; \frac{1}{2}e^{-\beta t}(\hat{\theta}(t) - \hat{\theta}_0)^T Q_0(\hat{\theta}(t) - \hat{\theta}_0), \tag{3.33}$$

where $\hat{\theta}$ is the estimated parameters vector, $\hat{\theta}_0$ is the initial estimated parameter, ϕ is the measured input signal, z is the measured output signal, β is a forgetting factor, Q_0 is a weighting matrix, P is covariance matrix and m_s^2 is a normalizing term that can be chosen as: $m_s^2 = 1 + \alpha\phi^T\phi, \quad \alpha \geq 0$. By minimizing this cost function, the algorithm can find an estimation of the parameters. The first term penalizes the estimation error and the second term penalizes the convergence rate with a decaying factor to increase estimation robustness against disturbances. Forgetting factor gives a higher weight to the new measurements so that in case of a change in a parameter the algorithm can adapt to it. Based on this cost function a recursive least square algorithm is defined as follows.

$$\begin{aligned}
\dot{\hat{\theta}}(t) &= P(t)\epsilon(t)\phi(t), \\
\dot{P}(t) &= \beta P(t) - P(t)\frac{\phi(t)\phi^T(t)}{m_s^2(t)}P(t), \\
\epsilon(t) &= \frac{z(t) - \hat{\theta}^T(t)\phi(t)}{m_s^2(t)}.
\end{aligned} \tag{3.34}$$

This algorithm updates the covariance and estimated parameters online when the vehicle is running. To prevent wrong estimations, it is necessary to limit the estimated parameters. Therefore, parameter projection is used to put constraints on estimations. Moreover, to avoid the covariance matrix from becoming very large, it is necessary to put a constraint on its maximum value. Assuming the desired constraint on the parameters is defined by: $S = \{\theta \in R^n | g(\theta) \leq 0\}$, where g is a smooth function and R_0 as an upper bound for P,

projection can be defined as follows.

$$\dot{\theta} = \begin{cases} P\epsilon\phi & \text{if } \theta \in S^o \text{ or} \\ & \theta \in \delta(S) \;\&\; (P\epsilon\phi)^T \nabla g \leq 0 \\ P\epsilon\phi - P\frac{\nabla g \nabla g^T}{\nabla g^T P \nabla g} P\epsilon\phi & otherwise \end{cases}$$

$$\dot{P} = \begin{cases} \beta P - P\frac{\phi\phi^T}{m_s^2} P & \text{if } \|P\| \leq R_0 \;\&\{\theta \in S^o \text{ or} \\ & \theta \in \delta(S) \;\&\; (P\epsilon\phi)^T \nabla g \leq 0\} \\ 0 & otherwise \end{cases} \tag{3.35}$$

Projection ensures that the estimation will not go out of the constraint region and will move along the border when it reaches its limits. This adaptation algorithm is used to estimate fuel consumption, electricity consumption and longitudinal dynamics parameters based on the reduced model presented in the modeling section. The estimated parameters will be used in the control-oriented model of AT-NMPC so that the optimization problem will find the updated optimal point of the system.

3.3.3 Adaptive robust controller

Using the disturbance set and the parameter adaption method above, it is possible now to define our adaptive robust control problem. This controller includes a linear controller that stabilizes the system and an NMPC that controls the system based on its nominal control-oriented model and without considering the uncertainty and disturbances. The NMPC keeps the nominal state of the system in a tighter region to ensure that the actual system states will remain inside the defined constraints. Moreover, parameter adaptation updates the control-oriented model that is used in the definition of the cost function to maintain system performance. Therefore, two control-oriented models are used here, one for updating the cost function and performing a future prediction and another one for handling the constraints. The AT-NMPC optimization problem is defined as follows:

$$\min_{c_o}\Big\{\sum_{i=1}^{Np}\big(\omega_1 e_p^2(\hat{x},\hat{u})+\omega_2 e_v^2(\hat{x},\hat{u})+\omega_3\hat{u}^2+\omega_4 E_{cost}(\hat{x},\hat{u},PR,\hat{A},\hat{\Gamma})\big)\Big\},$$

subject to

$$\begin{aligned}
&\bar{x}[n]=x[n],\quad \hat{x}[n]=x[n],\\
&\hat{x}[n+i+1]=\hat{f}_n(\hat{x}[n+i],c_0[n+i]),\\
&\bar{x}[n+i+1]=\bar{f}_n(\bar{x}[n+i],c_0[n+i]),\\
&\hat{u}[n+i]=-K_c\hat{x}[n+i]+c_0[n+i],\\
&\bar{x}[n+i+1]\in X\ominus\Phi[i],\\
&c_0[n+i+1]\in U\ominus(-K_c\Phi[i]),
\end{aligned}\tag{3.36}$$

where $\bar{f}$ and $\hat{f}$ are the nominal and estimated nonlinear longitudinal dynamic model of the vehicle, N_p is the prediction horizon's length, ω_1, ω_2, ω_3, and ω_4 are weights on each term and other parameters are as defined before. The control problem finds a vector c_0 that minimizes the cost function in the prediction horizon while the actual input to the system is the combination of c_0 and the linear controller.

Remark 3.1. *In this control problem, the current state of the system is not a decision variable and $x[n]$ has a fixed value. Both nominal and adapted control-oriented models start from the same initial point but perform future predictions based on their own parameters.*

Remark 3.2. *The tighter constraints on the states ensure that the system states will remain inside X, the defined state constraint, for any amount of disturbance that satisfies $w\in\mathbb{W}$. Moreover, the tighter constraints on the input reserves a part of available actuation for the linear controller to maintain system's robustness.*

Remark 3.3. *Having two separate control-oriented models ensures that constraints are always satisfied based on the fixed nominal control-oriented model. Therefore even if the adapted control-oriented model has low accuracy, the system's robustness will be maintained. This is especially important in the case that a sudden change in model parameters occur. Because the parameter estimator may not be able to recognize the change in the model immediately, it is necessary to make sure that the system will remain safe and robust while the model is getting adjusted which is achievable by using separate models as has been done here.*

3.3.4 Fast optimizer

To implement the proposed robust Eco-ACC on a vehicle control system, the AT-NMPC problem must be solved in real-time. Therefore a fast solver is required that can solve the nonlinear optimization problem with little computational demand. To this end, in this research, Newton/GMRES method has been used to solve the control optimization problem. This method is claimed to be very fast as it solves the differential equation once at each time step [104]. In the current study, the author uses an automatic multi-solver NMPC code generator, called MPSee, to generate the NMPC code based on the Newton/GMRES algorithm which has been previously developed and tested in the author's research group [105],[106]. MPSee is a MATLAB-based mathematical program that enables users to develop GMRES-based NMPC codes for different optimal control problems and carry out simulations in Simulink. To define the Newton/GMRES solver, field vector and constraints have been defined as follows:

$$f(x,u) = \frac{d}{dt}\left(\begin{bmatrix} \bar{p}_h \\ \bar{v}_h \\ \bar{T}_w \\ \bar{p}_p \\ \bar{v}_p \\ \bar{a}_p \\ \hat{p}_h \\ \hat{v}_h \\ \hat{T}_w \end{bmatrix} \right) = \begin{bmatrix} \bar{v}_h \\ \frac{\bar{T}_w}{m r_w \eta_p} - \{\frac{\frac{1}{2}\rho_a C_a A_w}{m}\bar{v}_h^2 + g\bar{\phi}_r + \mu_v g \bar{v}_h + g\mu_0\} \\ -\frac{\bar{T}_w}{\tau_a} + \frac{u - K[\bar{e}_p \;\; \bar{e}_v \;\; \bar{T}_w]}{\tau_a} \\ \bar{v}_p \\ \bar{a}_p \\ -\sigma \bar{a}_p \\ \hat{v}_h \\ \Theta_1 \hat{T}_w - \Theta_2 \hat{v}_h^2 - \Theta_3 \hat{\phi}_r - \Theta_4 \hat{v}_h - \Theta_5 \\ -\frac{\hat{T}_h}{\tau_a} + \frac{u - K[\hat{e}_p \;\; \hat{e}_v \;\; \hat{T}_h]}{\tau_a} \end{bmatrix} \tag{3.37}$$

$$C(x,u) = H_x[i] \begin{bmatrix} \bar{e}_p[i] \\ \bar{e}_v[i] \\ u[i] - K[\bar{e}_p[i] \;\; \bar{e}_v[i] \;\; \bar{T}_w[i]] \end{bmatrix} - h_x[i] \tag{3.38}$$

where $H_x[i]$ and $h_x[i]$ are used to define the polytopic constraints in each step equivalent to $X \ominus \Phi[i]$ in Equation (3.36). The two separate control-oriented models have been implemented in here in the field vector and constraints where barred variables show the nominal values and hatted variables are the estimated ones. σ is the decaying factor for the preceding vehicle's acceleration as defined in [22].

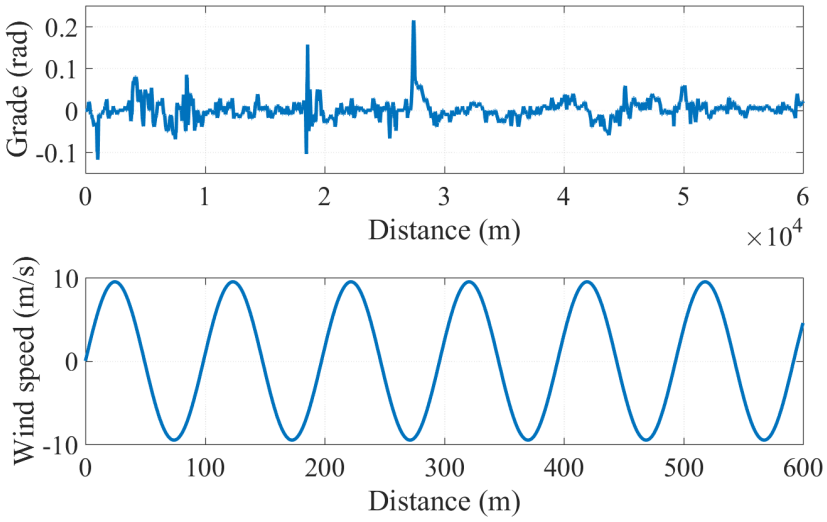


Figure 3.4: Grade and wind profile injected in the simulation environment

3.4 Control evaluation

This section presents the evaluation of the proposed Eco-ACC in terms of estimation, robustness and ecological improvement. The high-fidelity model of the baseline vehicle, developed in Autonomie, is used for evaluation tests. First, the performance of the three estimators is presented. Second, the robustness of the proposed controller is tested using the high-fidelity model of the baseline vehicle with injected disturbances as shown in Figure 3.4. Third, ecological improvement caused by the proposed Eco-ACC is discussed. Finally, the result of HIL experiment is presented that shows the real-time implementation capability of the proposed controller. The weighings in these section for the controller was chosen as: $\omega_1 = 0.25$, $\omega_2 = 1$, $\omega_3 = 4.5$ and $\omega_4 = 100$. The LQR controller was chosen as: $K_c = [-1.225,\ -1.224,\ 0.336]$.

3.4.1 Parameter estimation

We used three least-square parameter estimators to improve the control-oriented model of our predictive controller. The first estimator gets velocity, road grade, and propulsion or braking torque and then finds the parameters of the longitudinal model based on (3.15). Figure 3.5 (a-e) illustrate the results of online parameter estimation for this estimator

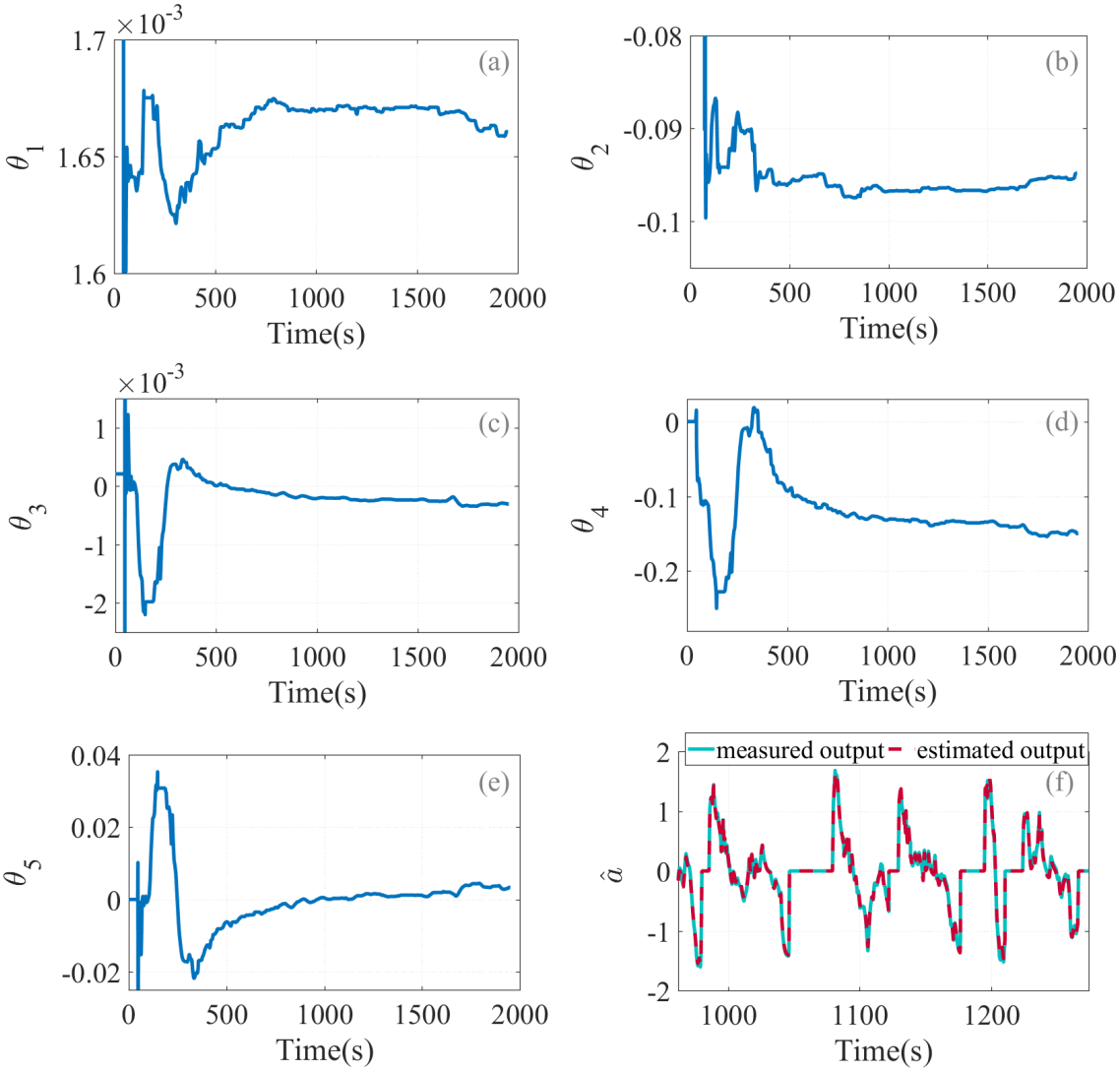


Figure 3.5: Longitudinal dynamic parameter estimator: (a) $\theta_1[\frac{1}{Ns^2}]$, (b) $\theta_2[\frac{1}{m}]$, (c) $\theta_3[\frac{1}{s}]$, (d) $\theta_4[\frac{m}{s^2}]$, (e) $\theta_5[\frac{m}{s^2}]$, and (f) $\hat{a}[\frac{m}{s^2}]$.

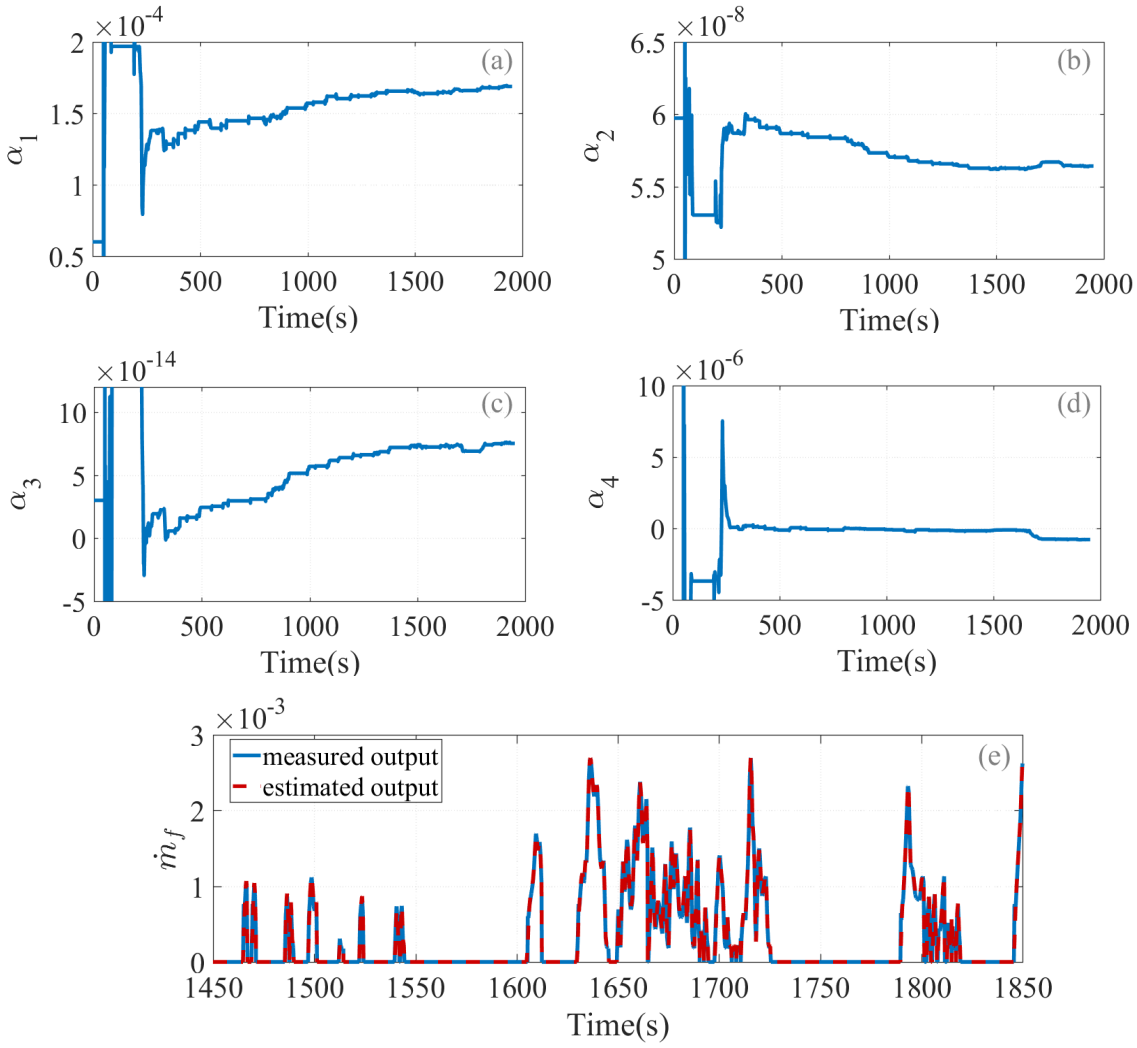


Figure 3.6: Fuel consumption parameter estimator: (a) $\alpha_1[\frac{kg}{s}]$, (b) $\alpha_2[\frac{kg}{Nm}]$, (c) $\alpha_3[\frac{kgs}{N^2m^2}]$, and (d) $\dot{m}_f[\frac{kg}{s}]$.

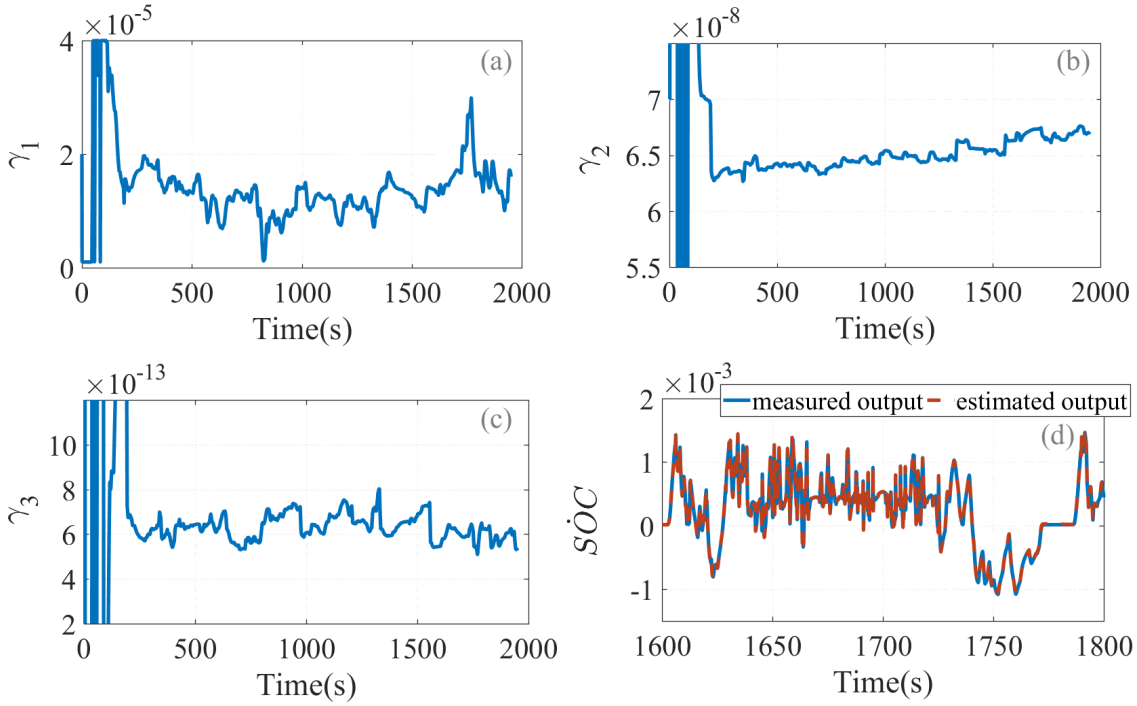


Figure 3.7: Electricity rate parameter estimator: (a)$\gamma_1[\frac{1}{s}]$, (b)$\gamma_2[\frac{s^2}{kgm^2}]$, (c) $\gamma_3[\frac{s^5}{kg^2m^4}]$, and (d) $\dot{SOC}[\frac{1}{s}]$

and Figure 3.5 (f) shows the estimated acceleration by the estimated model compared to the measured acceleration. It is worth noting that the estimated parameters may not converge to their real values which, for adaptive control use, is fine as long as the estimated output matches the real value [88]. This model predicts the motion of the vehicle inside the prediction horizon and adjusts to changes so that the prediction will be more accurate. The two other estimators are shown in Figure 3.6 for fuel consumption and in Figure 3.7 for electricity. As it can be seen both models are able to estimate the desired output closely and adapt to changes in the parameters. The initial guess for the estimators is chosen arbitrary which is the reason for initial oscillations in the parameter estimations. To improve the estimators' performance, forgetting factor has a higher value in the beginning and decreases gradually afterward to a lower value. The main objective of using estimators is for making sure that the optimizing the defined cost function will optimize the actual system. Therefore, the results of these estimators will be used in the cost function of the optimization problem.

3.4.2 Robust constraint handling

Other than online parameter adaptation, the proposed AT-NMPC based Eco-ACC is able to handle bounded uncertainties. To show the validity of this statement, we conduct simulations by utilizing a high-fidelity model of the baseline vehicle and then adding a variable wind speed and road grade to the simulation environment as shown in Figure 3.4. During the simulation, the host vehicle follows a preceding vehicle in a drive cycle by receiving inter-vehicular distance and velocity from the radar. To simulate the effect of delay in the radar's data, 400ms transport delay was injected into the inter-vehicular distance and velocity data during the simulation. Moreover, parametric errors were considered in the control-oriented model by 20% error in the vehicle mass, 50% error in the drag coefficient and ignoring the rolling resistance forces. Then, based on the method presented in Section IV, disturbance sets were calculated and used for defining the constraints of the AT-NMPC problem.

Figure 3.8 and 3.9 shows the result of the simulation in a standard FTP-75 drive cycle. The preceding vehicle follows the given drive cycle and the host vehicle follows the preceding vehicle during the simulation using a regular MPC-based ACC and proposed AT-NMPC Eco-ACC. As shown in Figure 3.8 (a) both controllers have acceptable velocity tracking performance while following the preceding vehicle. However, the NMPC-based ACC has harsher accelerations compared to the proposed Eco-ACC. The harsher accelerations are due to the fact that non-robust NMPC cannot handle the defined constraints due to the existing uncertainties in the system. Therefore, these uncertainties can push the system out of the defined constraints and the NMPC has to perform harsh braking and acceleration to go back into the constraints. However, AT-NMPC is robust against these uncertainties and therefore it has less harsh accelerations. Figure 3.9 (a) and (b) compare the velocity and position tracking error of the two controllers. NMPC is not able to handle the defined constraints and it has higher position errors than the defined limits. On the other hand, AT-NMPC handles the constraint perfectly because the effect of uncertainties has been considered in its design based on Equation (3.36).

3.4.3 Ecological improvement

The objective function of the proposed controller is defined to minimize the cost of energy in a driving cycle. Based on the given dynamic model of the vehicle and available road elevation, AT-NMPC predicts the future host vehicle's trajectory and based on that calculates the expected power demand in the prediction horizon. Then, based on the power-ratio

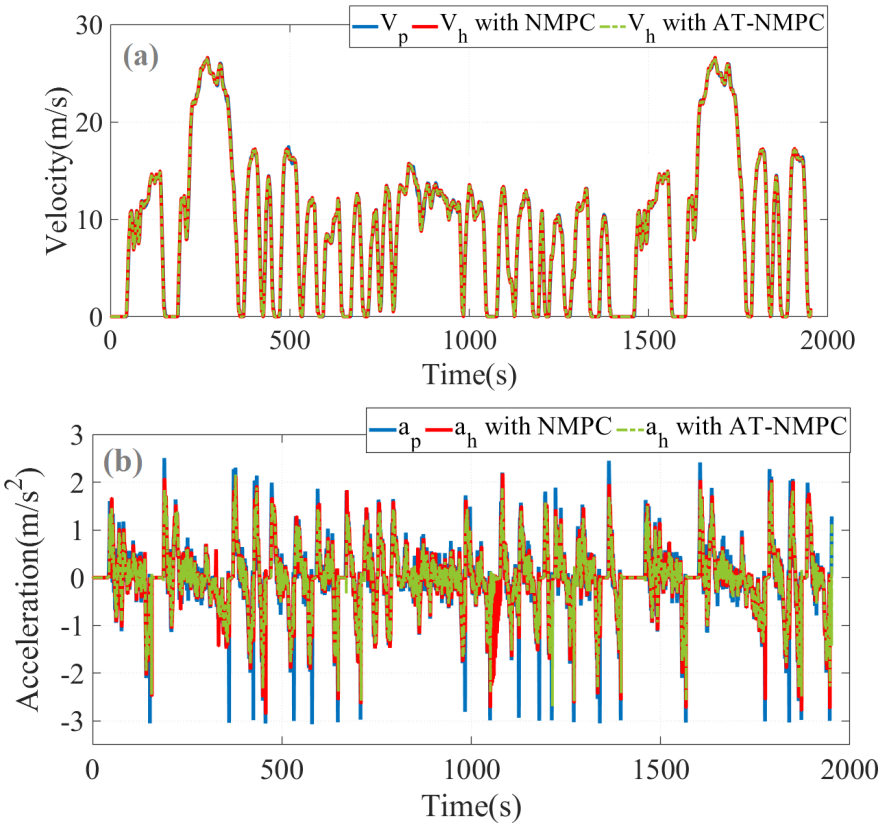


Figure 3.8: Comparison of AT-NMPC vs. NMPC in (a) velocity and (b) acceleration in car-following simulation

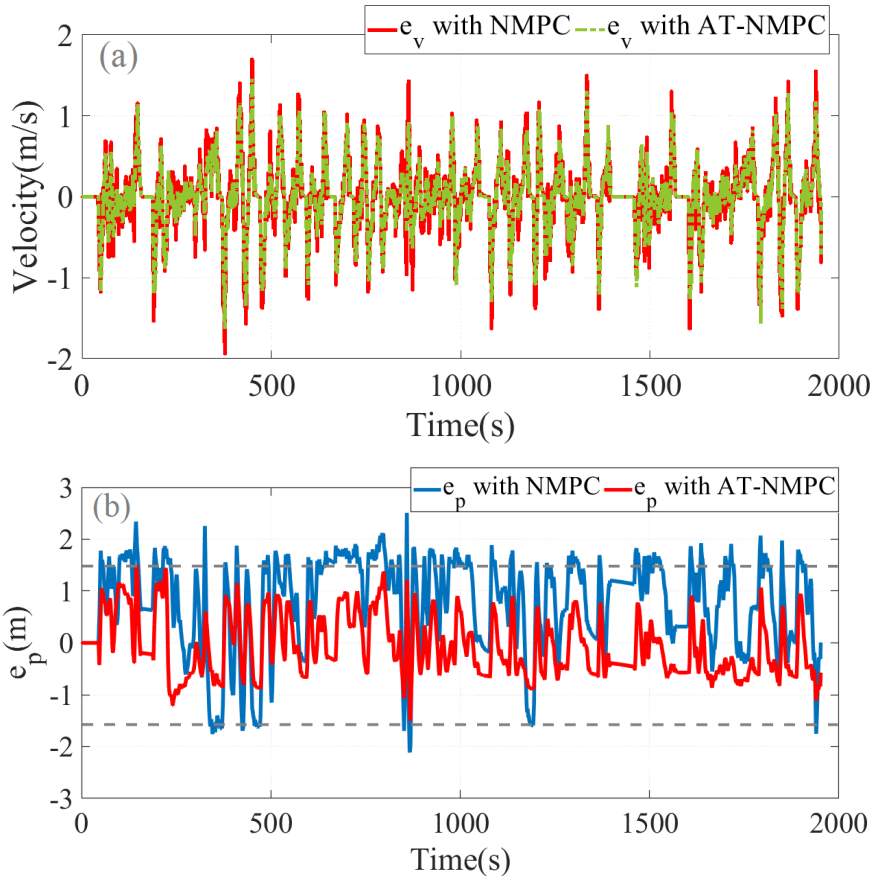


Figure 3.9: Comparison of AT-NMPC vs. NMPC in (a)velocity tracking error, and (b)position error in car-following simulation

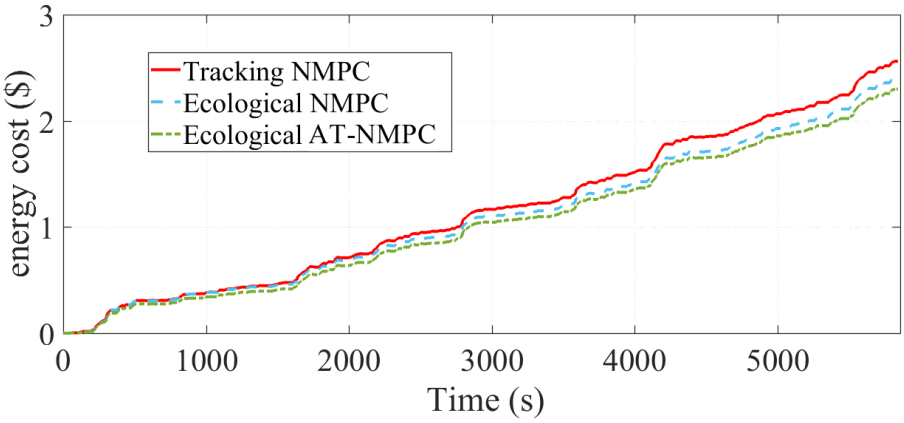


Figure 3.10: Energy consumption in three consecutive FTP-75 drive cycles

of the energy management system, it calculates the power demand of each energy source and also energy cost during the prediction horizon. Parameter estimators make sure that the optimal point of the cost function is the actual optimal point of the system. Figure 3.10 compares the trip energy cost of a tracking NMPC, ecological NMPC and the proposed AT-NMPC in three consecutive FTP-75 drive cycles. A longer drive cycle is chosen to minimize the effect of energy management system in the achieved results. The tracking NMPC has higher weightings on position and velocity tracking terms to increase the tracking performance. Therefore it mostly sacrifices energy cost for better tracking and therefore it has the highest energy cost in this driving cycle by $2.56. The Eco-NMPC has a higher weighting on energy cost term which means that it sacrifices tracking inside the defined constraints, to have better energy cost. Therefore, it has an energy cost of $2.4 which is about 6.2% lower than the tracking NMPC. However, in this case, the NMPC controller is not able to handle the defined constraints and uncertainties push the system out of the defined constraints set. To be able to move back into the constraints, the NMPC has to do harsh braking and accelerations which increase the energy cost. AT-NMPC, on the other hand, handles the defined constraints and also minimizes a cost function that adapts to the actual vehicle behavior. It has an energy cost of $2.29 which is about 10.5 % lower compared to the tracking NMPC.

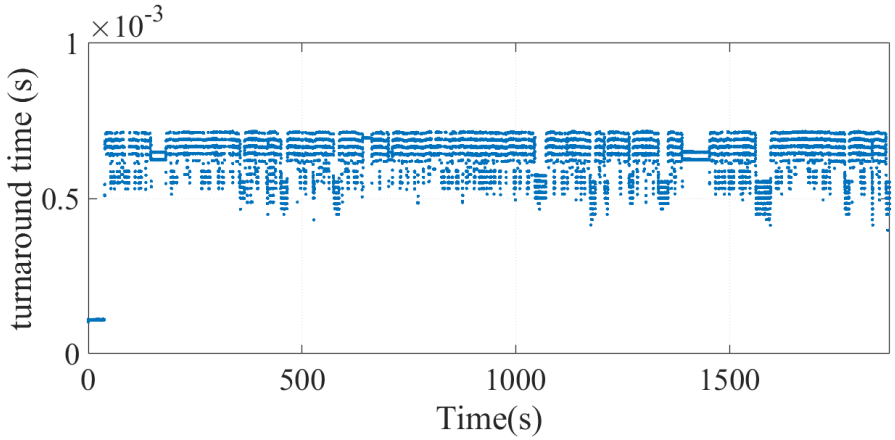


Figure 3.11: Turnaround time of the controller in FTP-75 HIL experiment

3.4.4 Real-time implementation

As mentioned before, in this research, to further examine the performance of the proposed controller, HIL experiments have been conducted to study the potential and capability of AT-NMPC for real-time implementations in a vehicle control system. HIL tests take into account the computational limits and communication issues, and their result is considered more practical than MIL simulations. Because physical prototyping in the early stages of development of vehicle control systems can be very expensive, HIL experiments, which are less expensive and also faster and safer, usually carried out before manufacturing the prototype vehicle [97]. Here, dSPACE Micro-Autobox II control prototyping hardware was used for HIL tests. This setup is one of the widely used instruments for calibration and testing of ECUs specifically for automotive applications. As explained in the previous chapter and shown in Figure 2.3, the HIL setup has three main components: 1) a prototype ECU (MicroAutoBox II), which is an independent processing module that runs the uploaded controller; 2) a real-time simulator (DS1006 processor board), which is responsible for running the complex high-fidelity model; and 3) a PC that serves as the human-machine interface. All the communications between the prototype ECU and the real-time simulator are performed through a CAN bus. The computational power of the ECU and also communication limits of the CAN bus impose a hard constraint on the design of the controller in terms of accuracy and complexity of the control-oriented model, number of variables, number of the prediction steps, size of the sampling time and the optimization method. A control engineer must consider and adjust these parameters in the design of the controller to be able to achieve real-time implementation capability.

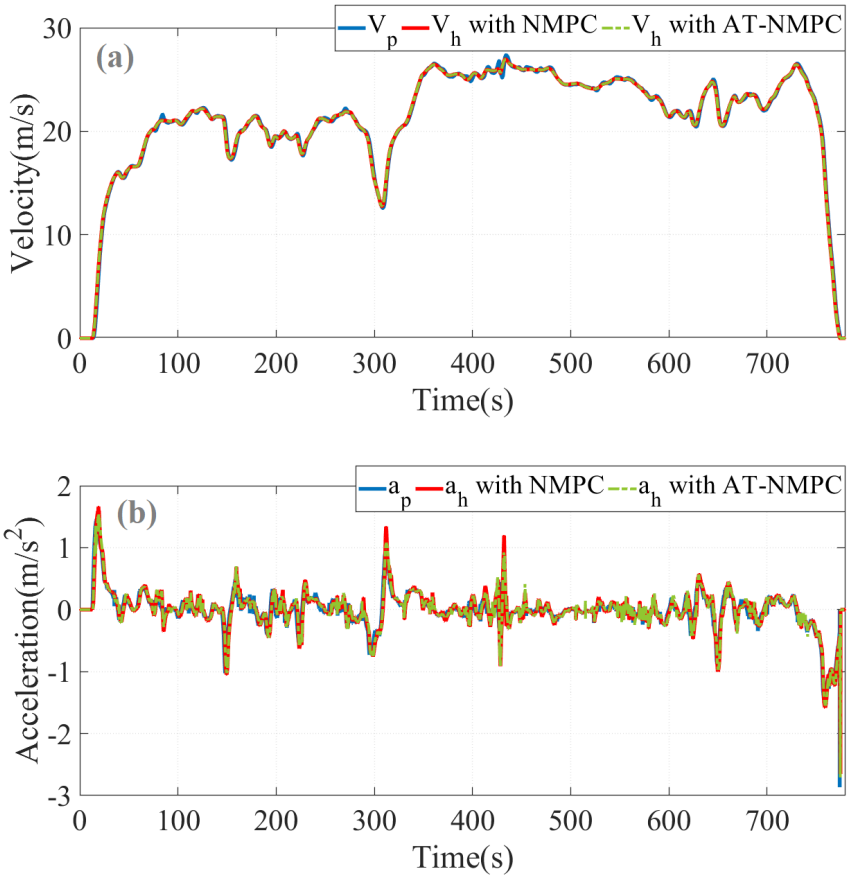


Figure 3.12: (a) velocity and (b) acceleration during car-following in a HWFET standard driving cycle

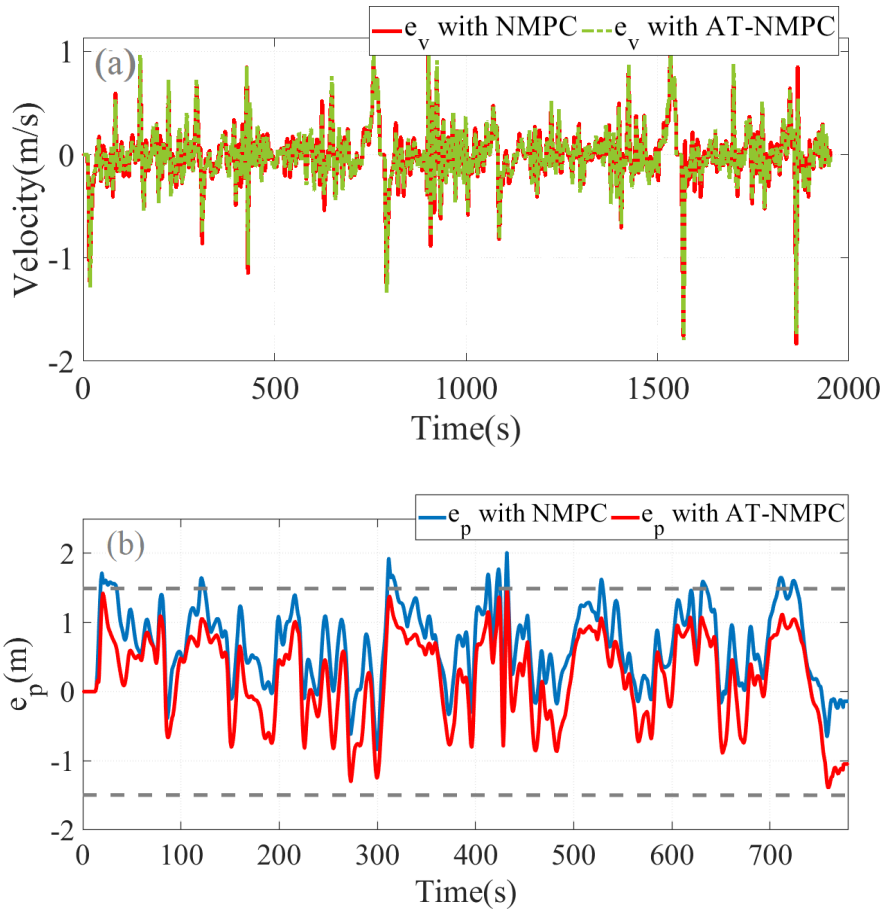


Figure 3.13: (a) velocity error and (c) position error during car-following in a HWFET standard driving cycle

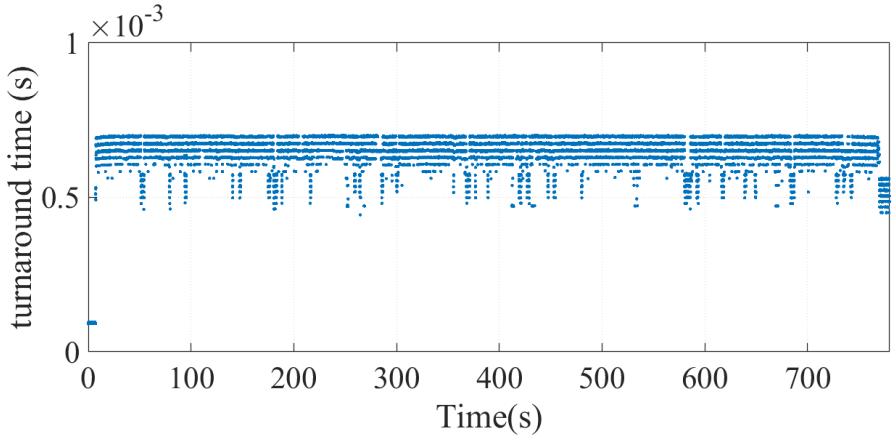


Figure 3.14: Turnaround time of the controller in HWFET driving cycle HIL experiment

For HIL tests, a C code is generated of the designed controller by using the dSPACE Real-Time Workshop code generator and then uploaded to the prototype ECU. With a similar procedure, the high-fidelity model is uploaded to the real-time simulator using the human-machine interface. The main goal of the HIL test is to determine the turnaround time of the controller on a vehicle ECU. If the resulted turnaround time from HIL experiments is less than the specified sampling time for the controller, there will be no concerns about iteration limits and time-delay caused by the computational cost of the optimization problem.

The results of the controller performance in terms of control task in a FTP-75 driving cycle has been presented in the last section. Figure 3.11 illustrates the turnaround time of the controller in FTP-75 driving cycle. The turnaround time for this controller is between $400\mu s$ and $700\mu s$ at all times for a prediction horizon of $N_p = 10$. The maximum inner and outer iterations set to be less than 5 to enable real-time implementations [105]. The sampling time of the controller is 1 ms; hence, the proposed controller can be implemented using typical automotive ECU hardware with no concern about computational demands and without causing time delays. To further examine the performance of the proposed controller in real-time, another experiment has been performed in a standard highway driving cycle, HWFET. Figure 3.12 shows the results of the controller's performance and Figure 3.14 shows the controller's turnaround time during the HIL experiment in the considered highway driving cycle. Due to the existing uncertainties and delayed data, the NMPC method was not able to maintain the defined constraints. AT-NMPC, however, is robust to the uncertainties and was able to handle the defined constraints during this

experiment. The turnaround time of the controller is below the specified 1ms sampling time limit, which means that the proposed controller can be executed on a vehicle ECU with any sampling time higher than 1ms.

3.5 Summary

This chapter proposed an adaptive and robust tube-based nonlinear model predictive controller for the design of ACC systems to enable two-vehicle car-following. This method ensures the robust satisfaction of the defined constraints in the presence of uncertainty and also improves the system's performance by adapting to the changes in the vehicle control-oriented model. Therefore, in a way, this method decouples performance and robustness by using separate models one for constraint handling and another one for defining the objective function.

In the modeling step, a nonlinear control-oriented model was presented for a vehicle that performs car-following. This model was used for evaluation of the safe sets in the presence of additive disturbances. Moreover, models for fuel consumption and electricity rate were presented to estimate the cost of energy in the prediction horizon and based on them, reduced models for parameter estimation were generated. A high-fidelity model of the baseline PHEV, Toyota plug-in Prius, was used to evaluate the controller.

In the control design step, first, a linear controller was used to stabilize the system. Then by analyzing the existed uncertainties, they got translated into an additive disturbance term to define a bound for maximum uncertainty. Next, the design of the three least square parameter estimators was explained that estimate the parameters of the control-oriented model. Then, AT-NMPC control problem was defined that uses two separate models, one for defining the objective function and another one for constraint handling. Using separate models ensures that constraints are handled based on the fixed nominal control-oriented model while the objective function is defined based on an adapted control-oriented model to ensure that the optimal point of the cost function and the actual system are equivalent. The objective function of the controller was defined to minimize the energy cost while following a preceding vehicle.

In the controller evaluation step, simulations on the high-fidelity model were performed by injecting uncertainties and delay into the simulation environment. The controller evaluations showed that the proposed AT-NMPC is able to handle the defined constraints in the presence of uncertainty while improving the trip energy cost by 10% compared to a

tracking NMPC. Finally, HIL experiments were conducted to show the real-time capability of the proposed controller which showed that AT-NMPC had low computation costs while running on a prototype ECU.

The AT-NMPC based Eco-ACC developed in this chapter is a time-domain controller. Although this controller can significantly improve energy economy and maintain robust safety of the system, because of its time-domain nature, it can not enforce frequency-domain criteria. Hence, this controller can not be used for achieving string stability in a vehicle platoon. Therefore, in the next chapter, we present a reference governor approach to vehicle platooning that can achieve time-domain and frequency-domain criteria at the same time. The AT-NMPC controller developed in this chapter can be used for single car following and also control of the platoon's leader, in order to improve platoon's energy economy.

Chapter 4

Distributed reference governor approach to safe and ecological platooning

In this chapter, a predictive platooning controller is presented to achieve Eco-CACC for a string of vehicles. We develop a predictive reference governor that enforces constraints on a string stable platoon. Since the focus of this section is on platooning, we assume that the leader is not performing car-following and instead it is following a drive cycle. Therefore, a predictive controller based on NMPC is presented to control the platoon's leader to improve the energy economy of the whole platoon. For a leader that performs car-following, the controller in the previous chapter should be used. The proposed controllers are fine-tuned for the baseline vehicle and will be evaluated using an Autonomie-based, high-fidelity model of this vehicle, which was discussed before. Although the control design and evaluations in this chapter are performed on the Toyota Prius PHEV, the proposed approach can be applied to other vehicle types with a few adjustments and simplifications.

The main contribution of this chapter is the design of a platooning controller that can simultaneously achieve string stability in the frequency-domain and constraint handling in the time-domain. The RG approach is proposed to separate platooning time-domain requirements from frequency-domain requirements. The proposed predictive RG in this chapter can handle time-varying references unlike classical RG methods, which is another contribution of this research. This design is separate from the rest of the control system and does not interfere with the main control task.

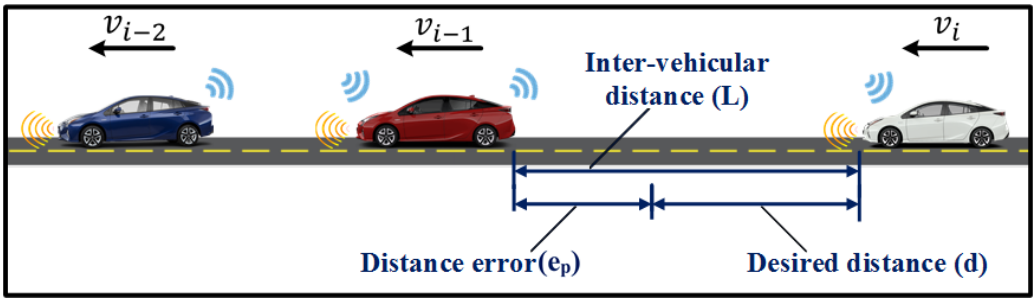


Figure 4.1: Car-following parameters in a vehicle platoon

The remainder of this chapter is organized as follows: Section 4.1 presents the vehicle and platoon models used in the control design and evaluation. Section 4.2 is devoted to the control design procedure for achieving string stability, constraint handling, and improved energy cost. The simulation of the proposed controllers for evaluating the string stability, constraint handling, energy cost improvement and real-time implementation are presented in Section 4.3. Finally, Section 4.4 summarizes and concludes the contributions of this study.

4.1 Modeling

This section presents a control-oriented model developed for the proposed predictive controllers. Different platoon models can be created, depending on the architecture of the communication network between the vehicles. Here, we consider communications only between a follower vehicle and its immediate predecessor. However, this method can be applied to any other CACC system with different network topologies. Moreover, in this chapter, we assume that all vehicles are identical.

4.1.1 Platoon model

Figure 4.1 shows three vehicles in a platoon with effective car-following parameters and vehicle indexes. Obtaining the desired distance is the goal of car-following which will be achieved by correcting an error equal to e_p. The goal of platooning controller is to find control inputs that can minimize this error. To design our model-based controller, a simple model of the platoon is required. To model our platoon, we must first specify a car-following rule that specifies the desired spacing between vehicles.

There are different policies in the research literature for achieving the desired inter-vehicular distance. Here, we choose a constant time headway rule: $d = d_0 + h_i v_i$, where d_0 is the standstill vehicle gap and h_i is the constant headway-time. With this gap policy, we can write the state-space equations in the following form:

$$\begin{aligned} e_{p_i} &= p_{i-1} - p_i - h_i v_i \\ \dot{e}_{p_i} &= e_{v_i} - a_i h_i \\ \dot{e}_{v_i} &= u_i - \frac{1}{m} F_r - a_{i-1} \\ F_r &= \frac{1}{2}\rho_a A C_d (v_i)^2 + \mu m g cos(\theta_g) + m g sin(\theta_g) \end{aligned} \tag{4.1}$$

where a_i, v_i and p_i are acceleration, velocity, and position, e_{p_i} is the error for inter-vehicular distance, e_{v_i} is the velocity difference, F_r is the resistance force, ρ_a is the air density, A is the frontal area of the car, C_d is the drag coefficient, μ is the rolling resistance coefficient, m is the vehicle's mass and θ_g is the road grade. u is the input which is calculated from the wheel torque by: $u = \frac{T_w}{m r_w}$, where T_w is the wheel torque and r_w is the wheel radius. To model the response of each vehicle, we assume a first-order model with a constant delay [75]:

$$u_i = \frac{K_i}{\eta_i s + 1} e^{-\tau_i s} u_{d_i} \tag{4.2}$$

where η_i is the actuator's time constant, τ_i is the actuator's delay, K_i is the steady-state gain and u_{d_i} is the desired input. The nonlinear model presented here has been used in the NMPC-based control of the leader vehicle. To design our reference governor, we need a linear platoon model:

$$\begin{aligned} \dot{x}_i(t) &= A x_i(t) + B a_{d_i}(t - \tau_i) + B_p a_{i-1}(t) \\ A &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -h_i \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & \frac{-1}{\eta_i} \end{bmatrix}, \; B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{K_i}{\eta_i} \end{bmatrix}, \; B_p = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \\ x_i &= \begin{bmatrix} \varsigma_{e_{p_i}} & e_{p_i} & e_{v_i} & a_i \end{bmatrix}^T \end{aligned} \tag{4.3}$$

where ς_{e_p} is the integral of inter-vehicular distance error that has been included for the control design and a_{d_i} is the desired acceleration which is related to the desired input by: $u_{d_i} = a_{d_i} + \frac{F_r}{m}$. We are assuming that the resistance force F_r is compensated with a feed-

forward compensator. Similar models have been used widely in the literature [60],[76]. In this model, the preceding vehicle's acceleration is an additive disturbance acting on the dynamic system.

4.1.2 Powertrain model

High-fidelity model

The high-fidelity model of the Toyota plug-in Prius, which was explained in chapter 3, is used here to develop a platooning high-fidelity model. To construct a platoon model, we used a combination of six identical high-fidelity vehicle models in Simulink. This model has accurate component models and mappings for different subsystems in the vehicle and it is a reliable tool for evaluating the performance of the controllers in the vehicle's longitudinal motion. Each vehicle uses the proposed controllers to follow its predecessors in the platoon. The high-fidelity platoon model is used for evaluating the proposed controllers in terms of string stability, handling of the constraints and energy economy.

Control-oriented model

The high-fidelity model has high accuracy, but it is very complex; therefore, it cannot be used inside the proposed controllers. Instead, a control-oriented model needs to be developed that is able to capture the dominant dynamics of the vehicle but has lower complexity. We need a simple model for the engine, electric motors, and battery. An energy management controller in hybrid vehicles distributes the power demand between the vehicle's energy sources to keep the powertrain system near its optimal working condition. Therefore, it can be assumed that the engine is always operating at its optimum working point. Similar to last chapter, the fuel consumption can be approximated with the following equation [53]:

$$\dot{m}_{f_i} = a_o + a_1 P_{e_i} + a_2 P_{e_i}^2 + b_1 v_i \tag{4.4}$$

where $\dot{m}_{f_i}[\frac{L}{s}]$ is the fuel rate, $P_{e_i}[W]$ is the engine power and $a_0[\frac{L}{s}]$, $a_1[\frac{L}{J}]$, $a_2[\frac{L.s}{J^2}]$ and $b_1[\frac{L}{m}]$ are constant coefficients. A PHEV has gasoline and electricity as its two energy sources. Therefore, it is necessary to define the trip cost based on the energy cost of both sources:

$$E = -\frac{C_f \dot{m}_{f_i}}{v_i} - \frac{C_e P_{bat_i}}{\eta_{ch}\eta_{dis} v_i} \tag{4.5}$$

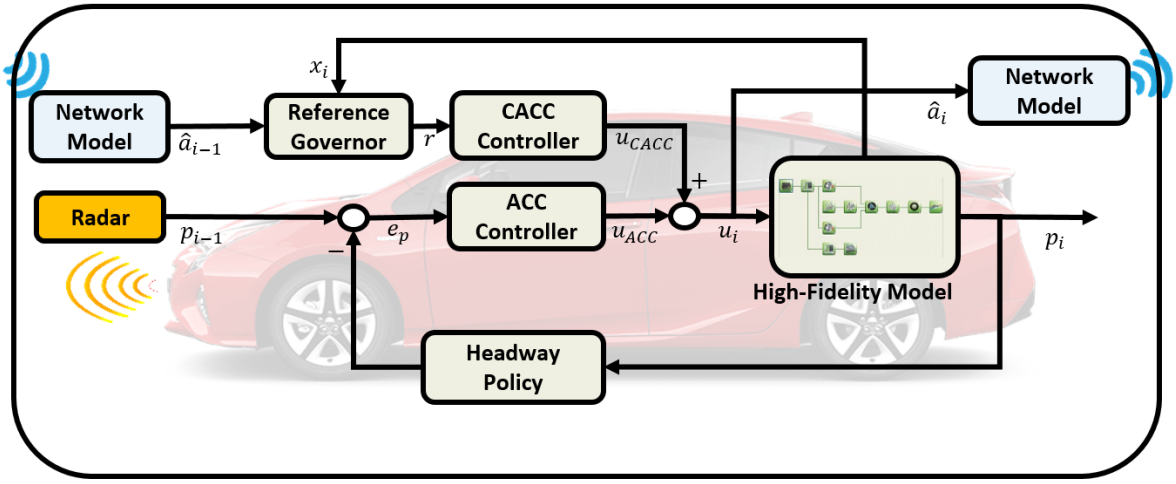


Figure 4.2: Platooning controller architecture

where E is the energy cost, C_f and C_e are the gasoline and electricity cost, respectively, P_{bat} is the battery power, η_{ch} and η_{dis} are the charging and discharging battery efficiency, respectively. The energy cost has been divided by the velocity to eliminate the effect of traveled distance. At lower velocities, the energy cost will be assumed constant to avoid singularities.

4.2 Control design

This section presents the control objectives and the mathematical procedure of controller design. There are three control objectives that must be achieved simultaneously: string stability, constraint handling, and ecological performance improvement. In this chapter, we limit the wireless communication network to the follower-predecessor and, based on that, we assume the control architecture shown in Figure 4.2. The radar provides the inter-vehicular velocity and distance and the wireless V2V communication gives the acceleration set-point of the preceding vehicle. Combined ACC and CACC controllers will produce acceleration set-points to control the vehicle. In this architecture, ACC is responsible for maintaining the desired inter-vehicular distance and car-following, while CACC improves string stability of the system. Although we only consider follower-predecessor communications, the proposed method can be applied to the platoons with different types of network

topology. However, in this chapter, we are limiting our discussion to this specific type of network topology and will discuss network topology in the next chapter.

4.2.1 String stability

One of the widely discussed topics in platooning is string stability. A string stable platoon attenuates disturbance signals along the string of vehicles, whereas the string unstable platoon amplifies them. The string instability will result in high oscillations in upstream traffic and can cause traffic congestion or even accidents. For instance, in a string unstable platoon, if the leader starts braking to stop, the next car will have attenuated signals and therefore will brake harsher. The same behavior will continue in the next vehicles and each follower will brake harsher compared to its preceding. Therfore, after a number of vehicles, there will not be enough braking force to stop the follower cars and collision will be inevitable. Therefore, it is important to consider this criterion in the controller design. It is possible to define string stability based on the attenuation of different types of signals. For example, reference [60] defined string stability based on the transfer function between the acceleration of a follower to the predecessor and reference [76] defined it based on the transfer function between the velocities. Here, we adopt the method in [76] but consider acceleration signals for each follower-predecessor pair. The following criteria will ensure the system's string stability:

$$\|\mathcal{S}(j\omega)\|_\infty = \sup_{\omega\in\mathbb{R}} \left\| \frac{a_i(j\omega)}{a_{i-1}(j\omega)} \right\|_\infty \leq 1 \tag{4.6}$$

where $\mathcal{S}$ is the string stability signal, a_i is the follower's acceleration, a_{i-1} is the predecessor's acceleration and $\|.\|_\infty$ is the infinity norm. The criterion requires that the transfer function between the acceleration of a vehicle and its predecessor must be less than or equal to one at all frequencies. Based on Figure 4.2, two controllers must be designed to stabilize the system. Here, we use a PID controller for ACC since it is easy to tune and provides a good tracking performance.

$$u_{d_i}^{acc} = \begin{bmatrix} K_I & K_P & K_D & -K_D h \end{bmatrix} x_i \tag{4.7}$$

where K_I,K_P and K_D are integral, proportional and derivative coefficients and u_d^{acc} is the generated input by the ACC controller. Similar controllers have been used in the literature before [58],[60],[76]. In the presented control architecture, the acceleration set-points by CACC are actually supplementary to the main ACC system to improve its performance. For a homogeneous platoon of vehicles which have the same response, the

CACC transfer function can be equal to one and in non-homogeneous platoons, CACC adapts the acceleration set-point to the dynamics of the host vehicle. To improve the stability of the system and reduce the effect of disturbances on its performance, a filter is also considered in the CACC controller.

$$C_{CACC} = \frac{1 + \eta_{i-1}s}{(1 + \eta_i s)(1 + \eta_c s)} \tag{4.8}$$

where η_{i-1} and η_c are the preceding vehicle and filter's time constants.

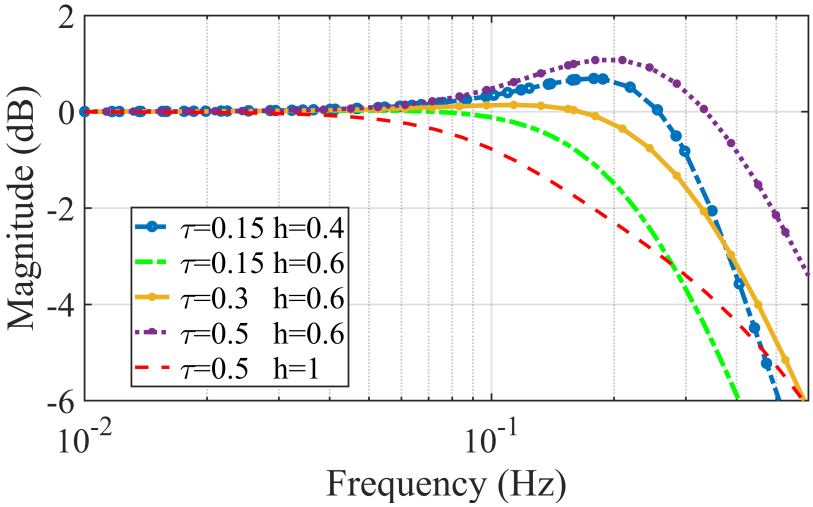


Figure 4.3: String stability transfer function for different values of delay ($\tau[s]$) and headway time ($h[s]$).

As with many other systems, the major factor affecting string stability is the delay in radar and wireless communications. To achieve a string stable platoon, we design a PID controller to stabilize the system and minimize the distance error, then by tuning the PID controller and changing the headway time, study the string stability of the platoon for different values of delay. Figure 4.3 illustrates the frequency response of the string stability signal for different delays and headway times. The increased delay results in a larger magnitude in string stability signal, which means that the upstream vehicles will oscillate with a higher amplitude at higher delays. Increasing the headway time will improve the string stability and for each amount of the wireless delay there is a stabilizing minimum headway time. Here, we have considered 150ms second delay in communication and a headway time equal to 0.6 seconds.

4.2.2 Platooning constraint handling

String stability is not sufficient for safe and high-performance platooning. To ensure the vehicles' safety and satisfactory performance, they must be able to handle constraints. The usual constraints in platooning are as follows:

Safety constraint

One of the platoon's objectives is to minimize inter-vehicular distances to increase road throughput and to decrease drag while keeping a minimum distance for the safety. Thus, a maximum and minimum inter-vehicular distance error can be defined. Also, in platooning, it is necessary to have the minimum platoon length and based on that, a maximum inter-vehicular distance can be defined:

$$e_{p_{min}} \leq e_p(t) \leq e_{p_{max}}. \tag{4.9}$$

Limitation constraint

In real-world applications, there are always constraints on the actuation limit. It is the same in platooning, with constraints on the maximum propulsion and maximum braking force:

$$u_{min} \leq u_i(t) \leq u_{max}. \tag{4.10}$$

The vehicles' maximum and minimum allowable velocity are always limited and the limitation changes based on the route type and road condition:

$$v_{min} \leq v_i(t) \leq v_{max}. \tag{4.11}$$

Performance constraint

Each vehicle's velocity needs to be close to the platoon's velocity set-point; a large velocity difference is undesirable:

$$e_{v_{min}} \leq e_v(t) \leq e_{v_{max}}. \tag{4.12}$$

To make sure that all vehicles deliver the same range of acceleration in a platoon and to consider the comfort of the passenger in the design of the platoon, a constraint on the

acceleration must be defined:

$$a_{min} \leq a_i(t) \leq a_{max}. \tag{4.13}$$

These constraints ensure the performance and safety of platooning. It is necessary to employ a controller that is able to handle the platooning constraints. The string stability has a frequency-domain definition and combining it with constraint handling, which has a time-domain definition, is not trivial in a single controller. Here, to enforce the constraints, we have designed an RG that, if necessary, changes the reference to enforce the constraint handling. The main benefit of RG is that it separates the control design from the constraint handling, allowing classical popular control approaches to be used with no concern about the constraints. This benefit makes RG very useful in real-world applications since most control engineers are more comfortable with classical controllers, which are easier to tune and have been used for many years. In our case, in addition to the mentioned benefits, we are seeking to enforce the constraints on a string stable platoon without interfering with the main platoon control. RG works based on the system's output admissible set.

Definition 4.1. *(Output Admissible Set) The output admissible set is the concatenated set of states and constant references that, if given to the system, will never violate the constraints on the output in the presence of a bounded disturbance [67]:*

$$O_\infty = \{ \, (x(t), r(t)) \mid y(t+k) \in Y, \quad \forall w(t+k) \in W, \; r(t+k) = r, \; \forall k \geq 0 \}. \tag{4.14}$$

In this definition, O_∞ is the output admissible set, r is the constant reference, W is the disturbance bound and Y is the constraint on the output. Based on this set, RG will modify the reference to the system. If the original reference and states are inside O_∞, RG will not change the reference and, if necessary, RG will find the nearest reference that keeps the constraints. The problem with the classic RG is that it is assuming a constant reference, which limits its performance. Also, in real-world applications, $O_\infty(x(t))$ can become void, which means there may be no constant reference that could maintain the constraints on the output. In this case, it is more practical to use a reference management, which is an extended version of RG. Reference management can work with void admissible sets by pushing the system toward admissibility in finite time. We define the new RG problem as presented in Algorithm 1. In Algorithm 1, s is a slack variable, ϵ_{RG} is a design parameter, τ shows the time in the prediction horizon, y_i is the output, $\bar{y}_d$ is the desired output, T is the terminal time, $\hat{a}_{i-1}$ is the estimated acceleration set-point that comes through V2V communication and r is the generated reference by RG. $\bar{\hat{a}}_{i-1}$, $\bar{y}_i$ and $\bar{r}$ respectively show the vector of the predicted preceding vehicle's acceleration, predicted output and vector

Algorithm 1 Reference Management Problem

Input: $\{\hat{a}_{i-1},\ x_i,\ y_i\}$
Output: $\{r\}$

if $\hat{a}_{i-1} \notin O_\infty(x_i)$ **then**
  $activeRG \leftarrow True$
end if
if $activeRG == False$ **then**
  $r \leftarrow \hat{a}_{i-1}$
else
  $\min_{\bar{r}^T} \{\|\bar{y}_i - \bar{y}_d\|_{2,Q} + \|\bar{r} - \bar{\hat{a}}_{i-1}\|_{2,R} + \|s\|_{2,P}\}$
   $s.t : x_i(t+T) \in O_\infty(\hat{a}_{i-1})$
    $y_i(t+\tau) + s \in Y,\ \ \tau = \{0, 1, ..., T-1\}$
  $r \leftarrow \bar{r}(1)$
  if $\hat{a}_{i-1} \in O_\infty(x_i)$ & $\|r - \hat{a}_{i-1}\| < \epsilon_{RG}$ **then**
    $activeRG \leftarrow False$
  end if
end if

of decision variables which its first component is given to the system as a reference value: $r = \bar{r}(1)$. Q, R, and Prepresent the weightings on the output error, reference error, and the slack variable. RG will determine if $\hat{a}_{i-1}(t) \in O_\infty(x_i(t))$, in this case, it does not change the original reference. But if $\hat{a}_{i-1}(t) \notin O_\infty(x_i(t))$, or if $O_\infty(x_i(t))$, is void, RG will move the system states toward the admissible set $O_\infty(\hat{a}_{i-1}(t))$ in a finite time while maintaining the constraints. The first term in the objective function of RG is the tracking term and the second term is the reference keeping term.

Remark 4.1. *The RG added to the control system, will not change the autonomous system's behavior when it recognizes that the constraints will not be violated. Therefore, if our original system is string stable, the new platooning system will remain string stable. In a situation where the constraints can be violated, RG will change the reference to keep the defined constraints and push the system toward the admissible set so it can give the original reference to the system and preserve the string stability. This will provide both string stability and constraint handling for the platoon.*

Remark 4.2. *The design parameter ϵ_{RG} is set to avoid large changes when switching back to the original reference. A higher value of ϵ_{RG} will reduce the time window of active RG*

but might result in an oscillatory behavior.

Remark 4.3. *T, the minimum terminal time that can satisfy the condition: $x_i(T) \in O_\infty(\hat{a}_{i-1})$, will be calculated using the method in [107]. To ensure the satisfaction of the terminal constraint, the terminal time must be set to a number larger than the minimum value.*

Remark 4.4. *For a polytopic Y, an inner polytopic approximation of O_∞ is available with the following form:*

$$\tilde{O}_\infty = \{(x_i(0),\ r)|\ H_x x_i(0) + H_r r \leq c\} \tag{4.15}$$

where H_x, H_r, and c are the polytope's parameters. The accuracy of the approximation can be controlled by modifying a small value for approximation criteria [67].

Figure 4.2 shows the location of RG in our platoon control architecture. RG will get the acceleration set-point of the preceding vehicle and, if the current states of the system are in the admissible set, will not change the acceleration set-point. In the case of constraint violation, RG will move the system toward admissibility in a finite time by modifying the original reference. When the states are back in the output admissible set, RG will give the original reference to the system. This way, the platoon will be string stable and RG will only intervene when there is a chance of violating the constraints. To solve the problem stated in the Algorithm 1, we used CVX, a package for specifying and solving convex optimizations [108],[109].

4.2.3 Ecological improvement

The ecological performance improvement is done by the control of the lead vehicle. A smoother speed trajectory for the lead vehicle will result in lower energy costs for the whole platoon. The assumption for this section is that the leader is doing the car-following task because its preceding vehicle is neither connected nor part of the platoon. Because the leader's controller is not considered in the frequency-domain string stability analysis, a time-domain approach can be used to control it. The following problem is defined for

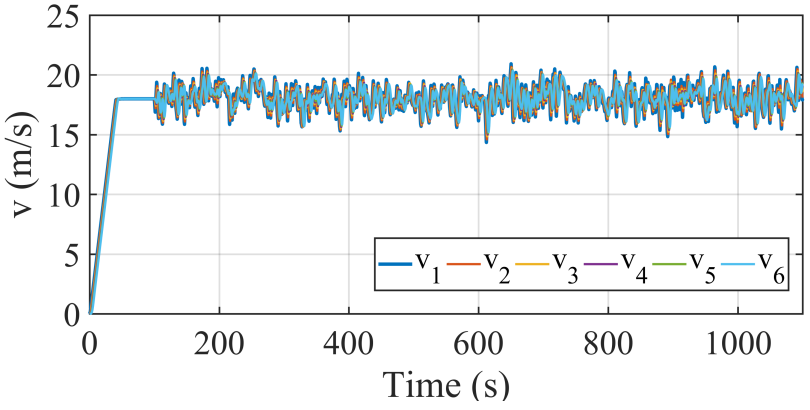
minimizing the energy cost of its car-following task [53]:

$$\begin{aligned} \mathcal{J} &= \int_{t_k}^{t_k+T} (w_1 e_p^2 + w_2(v - v_{ref})^2 + w_3 E(x, u, \mathcal{PR}))dt \\ &\text{subject to:} \\ &\dot{x}_i = \mathcal{F}(x_i, u, a_{i-1}) \\ &v_{min} \leq v_i(t) \leq v_{max} \\ &u_{min} \leq u_i(t) \leq u_{max} \end{aligned} \tag{4.16}$$

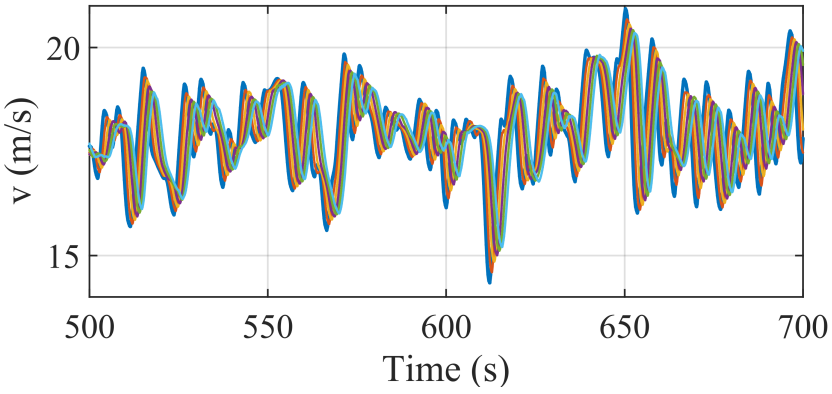
where a_{i-1} is the acceleration of the preceding vehicle. Inside the prediction horizon, a prediction of the preceding vehicle's acceleration is needed. Investigating standard drive cycles shows that vehicles tend to move with low accelerations and at constant velocities [53], therefore, we used the following prediction, with an exponential decay in the acceleration:

$$\bar{a}_{i-1}(t) = a_{i-1}(t_k)e^{-\lambda(t-t_k)}. \tag{4.17}$$

This model generates a prediction of the preceding vehicle's acceleration over the prediction horizon. We are assuming that the preceding vehicle will reach a constant velocity and λ is a parameter for modifying this prediction. t_k is the current time and t is the prediction time. In Equation (4.16), E is the energy cost as defined in the Eq.(4.5), $\mathcal{PR}$ is the power ratio between the electric motors and the engine and T is the length of the prediction horizon that is chosen based on its effect on the energy economy and computation time [53,110]. This cost function penalizes the inter-vehicular distance, velocity error and energy cost. We assume a constant power ratio in the prediction horizon that is equal to $\mathcal{PR}$ and specified by the energy management controller at the current time instance. Therefore, the cruise controller will interact with the energy management system through this variable and different energy management techniques will result in separate ecological performances for the cruise control. The leader uses the control-oriented model to predict the future states of the system in the given prediction horizon. Based on this prediction, the objective function is evaluated and minimized in each time step to find the optimal control values. Then, in each time step, the first calculated input is given to the system as the control input.



(a)



(b)

Figure 4.4: (a) Identification signal, with rich frequency content, used for string stability identification of the platoon. (b) close-up view of the identification signal. This velocity profile was given as the reference to the platoon's leader.

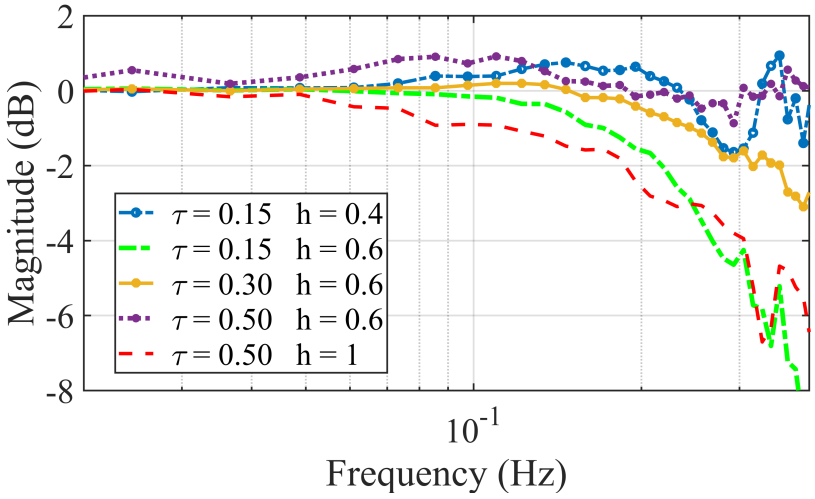


Figure 4.5: Identification results for string stability of the platoon with different headway times ($h[s]$) and communication delays ($\tau[s]$)

4.3 Control evaluation

In this section, the evaluation of the proposed controller is presented for string stability, constraint handling and ecological performance improvement. A platoon of vehicles was constructed in Simulink by combining six high-fidelity models of the baseline PHEV. The leader is controlled using the proposed NMPC for ecological car-following and the rest of the vehicles form a platoon that follows the leader. Each vehicle communicates with its immediate predecessor through V2V communications. It should be mentioned that, for the simulations, a linear Adaptive Equivalent Consumption Minimization Strategy (A-ECMS) energy management method was used to optimize the power distribution between the vehicles energy sources [111], [112]. This method maintains a linear state-of-charge (SOC) profile during the trip based on the available route length. Energy management system is not a focus of this research; therefore it is not discussed here. In this section the control parameters are chosen as: $h = 0.6$, $\lambda = 0.3$, $\tau_w = 150ms$, $\omega_1 = 0.4$, $\omega_2 = 1$, $\omega_4 = 500$ and $\epsilon_{RG} = 0.025$.

4.3.1 String Stability

Since the reference governor does not change the regular behavior of the system, the platoon will remain string stable. To show the validity of this statement, we have performed a string stability identification test on the vehicle platoon. This test will determine if the designed controllers can achieve string stability for a high-fidelity platoon model. To evaluate the designed platoon's string stability, we gave the leader a filtered white noise signal added to a constant velocity as the reference. Then, we measured the vehicles' response for 1000 seconds and investigated the platoon's string stability by calculating the frequency response between the lead and tail vehicles. To estimate the string stability frequency response function (FRF) from time-domain experimental data, it is necessary to use an excitation signal that is able to excite the frequency range of interest [76], [113]. Therefore, we chose a colored noise signal, generated by filtering a white noise signal that has a reach frequency content and it is appropriate for assessments of the FRF of the system. Figure 4.4 shows the reference signal used for system identification. The transfer function estimation was performed by the tfestimate command in Matlab with 16384 windows and 50% overlap. Figure 4.5 illustrates the magnitude of the transfer function generated in the identification test between the accelerations of the lead and tail vehicles. The frequency analysis shows a result similar to that in Figure 4.3, for different values of delay and headway time. Using this method, we choose a constant headway time equal to 0.6 seconds for the presented simulations and considered a wireless delay equal to 0.15 seconds in the V2V communications.

4.3.2 Constraint handling

The next step after designing the string stable platoon is to enforce the defined constraints on the system. As explained above, the RG approach was used for constraint handling. RG works based on the output admissible set; therefore, the first step in designing RG is calculating this set. The discretized version of the model presented in Equation (4.3) is used to calculate the output admissible set. A polytypic inner approximation of the output admissible set was generated using the method presented in [66], [67]. Figure 4.6 illustrates projections of three states of the output admissible sets for three different values of the preceding vehicle's accelerations. Figure 4.6 is only generated to show the general behavior of the output admissible set. The constraints that we assumed, in this case, are on the position error, velocity error, and acceleration limits. The velocity profile in Figure 4.7 is used, therefore, a constraint on the distance error is chosen as: $e_p \in [-0.6\ 0.75]$ to

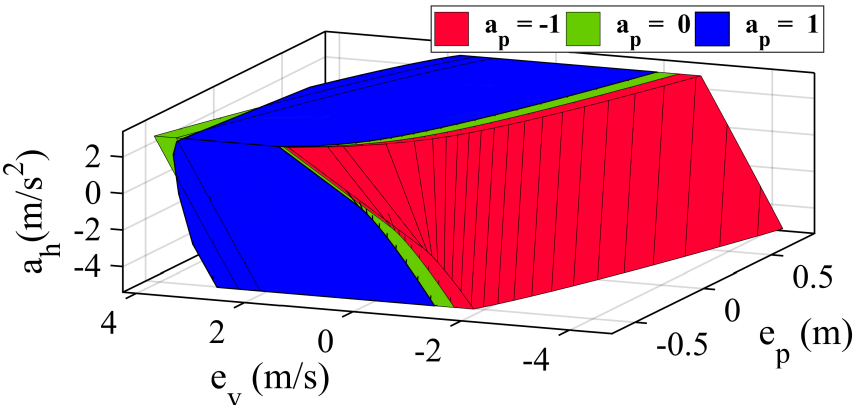
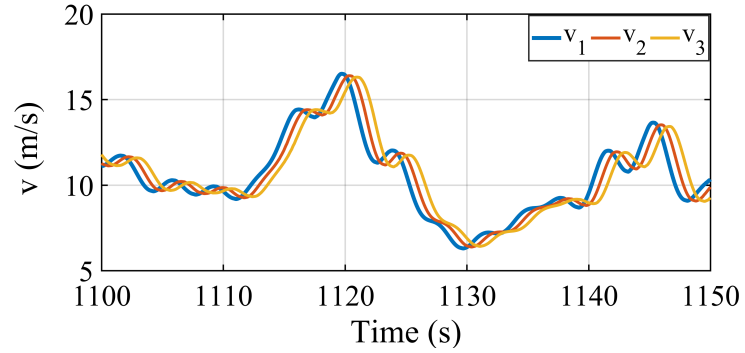


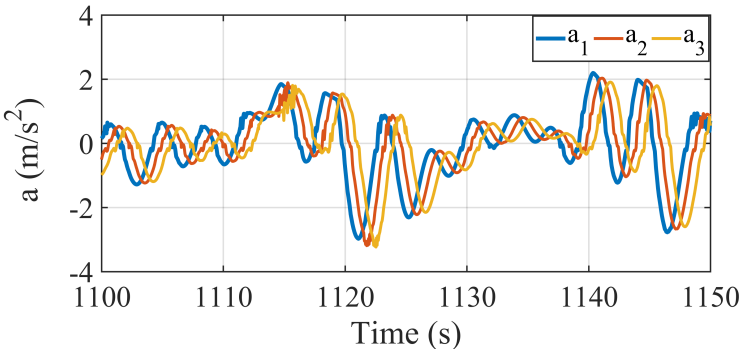
Figure 4.6: Projection of the calculated output admissible set for three different preceding vehicle's acceleration set-points. In this plot, the blue color has priority to green and green has priority to red. Therefore, the overlapping areas are shown with the color of higher priority.

avoid infeasibility. The constraint on the inter-vehicular velocity is chosen as $e_v \in [-5\ 5]$ to avoid large velocity difference but allowing vehicles to have slightly different speeds. The acceleration limit was chosen as: $a_i \in [-5\ 2.5]$ based on the available actuation in the vehicle. The output admissible set contains the points in the state space that are safe in terms of handling the constraints for the specified reference value. Therefore, if, for example, the preceding vehicle's acceleration set-point a_p is equal to -1 and states are in the red region, the defined constraint will not be violated. However, if the system's states are not in the red region, giving a_p as the reference to the CACC controller will result in the violation of the defined constraints. In this case, RG becomes active, moves the states back into the red region and then becomes inactive.

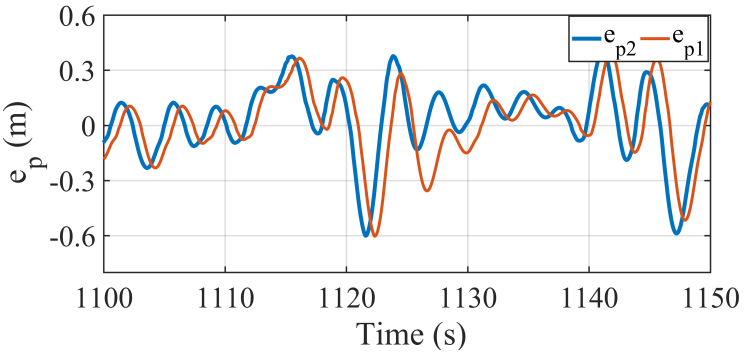
RG pushes the system states inside the presented output admissible set to enforce the defined constraints. To evaluate the performance of the RG, the same platoon as the previous section is used in Simulink. To have clearer test figures, only the results of the first three platoon vehicles are plotted in this section. Figure 4.7 illustrates the performance of the designed platoon in following a reference vehicle. It can be seen that the platoon is string stable and there is less oscillation for the follower vehicle compared to the proceeding ones. The vehicles follow each other by a reasonable acceleration and velocity. Figure 4.7 (c) shows the inter-vehicular distance error between the three vehicles. The distance error is mostly well behaved and inside the defined constraints, only in 1122 and 1147 seconds, the distance error reaches the lower limit of -0.6m. Figure 4.8 shows the reference generated



(a)

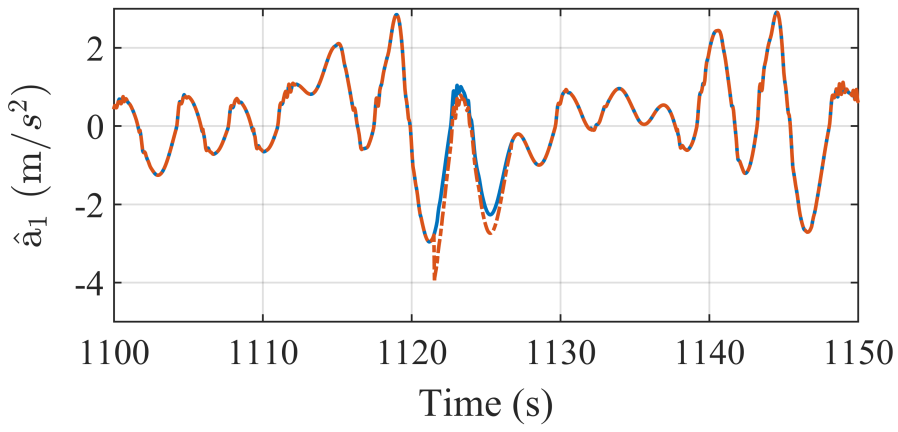


(b)

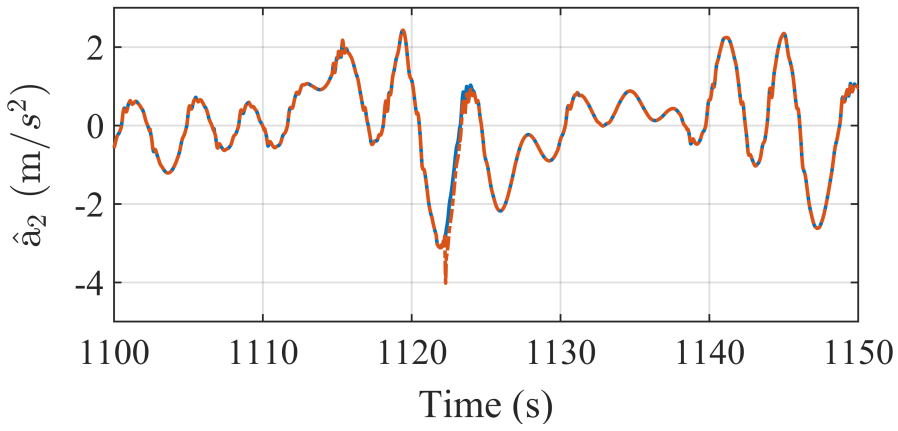


(c)

Figure 4.7: (a) Velocity profile of the platoon's first three vehicles following a reference vehicle. (b) Acceleration profile of the vehicles in the platoon. (c) Inter-vehicular distance error of the first three vehicles in the platoon.



(a)



(b)

Figure 4.8: Control signal generated by RG (red dashed line) and original acceleration set-point of the preceding vehicle in: (a) first vehicle and (b) second vehicle.

by RG and the original references for the two follower vehicles in the same drive cycle. It can be seen that in 1122 seconds, RG has modified the original reference for about five seconds to avoid constraint violations. In 1147 seconds, the distance error reaches -0.6m but RG does not modify the reference because it recognizes that the system remains inside the constraint with the original CACC reference.

4.3.3 Ecological improvement

The ecological performance improvement is done by controlling the leader to achieve smoother speed trajectories with reduced unnecessary accelerations. The designed NMPC controller minimizes the energy cost of the leader. Because in a string stable platoon, the follower vehicles try to follow the same trajectory as the leader with reduced oscillations, it is expected that the followers will have a better energy economy. Thus, a higher energy efficiency can be achieved for the whole platoon. Figure 4.9 shows the speed trajectory of a string stable platoon of vehicles following the FTP-75 driving cycle. As it can be seen, the tail vehicle has a much smoother speed trajectory compared to the leader which is because of the platoon's string stability. This smoother trajectory will result in less unnecessary accelerations and a better energy economy for the platoon.

Figure 4.10 and Table 4.1 show the total trip energy cost of each vehicle in the platoon. As expected, for three consecutive FTP-75 drive cycles, the lead vehicle has the highest energy consumption ($1.85), followed by the last two vehicles ($1.74). Therefore, the tail vehicle has a 6% lower energy cost compared to the lead vehicle that has already reduced its energy cost using an ecological ACC for car-following. In the case of the leader with PID controller, the tail vehicle has a 12% lower energy cost compared to the leader. The trend in the energy consumption of the string of vehicles suggests that, in the proposed platoon design, there is a limit in the improved energy cost caused by the reduced unnecessary acceleration and deceleration. The oscillations in the lead vehicle's velocity profile get attenuated in the string of vehicles and they vanish after a number of vehicles. After that, the other followers have a very close velocity profile, and therefore a similar energy consumption.

Figure 4.11 and Table 4.1 show the total trip energy cost of the platoon in three consecutive FTP-75 drive cycles. The platoon with the proposed NMPC leader controller has an energy cost of $10.64 which is about 6% lower than the platoon with PID leader, which has an energy cost of $11.32. This performance is mostly due to the reduced unnecessary accelerations and decelerations in the vehicle motion because of using a predictive

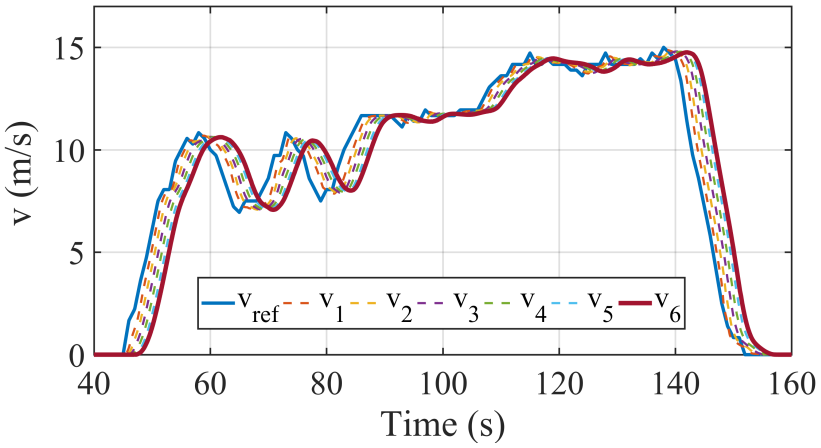


Figure 4.9: Speed trajectory of platoon vehicles following FTP-75 standard drive cycle.

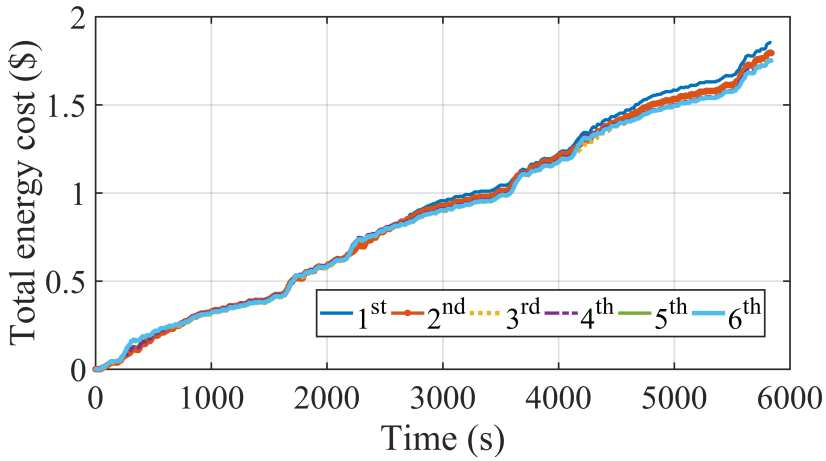


Figure 4.10: Total energy cost for each vehicle in the platoon following the FTP-75 driving cycle.

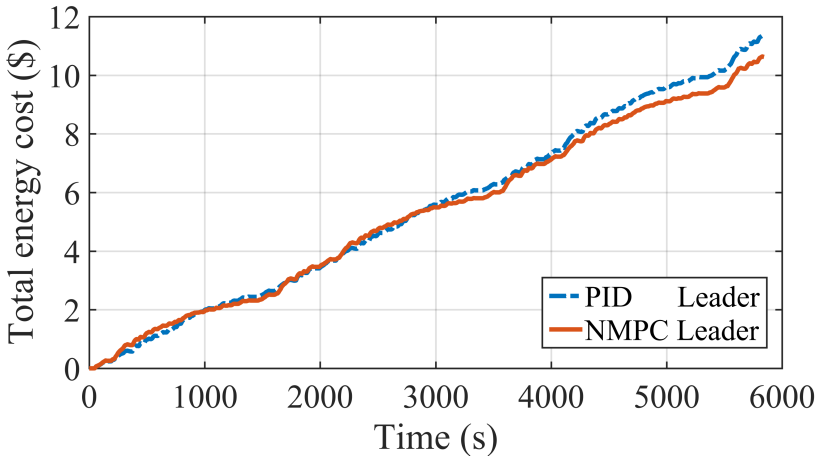


Figure 4.11: Total energy cost of the whole platoon following the FTP-75 driving cycle with PID and NMPC controlled leaders.

controller.

The presented Eco-ACC controller for the leader has resulted in 10% lower energy cost for the leader. It has also resulted in lower energy costs for the follower vehicles, compared to a regular PID controller. The tail vehicle in the NMPC lead platoon consumed about 3.9% less energy compared to the tail vehicle in the PID lead platoon. These results are a comparison between platoons with different leaders and do not show the improvements caused by platooning. Here, we have only considered the energy cost improvement due to minimizing unnecessary accelerations and decelerations in the vehicle platoon. A lower energy cost is also expected because of the platoon's reduced drag; however, our simulations did not consider the effect of the reduced drag due to its uncertain and complex behavior. Moreover, in passenger cars, lower drag in follower vehicles is only noticeable in inter-vehicular distances, at least, less than the length of the vehicle [114],[115]; however,

Table 4.1: Platoon's energy cost for different leader controllers

	Energy Cost ($)						
Leader's Controller	1^{st} Car	2^{nd} Car	3^{rd} Car	4^{th} Car	5^{th} Car	6^{th} Car	Sum
PID	2.06	1.94	1.86	1.83	1.83	1.81	11.32
NMPC	1.85	1.79	1.76	1.75	1.74	1.74	10.64

considering the safety requirements, network imperfections and string stability, such low distances are not currently feasible for practical platooning of passenger vehicles. In summary, the effect of the reduced drag on the indicated ecological performance results will not be significant.

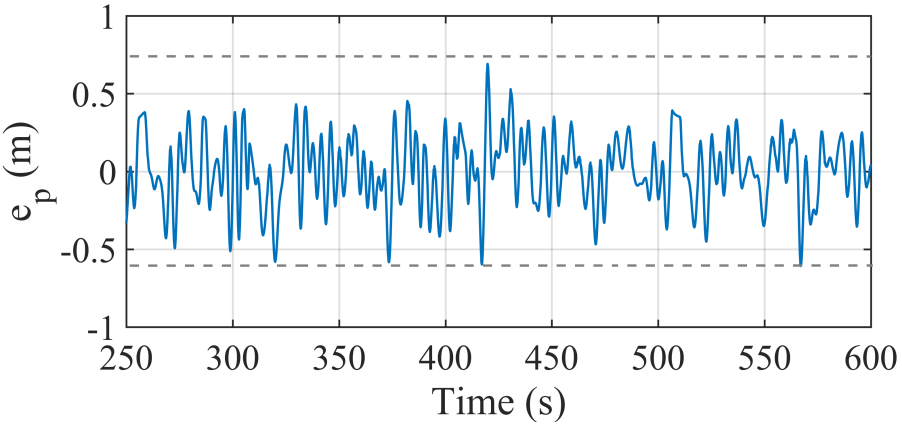
4.3.4 Real-time implementation

To further examine the performance of the proposed controller, HIL experiments have been carried out to ensure the potential and capability of the controller for real-time implementations in practice. Like before, dSPACE Micro-Autobox II control prototyping hardware is used for HIL tests. The HIL setup consists of three main components: a real-time machine (DS1006 processor board), a prototype ECU (MicroAutoBox II), and a human-machine interface. Like the previous chapter, all communication between the prototype ECU and the real-time machine is performed through a CAN bus.

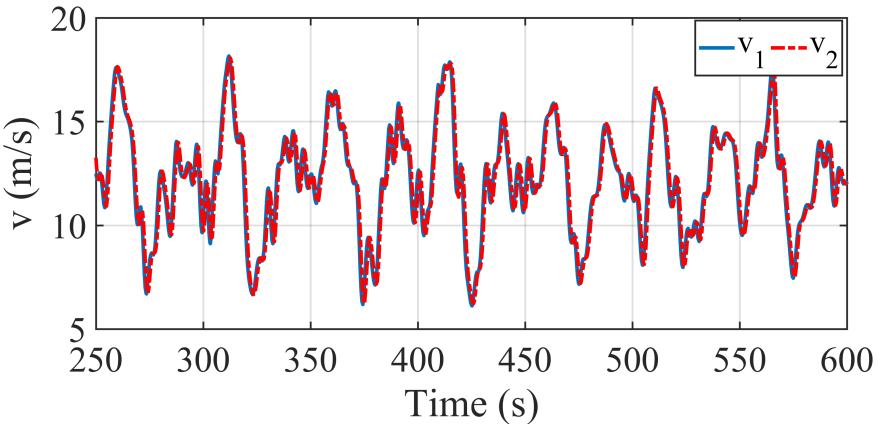
For the HIL tests, only two vehicles were considered, a follower that runs the proposed controller and a leader. The platooning controller with RG was uploaded to the test ECU to evaluate its computation time. This was done because, in HIL experiment, we are only interested in computational demand of one ECU and the RG algorithm for that specific car.

Figure 4.12 shows the velocity and position error during the HIL experiment and Figure 4.13 shows the turnaround time of the ECU running the proposed platooning controller. In regular situations when RG only checks the admissibility, the turnaround time of the controller is about 0.05ms. When RG detects a chance for violation of the constraints, it becomes active; therefore, the turnaround time of the controller increases for a limited time. Even when there is an increase in the computations, the turnaround time is below 1ms. To better illustrate the performance of the controller, only a part of the experiment has been shown in Figure 4.12.

These results show that the proposed controller can be executed on a real vehicle ECU and work in real-time with no concern about the delay caused by high computational demand of the optimization problem.



(a)



(b)

Figure 4.12: (a) Speed and (b) position error of the leader and the first follower vehicle during the HIL experiment

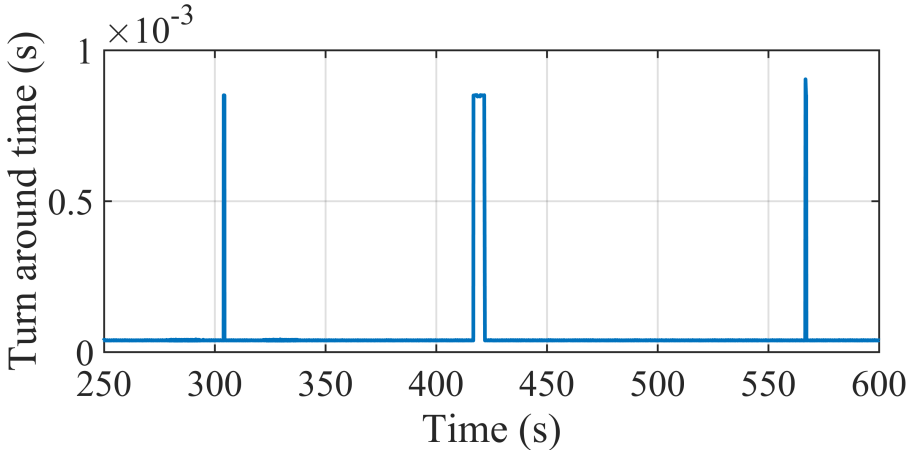


Figure 4.13: Turnaround time of the prototype ECU during the HIL experiment.

4.4 Summary

This chapter targeted an important issue in the platooning of vehicles: the simultaneous satisfaction of frequency-domain criteria, namely the string stability and time-domain criteria, that is, the constraint handling. A novel RG approach was proposed to enforce the constraints while preserving the string stability of the designed platoon.

In the first step, a control-oriented model was presented to be used in platooning and ecological control of the baseline PHEV, namely Toyota plug-in Prius. The linear platooning model was used in the RG design and the nonlinear PHEV model was used in the heart of the NMPC to control the lead vehicle. A high-fidelity model of the baseline PHEV was used to evaluate the controllers and validate the control-oriented model. The linear platooning controllers were designed to achieve a string stable platoon based on the induced norm and the platooning model proposed in the modeling section. Then, the RG-based controller was designed to enforce the defined constraints on the stable system. The RG sits behind the CACC controller and, by getting the host vehicle's states and the preceding vehicle's acceleration set-points from a wireless network, modifies the reference signal to keep the defined constraints. RG produces the same reference as its input to the system and only changes it when it recognizes a possibility for violation of the constraints. Therefore, based on the first step's design, the platoon remains string stable and can handle the platooning constraints.

To improve the platoon's energy economy, an ecological ACC was presented that en-

ables the lead vehicle to follow an unconnected preceding vehicle that is not part of the platoon. The follower vehicles consume less energy, as they follow the same trajectory as the leader, with reduced oscillations due to string stability. Therefore, as suggested in the existing literature, there is no need to control each vehicle separately and specifically for improving its energy economy. This control approach combines string stability; safety and performance constraints handling; and energy economy improvement and satisfies all of them at the same time.

A frequency-domain analysis showed that the proposed controller results in a string stable platoon based on the controller design. The result for constraint satisfaction demonstrated that RG is able to keep the system inside the defined constraints only by modifying the original reference to the CACC controller. As shown in Figure 4.8, the added RG did not change the general behavior of the controlled system and only intervened when there was a chance for constraint violation. Consequently, the system remained string stable and was able to handle the defined constraints.

The major contribution of this chapter is that the proposed control approach has separated the main platooning design for car-following and string stability from constraint handling. The results show that a platoon whose leader had the NMPC controller recorded an ecological improvement of about 6%, compared to the one with a PID controller. The follower vehicles had up to 10% lower energy costs compared to the leader, which is due to the string stable platooning. The HIL test with a dSPACE setup showed that, on average, the proposed controllers had low computation costs with increased turnaround time only when RG became active. However, the turnaround time was below 1ms at all cases which means that the proposed controller can be executed on a real vehicle ECU with any sampling time higher than 1ms.

In this chapter, a reference governor-based controller was developed that can achieve frequency-domain and time-domain requirements at the same time. The given string stable controller was developed based on homogeneity assumption. In the next chapter, we will present an MRAC based platooning controller that can achieve string stability for a heterogeneous vehicle platoon.

Chapter 5

String stable heterogeneous vehicle platooning

This chapter presents an adaptive control approach to the platooning of heterogeneous vehicles. This research is mainly concerned with the design of controllers for PHEVs. Therefore in the last chapter, platooning of homogeneous PHEV platoons was considered. However, in practice, a PHEV might have to perform platooning with the other types of vehicles. Therefore, to have a more practical controller, it is necessary to study platooning of PHEVs in interaction with heterogeneous platoons as well. This heterogeneity may be caused by different network topologies, different vehicle dynamical behaviors or the external disturbances. To take into account these differences, a controller should be developed that can adapt to changes to maintain string stability despite these differences. In order tho achieve this gaol, in this chapter, a direct model reference adaptive control (MRAC) approach to this problem is explained along with the necessary evaluations to show the performance of the presented control method. MRAC can enforce a desired behavior on the system by performing an online parameter estimation and using an adaptive control law. This way a string stable behavior can be enforced on the platoon despite dynamical differences and external disturbances.

In the rest of this chapter, first, a nonlinear dynamical model is presented that explains a heterogeneous platoon subjected to external disturbances. Next, a direct MRAC is designed that enforces a string stable behavior on the vehicle platoon despite different dynamical models of the platoon members and the external disturbances acting on the system. The proposed method estimates the controller coefficients online to adapt to

disturbances such as wind, changing road grade and also to different vehicle dynamic behaviors. Finally, the proposed controller is evaluated with a high-fidelity heterogeneous platoon model with injected communications and radar delay. The controller evaluations show that the proposed controller is able to maintain string stability in the presence of delay, disturbances, and heterogeneous dynamical models. Moreover, HIL experiments are performed to show the performance of the proposed controller on a prototype vehicle ECU in real-time application. These experiments show that the proposed controller can control the vehicle platoon with the limited computational power available on a vehicle ECU and in a real-time fashion.

5.1 Modeling

This section explains the models that have been used for the design and evaluation of the proposed controller. First, a high-fidelity model is explained that has been used for constructing a high-fidelity heterogeneous platoon model. This model contains high-fidelity models of each component in each vehicle that combined together make an accurate complex model, which is appropriate for evaluation purposes. For the control design, however, this model is too complex. Therefore, a simpler nonlinear model is presented that is used for parameter estimation and adaptive control of the platoon. This model has less complexity compared to the high-fidelity model but it is descriptive enough to capture the general behavior of the system.

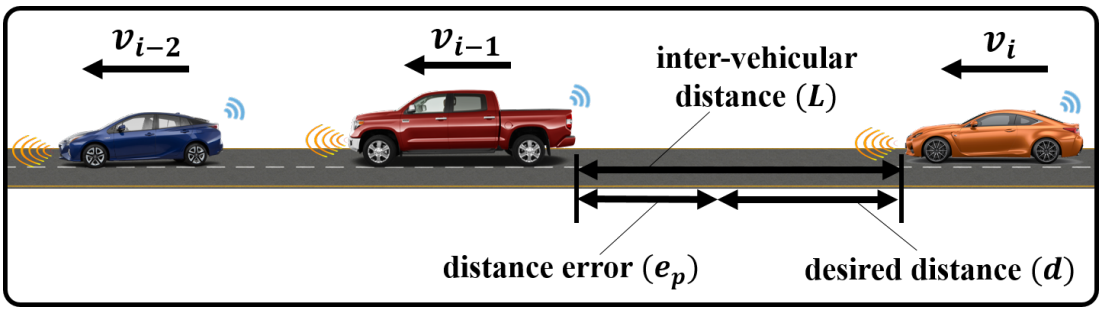


Figure 5.1: Car-following parameters of a heterogeneous vehicle platoon

5.1.1 High-fidelity model

For the evaluation of the proposed controller, accurate models of the vehicles are necessary. To construct such a high-fidelity model, dynamic models of the baseline vehicles are developed in Autonomie. As explained before, Autonomie is a MATLAB-based software package with a library for different vehicle models and components that can be selected to generate a high-fidelity model of the whole vehicle. Autonomie allows the selection of two or four-wheel vehicles with different types of powertrains and also the modification of its components and characteristics to simulate each baseline vehicle. The developed models of different vehicles are combined in a simulation model to construct a high-fidelity vehicle platoon model in Simulink. The designed controller will control the longitudinal motion of each car by receiving inter-vehicular distances, velocity and also control input signals from the preceding vehicles and generating control inputs that can maintain the desired distances and string stability. Road grade profile, changing wind speed and sensor delay for radar and V2V communications are also injected into the model to simulate realistic scenarios.

5.1.2 Platoon model

A vehicle platoon can be formed by a number of interconnected vehicles that receive data about their preceding vehicles through a wireless V2V network and radar. Figure 5.1 shows the effective car-following parameters of a three-vehicle heterogeneous platoon along with the assigned vehicle indexing. The main goal of car-following is to maintain a safe inter-vehicular distance which may be achieved with an error equal to e_p. Different types of policies can be found in the literature for the desired inter-vehicular distance, here a constant time headway rule is used: $d_i = d_0 + h_i v_i$, where d_0 represents the standstill desired vehicle gap and h_i shows the constant headway time. Considering this gap policy,

the nonlinear error dynamics of car-following can be written as follows:

$$
\begin{aligned}
&\dot{x}_i = Ax_i + Bu_i(t-\tau_{a_i}) + B_p a_{i-1} + B_g F_{r_i} \\
&x_i = \begin{bmatrix} e_{p_i} \\ e_{v_i} \\ T_{w_i} \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 & -\frac{h_i}{m_i r_{w_i}} \\ 0 & 0 & -\frac{1}{m_i r_{w_i}} \\ 0 & 0 & -\frac{1}{\eta_i} \end{bmatrix}, \\
&B = \begin{bmatrix} 0 \\ 0 \\ \frac{K_{a_i}}{\eta_i} \end{bmatrix}, \quad B_p = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad B_g = \begin{bmatrix} -h_i \\ -1 \\ 0 \end{bmatrix}, \\
&F_{r_i} = -\frac{1}{2}\rho_a A_{c_i} C_{d_i}(v_i + v_{w_i})^2 - (\mu_{r_i}) mg\cos(\phi_{r_i}) - mg\sin(\phi_{r_i}),
\end{aligned}
\tag{5.1}
$$

where p_h, v_h, and a_h are the position, velocity, and acceleration of the host vehicle and $e_p = p_p - p_h - hv_h$, $e_v = v_p - v_h$, and also p_p, v_p and a_p are the position, velocity, and acceleration of the preceding vehicle, F_r represents the sum of all the resistance forces, ρ_a is the air density, A_c is the frontal area of the car, C_d is the drag coefficient, μ_{r_0} is the rolling resistance coefficient, m is the vehicle's mass, v_w is the headwind speed and ϕ_r is the road grade. T_w is the wheel torque, r_w is the wheel radius, and i shows the vehicle index. Equation (5.1) represents a nonlinear vehicle platoon model that is used for control design. This nonlinear model includes drag forces by vehicle motion and wind, rolling resistance force, and gravity force due to the grade changes.

Each vehicle in this platoon has a different set of parameters that cause a dynamic heterogeneity in the platoon. The designed controller must be able to work with each different vehicle and maintain string stability despite the dynamic differences. The list of the vehicle parameters used here is presented in Table 5.1. There are four different types of vehicles in this list including battery electric vehicle (BEV), PHEV, diesel truck and conventional vehicle with an internal combustion engine.

5.1.3 Communication network topology

In this research, the communications between the platoon's vehicles is considered unidirectional from preceding vehicles to the followers. Different communication topologies are considered here which are commonly used in the literature [116]. These topologies include: communication only between each immediate predecessor and follower (PF) pair, commu-

Table 5.1: Model parameters of the heterogeneous platoon's vehicles

vehicle index	1	2	3	4	5	6
vehicle type	BEV	PHEV	Truck	Conv.	PHEV	Conv.
$m\ [kg]$	2150	1925	10000	1580	1720	1850
$r_w\ [m]$	0.301	0.301	0.478	0.32	0.301	0.35
$\tau_a\ [s]$	0.1	0.05-0.2	0.4	0.3	0.05-0.2	0.3
μ_r	0.008	0.010	0.014	0.0012	0.012	0.008
$A_c\ [m^2]$	2.20	2.25	8.918	2.18	2.25	1.8
C_w	0.35	0.3	0.7	0.32	0.3	0.32

nication between every two predecessors and follower (TPF), communication between the leader, immediate predecessor and follower (LPF) and communication between all predecessor vehicles, leader, and follower (APLF). Figure 5.2 shows the different communication topologies used in this chapter. In Figure 5.2 (d), due to the complexity of the whole network, only communication to the last vehicle is shown to have a clearer picture.

In a vehicle platoon, the lead vehicle chooses the set-point of the whole platoon. The follower vehicles with PF topology receive the set-point changes only from their immediate preceding; therefore, it takes at least a time equal to the sum of all the delays in each communication link from a follower to the leader to receive the changes. A direct communication from the leader to each follower, however, can transmit the platoon set-point changes with a much smaller delay and at approximately same time to each follower, which in turn will improve the performance of the platoon. But the problem is that the range of V2V communication is limited and therefore there is a maximum length for a platoon that requires direct communication with the leader. Another problem is that in a heterogeneous platoon, transmitted data from the leader might not match the safety requirement of a follower. Especially, different amount of propulsion or brake force can cause safety constraints violation for followers.

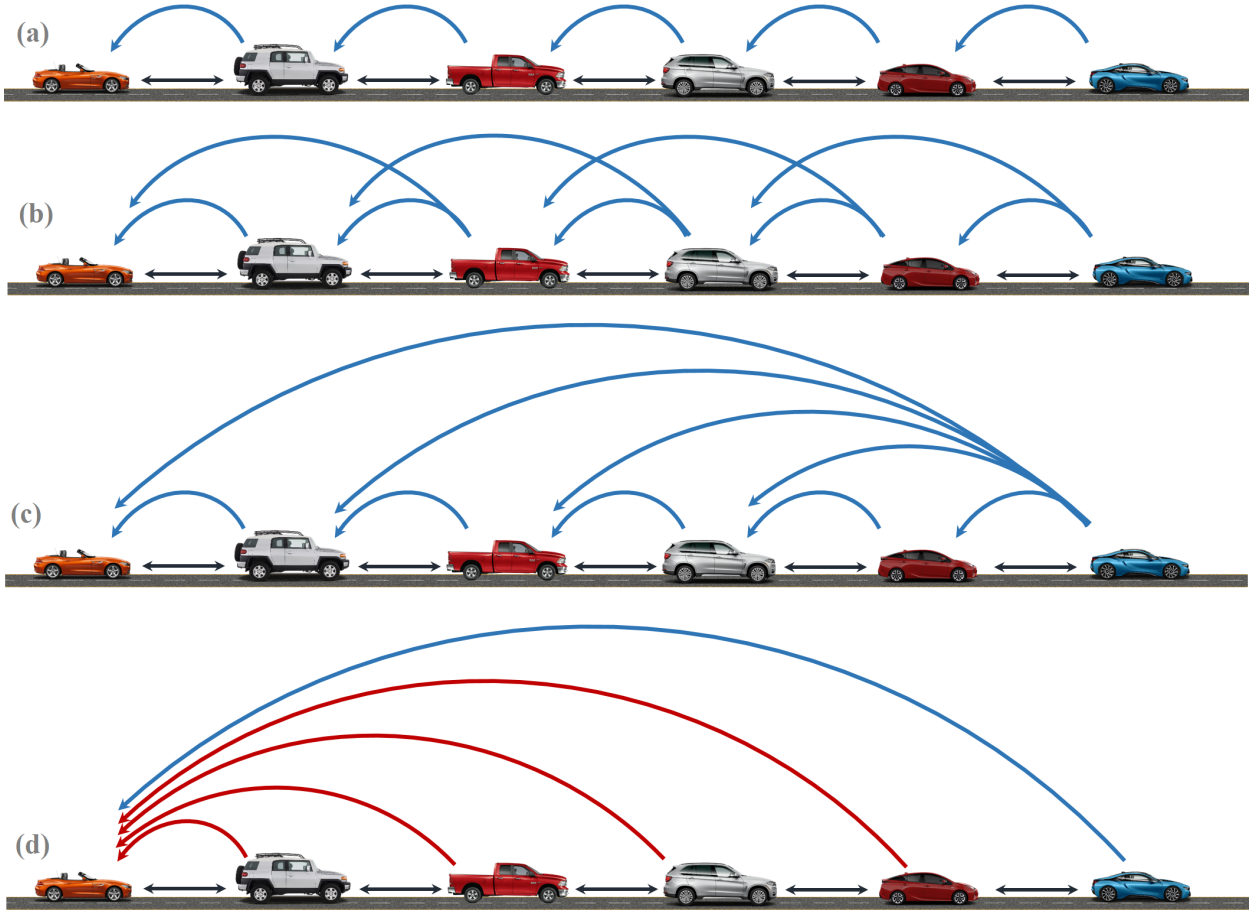


Figure 5.2: Different communication network topologies (a) predecessor-follower (PF) (b) two-predecessors-follower (TPF) (c) predecessor-leader-follower (PLF) (d) all-predecessors-leader-follower (APLF)

5.2 Control design

This section presents the controller design procedure for platooning of heterogeneous platoons. First, string stability is explained and the desired reference model for a string stable platoon is specified. Then, the parametric model and the adaptation law for the design of the adaptive controller is presented. Finally, direct MRAC is explained for adaptive string stable platooning. In this chapter, different wireless communications networks have been

used from predecessor vehicles to the follower based on the control architecture in Figure 5.3. In this architecture, the inter-vehicular velocity and distance are available from an onboard radar, and also acceleration control set-points of the preceding vehicles are given by the wireless V2V communications. The MRAC receives these data and generates appropriate control inputs to control the vehicle as desired.

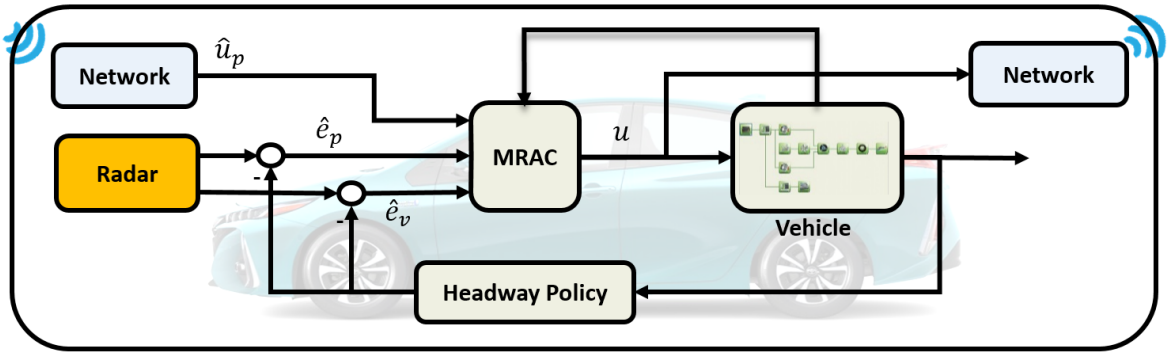


Figure 5.3: MRAC-based platooning control architecture

5.2.1 String stability

We defined string stability in the previous chapter. String stability is an important factor in vehicle platooning, which is essential for ensuring the safety of the platooning system. The same definition applies to heterogeneous platoons. A string stable heterogeneous platoon must ensure that disturbance signals will not detonate as they go along the vehicle string, so that the tale vehicle has attenuated disturbances compared to the lead vehicle. As mentioned before, it is possible to define string stability based on the attenuation of different types of signals. Here, acceleration signals are assumed in each predecessor-follower vehicle pair. However, any other definition of string stability can work the same for this control design and synthesis. Looking back at (4.6) , string stability of the system can be defined based on transfer function between the acceleration of a follower and its predecessor. This criterion requires that the transfer function between the accelerations of each follower-predecessor must be less than or, at most, equal to one at all frequencies.

Based on Figure 5.3, available data to the MRAC controller are preceding vehicles' acceleration set-points, inter-vehicular distance, and inter-vehicular velocity. Based on these signals, the reference model must define the desired acceleration, considering the

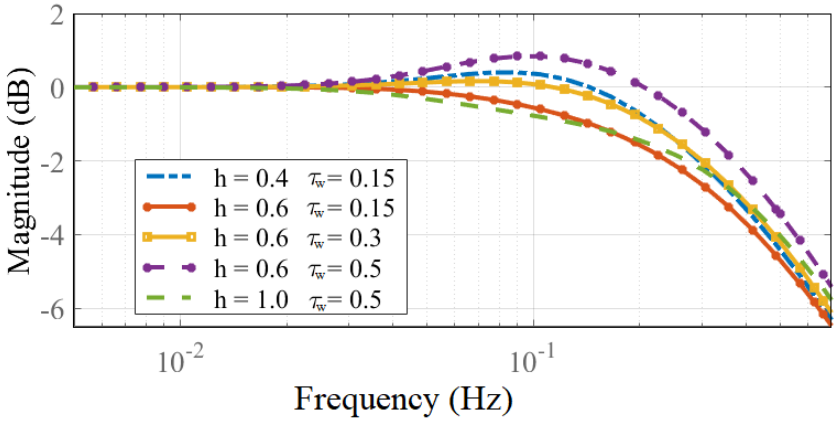


Figure 5.4: Frequency response of the transfer function between accelerations of each predecessor-follower pair for different time headway ($h[s]$) and wireless delay ($\tau_w[s]$)

existing delay in each of them. Therefore, we define the following reference model:

$$a_{h_m} = \frac{T_m}{s + T_m}(a_p\, e^{-\tau_w s} + k_p\, e_p\, e^{-\tau_R s} + k_v\, e_v\, e^{-\tau_R s}), \tag{5.2}$$

where a_{h_m} is the reference desired acceleration, τ_w is the wireless communication's delay, τ_R is the radar's delay, and T_m, k_p and k_v are design parameters. To find the string stable control parameters, we assume that the delay in the radar is equal to $\tau_R = 400\,[ms]$ [117] and then find stabilizing control parameters and constant time headways for different amounts of delay in wireless communications. Figure 5.4 shows the string stability signal for different wireless delays and constant time headways, considering the desired reference system. Increasing the wireless delay will result in a bigger magnitude for the string stability signal, which means that the upstream vehicles will be affected by disturbances and platoon set-point changes with a higher amplitude compared to their predecessors. Increasing the constant time headway value will improve string stability and there is a stabilizing minimum time headway for each amount of wireless delay. The design and synthesis of the controller is based on the first order model, presented in the Section 5.1.2, between the desired and actual wheel torque.

5.2.2 Parametric model

To develop a parametric model, we assume the following vehicle model:

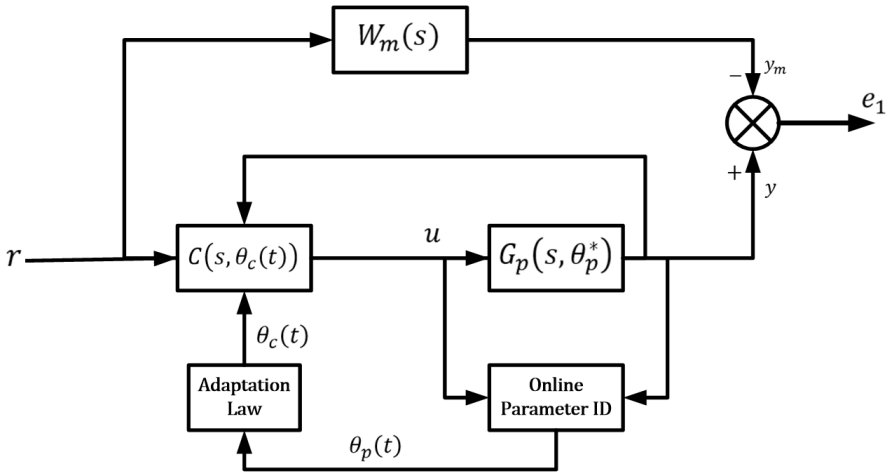


Figure 5.5: Schematic of a model reference adaptive controller

$$\begin{aligned} a_i =& \frac{R_g}{r_w m_i} \frac{E_i}{1+\eta_i s} T_{com_i} - \frac{\rho_a A_{c_i} C_{d_i}}{2m_i} v_i^2 - \frac{\rho_a A_{c_i} C_{d_i}}{m_i} v_{w_i} v_i \\ & - \frac{1}{2} \frac{\rho_a A_{c_i} C_{d_i}}{m_i} v_{w_i}^2 - g\mu_{r_i} cos(\phi_{r_i}) - g sin(\phi_{r_i}) \end{aligned} \tag{5.3}$$

where E is the vehicle's powertrain efficiency, R_g is the gear ratio between the engine and wheels, r_w is the wheel radius and the other parameters are as defined before. Assuming that only the desired torque T_{com} and speed of the vehicle are measurable, the unknown parameters can be lumped to find:

$$sv_i = \theta_1 T_{com_i} - \theta_2 s v_i^2 - \theta_3 v_i^2 - \theta_4 s v_i - \theta_5 v_i - \theta_6 s^2 v_i - \theta_7. \tag{5.4}$$

5.2.3 Adaptive control by direct MRAC

Figure 5.5 shows the general configuration of an MRAC controller. $W_m(s)$ is the transfer function of the desired behavior. The controller's main goal is to minimize e_1, the error between the output of the system and the desired model, by adjusting the controller's coefficients. In indirect MRAC, an online parameter estimator estimates the plant's parameters online and then the controller uses an adaptation law to find the appropriate

control coefficients. If a parameter in the plant changes during the operation of the system, the adaptive controller will be adjusted to account for that change and maintain the desired behavior. To further simplify this controller, it is possible to combine the parameter estimation, adaptive law and the controller in a single control block and make a direct MRAC. In the direct MRAC, instead of the system's parameters, the controller directly estimates the control coefficients that minimize the error between the desired and actual output of the system.

In this system, based on Figure 5.3, the main goal of MRAC is to enforce the desired reference behavior on to the system. The controller gets the preceding vehicles acceleration from the network, and also, the inter-vehicular distance and velocity from the radar. The MRAC must enforce the dynamic behavior in Equation (5.2) to ensure the string stability. The parametric model (5.4) has too many unknown parameters, which may cause convergence problems in the estimation of parameters. Therefore, we use a simpler model by feed-forwarding the known parts of disturbances to reduce their effect on the dynamic model and then consider them as a disturbance term shown as d in the following reduced parametric model:

$$\dot{a}_i = +\frac{\theta_1}{\theta_6}u_i - \frac{\theta_4 + 1}{\theta_6}a_i - \frac{\theta_5}{\theta_6}v_i - d, \tag{5.5}$$

where d is the disturbance that has replaced the previously compensated parts. The following control structure is used to achieve the desired string stable behavior:

$$u_i = k_1 a_i + k_2 a_{i-1} + k_3 e_{v_i} + k_4 e_{p_i} + k_5. \tag{5.6}$$

Using this control structure and the desired control behavior, the actual control coefficients to generate the desired model can be calculated based on the actual parameter values.

$$
\begin{aligned}
k_1^* &= -\frac{\theta_6^*}{\theta_1^*} + \frac{(\theta_4^* + 1)}{T_m \theta_1^*}, \\
k_2^* &= \frac{(\theta_6^*)}{T_m \theta_1^*}, \\
k_3^* &= \frac{(\theta_6^*)}{T_m \theta_1^*} k_v, \\
k_4^* &= \frac{(\theta_6^*)}{T_m \theta_1^*} k_p, \\
k_5^* &= \frac{\theta_5^* v_i}{T_m \theta_1^*} - \frac{\theta_6^* d}{T_m \theta_1^*},
\end{aligned}
\tag{5.7}
$$

where θ^* shows the actual value of parameters. Therefore, the error can be described by considering $e_a = a_i - a_{i_m}$ as:

$$
e_a = \frac{b}{s + T_m}(u_i - k_1^* a_i - k_2^* a_{i-1} - k_3^* e_{v_i} - k_4^* e_{p_i} - k_5^*). \tag{5.8}
$$

If $\widetilde{k} = k - k^*$ then:

$$
e_a = \frac{b}{s + T_m}(\widetilde{k}_1 a_i + \widetilde{k}_2 a_{i-1} + \widetilde{k}_3 e_{v_i} + \widetilde{k}_4 e_{p_i} + \widetilde{k}_5). \tag{5.9}
$$

This model is used to estimate adaptive control coefficients to achieve the desired string stable behavior by minimizing e_a. The main goal of MRAC is to find control coefficients that minimize the error between the desired and actual accelerations and therefore enforce the desired behavior on the system despite changes in the environment and different dynamical models. In case of a change in the system, an error between the desired and actual system will be observed through e_a. In this case, the MRAC adjusts the estimated k_i values to minimize this error.

5.2.4 Adaptive law

To estimate the control coefficients, here, least square parameter estimation is used. In this research, a least square parameter adaption method with a forgetting factor similar to [88] is used with the same notation. This algorithm updates the covariance and estimated

parameters online and when the vehicle is running. To prevent wrong estimations, it is necessary to limit the estimated parameters. Therefore, parameter projection is used to put constraints on estimations. Moreover, to avoid the covariance matrix from becoming very large, it is also necessary to put a constraint on its maximum value. Assuming the desired constraint on the control coefficients are defined by: $S = \{k \in R^n | g(k) \leq 0\}$, where g is a smooth function and R_0 as an upper bound for P, the projection can be defined as follows:

$$\begin{aligned}
\dot{k} &= \begin{cases} P\epsilon\phi & \text{if } k \in S^o \text{ or} \\ & k \in \delta(S) \;\&\; (P\epsilon\phi)^T \nabla g \leq 0 \\ P\epsilon\phi - P\frac{\nabla g \nabla g^T}{\nabla g^T P \nabla g} P\epsilon\phi & otherwise \end{cases} \\
\dot{P} &= \begin{cases} \beta P - P\frac{\phi\phi^T}{m_s^2} P & \text{if } \|P\| \leq R_0 \;\&\{k \in S^o \text{ or} \\ & k \in \delta(S) \;\&\; (P\epsilon\phi)^T \nabla g \leq 0\} \\ 0 & otherwise \end{cases} \\
\epsilon(t) &= \frac{e_a}{m_s^2(t)}.
\end{aligned} \tag{5.10}$$

where $\hat{k}$ is the estimated controller coefficients vector, $\hat{k}_0$ is the initial estimated coefficient vector, ϕ is the vector of measured input signals, β is a forgetting factor, P is the covariance matrix and m_s^2 is a normalizing term that is chosen as: $m_s^2 = 1 + \alpha\phi^T\phi, \quad \alpha \geq 0$.

5.3 Control evaluation

This section presents the controller evaluation results. A platoon of six vehicles is constructed in Simulink by combining the high-fidelity models of baseline vehicles to perform control evaluations. Each vehicle has its own separate model with different response time, mass, drag coefficient and efficiencies as presented in Table 5.1. The changing headwind and road grade profiles (acquired from Google Earth) used in the simulations are shown in Figure 5.6. The road grade and wind disturbances start at time $t = 500s$ in the simulations and their value is equal to zero at the beginning of each evaluation test. Initial control coefficients are set to zero for all of the vehicles and the control algorithm must adapt itself to each vehicle and also the changing environment to enforce the desired behavior. This

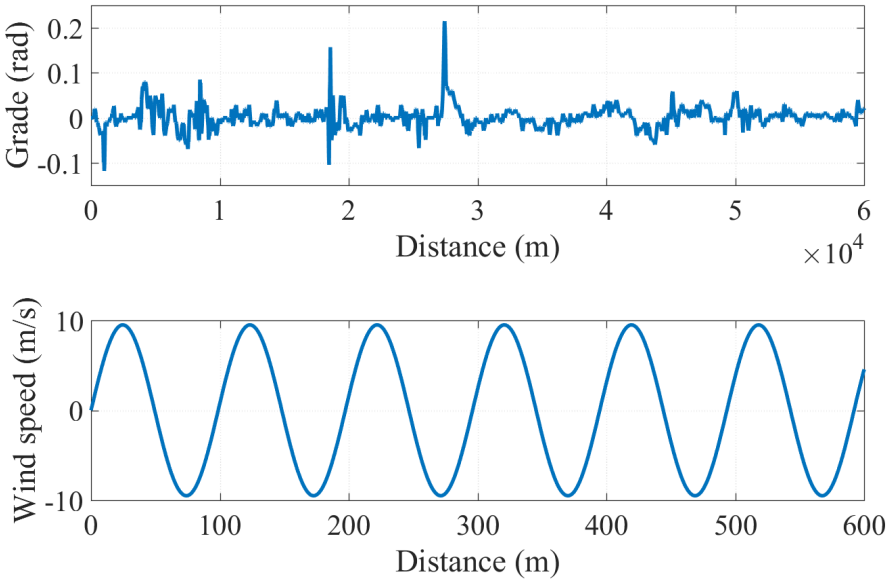


Figure 5.6: (a) road grade (b) headwind speed profile

adaptation will be achieved by estimating the control parameters based on the difference between the desired and actual accelerations of the vehicle during the simulation.

5.3.1 Control estimation

Figure 5.7 shows the estimated control coefficients of the first and last vehicles in the platoon. It is worth noting that the estimated control coefficients may not converge to their desired value (k^*), which, for the adaptive control case, is acceptable as long as the estimated output matches the desired value [88]. Figure 5.8 compares the desired and real acceleration achieved by the adaptive controller for the first and last vehicles in the platoon. The controlled acceleration follows the reference behavior closely, which shows that the adaptive controller is successfully adapting to the changing vehicle model and environment. Therefore, based on the control design and analysis done in the last section, the system must remain string stable regardless of the heterogeneity in the platoon and changing environment.

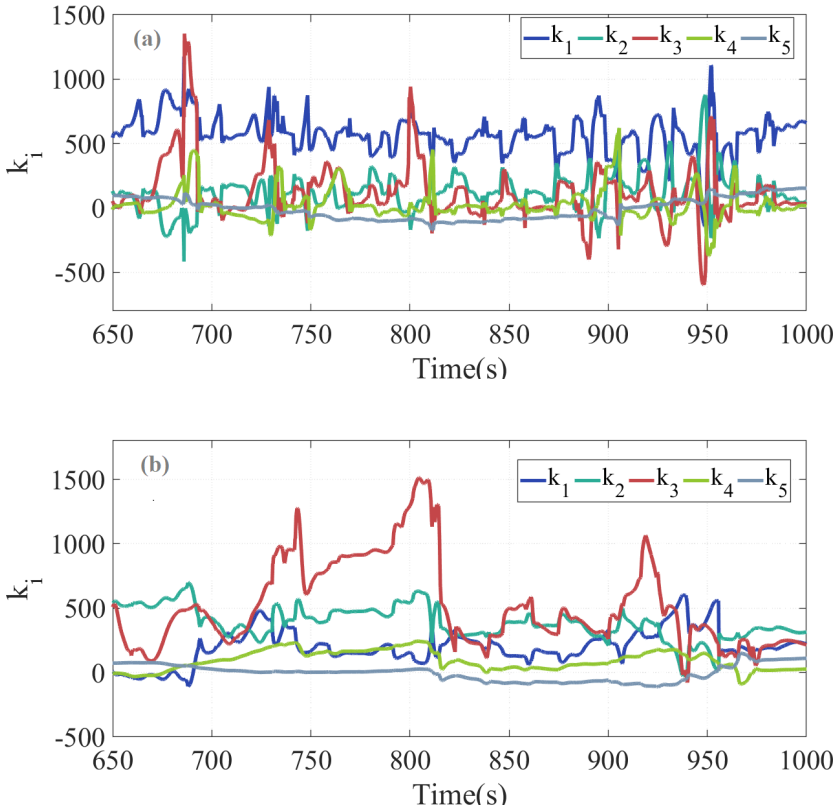


Figure 5.7: Estimated control coefficients of direct MRAC for (a) first vehicle and (b) last vehicle in the platoon

5.3.2 String stability under different network topologies

Figure 5.9 illustrates the speed trajectory of each vehicle in the heterogeneous vehicle platoon. This speed trajectory is given as the reference to the leader and other vehicles follow the leader using the adaptive control approach presented in this chapter. As it can be seen in Figure 5.9, the follower vehicles have less oscillations in their speed trajectory profile compared to their preceding vehicles, which means that disturbances are getting attenuated as they go upstream in the string of vehicles. Therefore, by using the MRAC to enforce the desired behavior on different vehicles in the platoon, a string stable platoon has been achieved. This performance evaluation has been done on a high-fidelity vehicle model with complex components models and in the presence of external disturbances. The

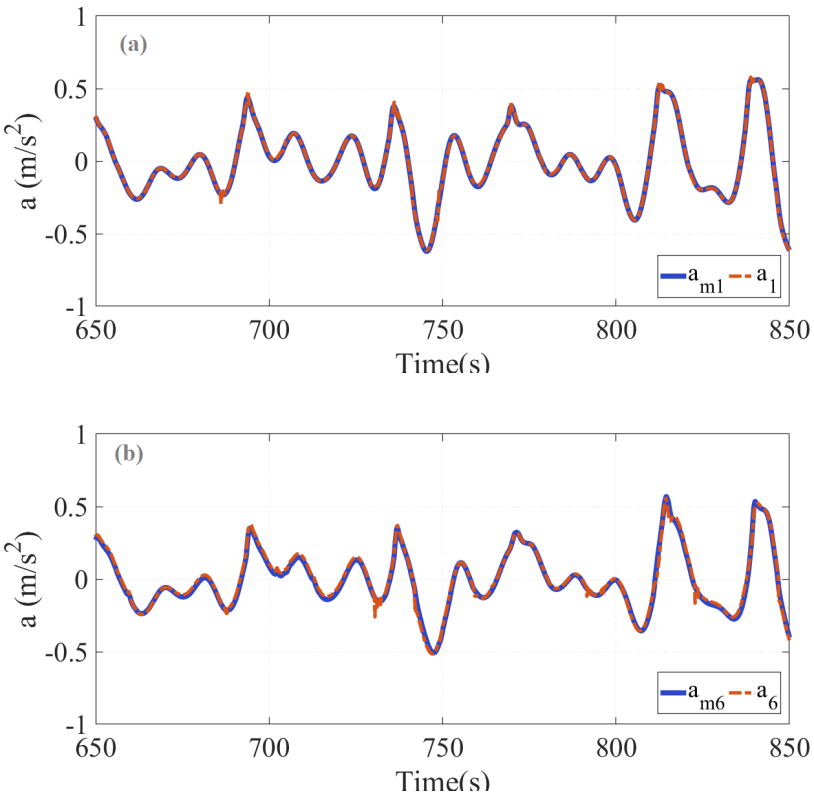


Figure 5.8: Desired acceleration and actual achieved acceleration in (a) first and (b) last vehicle in the platoon

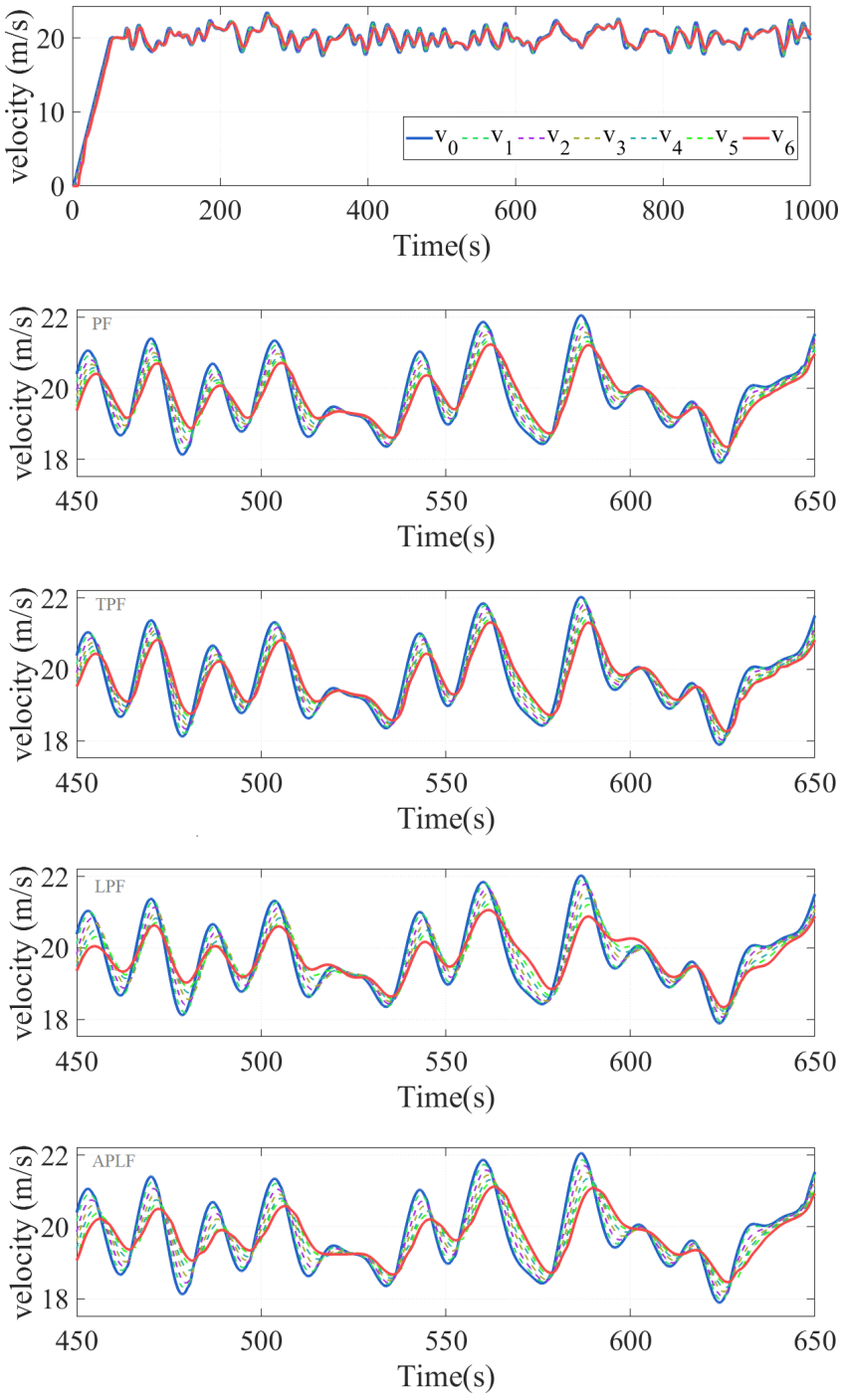


Figure 5.9: Velocity profile of the heterogeneous vehicle platoon following a leader

presented method has been able to capture the changes in the alternating environment and also in the vehicle model and enforce the desired behavior despite these differences.

Different network topologies have also been considered here. Figure 5.9 shows the results of the simulation with different network topologies. It can be seen that in all cases the presented method has been successful in preserving the string stable behavior of the platoon. Platoons with leader communication have better attenuated oscillations in the platoon set-point, which is, as explained before, due to the faster transmission of set-point changes to the tail vehicles. It is worth noting that this research is not concerned with finding the best network topology for heterogeneous vehicle platoons and evaluations with different topologies have only been performed to show the performance of the proposed controller.

5.3.3 Real-time implementation

To evaluate the performance of the proposed method for real-time applications, in this chapter, we have performed HIL experiments using the dSPACE HIL setup. As shown in Figure 5.10 and also as explained before, this setup includes a prototype ECU that runs the devised controller, a real-time simulator machine that runs the complex high-fidelity model in real-time, a personal computer that acts as a human-machine interface and facilitates the programming of subsystems. In our setup, MicroAutoboxII serves as the prototype ECU and all communications between MicroAutoboxII and DS1006 processor board are done through a CAN bus.

Figure 5.11 shows the turnaround time of the MRAC platooning controller executed on the MicroAutoBox II prototype ECU while controlling the high-fidelity vehicle model. Only the controller of one vehicle in the platoon was uploaded to the prototype ECU for the HIL test. The other vehicles are part of the simulation model that was uploaded to the real-time machine. The turnaround time of the controller is below $50\mu s$ at all times. Therefore, this controller has very little computational demand and easily can be executed on a vehicle ECU. Figure 5.12 shows the acceleration profile of the platoon vehicles during this experiment, which shows that the platoon is string stable with lower oscillation in the follower vehicles compared to the leader.

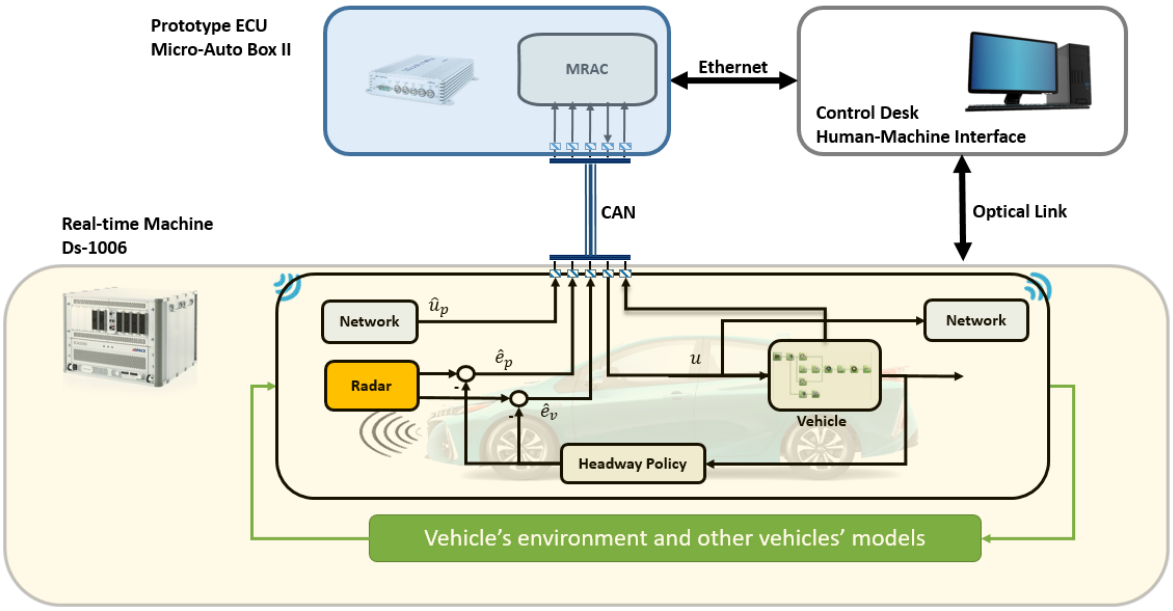


Figure 5.10: HIL experiment of the heterogeneous platooning controller

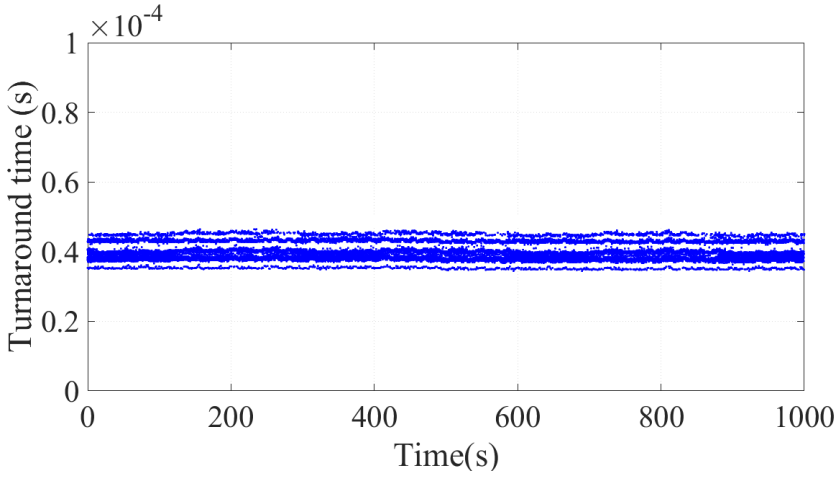


Figure 5.11: Controller turnaround time from HIL experiment

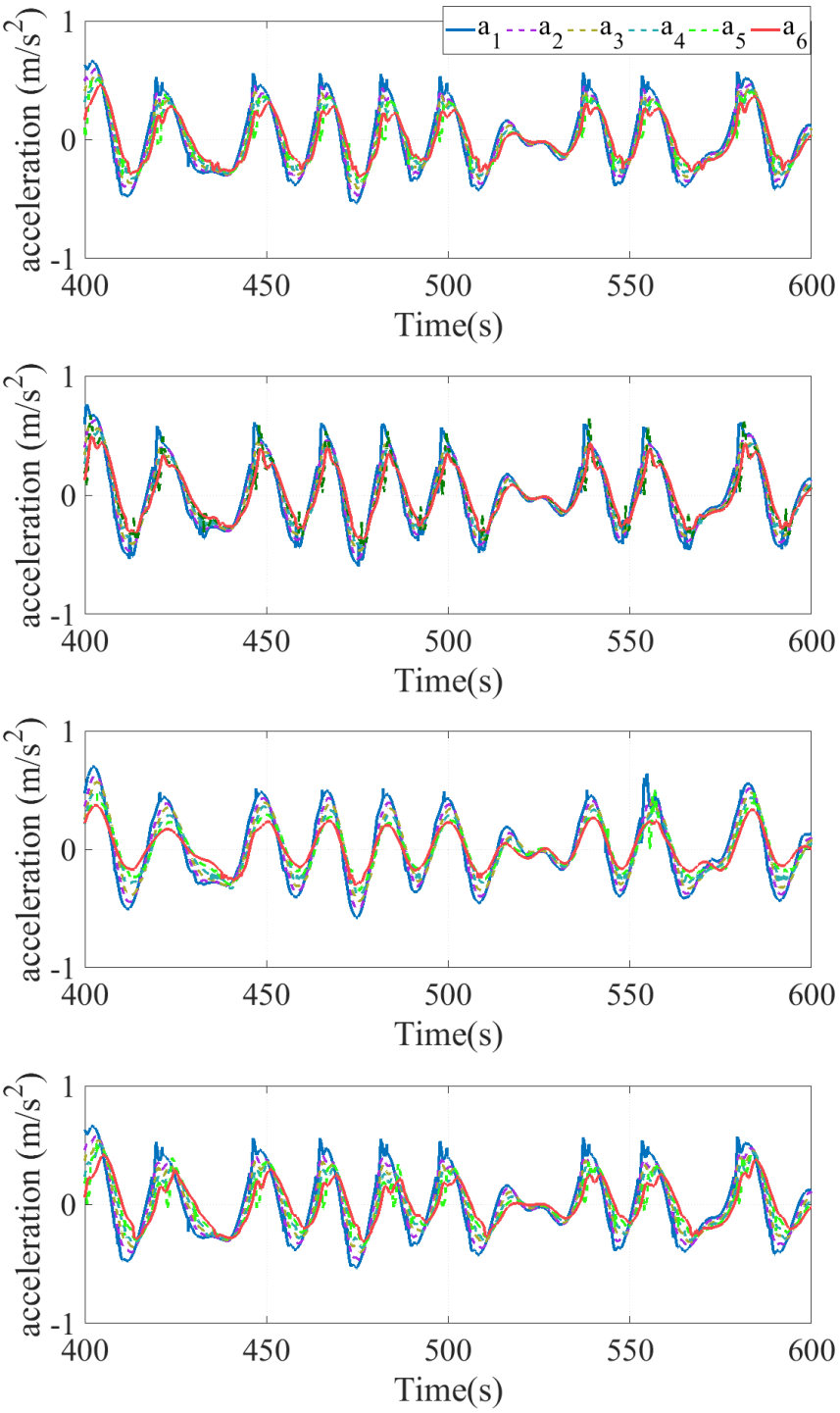


Figure 5.12: Acceleration profile of the heterogeneous vehicle platoon following a leader during the HIL experiment.

5.4 Summary

In this chapter, we proposed an MRAC approach to heterogeneous platooning. By enforcing the desired reference behavior on non-identical vehicles, the MRAC ensured a string stable behavior in the presence of heterogeneity, delay and altering environment. We began with modeling the vehicle platoon and presented a high-fidelity model for evaluation purposes and a simpler nonlinear control-oriented model used for the design of the controller. Then, string stability was defined based on the attenuation of acceleration signals in the vehicle platoon. A reference acceleration model was presented and based on it, the string stability was analyzed for different amounts of maximum network delay and headway time values. Then, the MRAC design was explained that enforces the desired behavior on the system.

The evaluation performed on a high-fidelity vehicle model with injected delay and disturbances showed that the proposed controller can achieve the string stability for a platoon of non-identical vehicles in a changing environment. Furthermore, the HIL experiment showed that the proposed controller can be executed in real-time on a prototype ECU with a small turn-around time.

Chapter 6

Conclusions

6.1 Thesis summary

In this PhD thesis, robust and adaptive controllers for real-time autonomous cruise control of connected plug-in hybrid electric vehicles were proposed, which ensure safety of the uncertain system while enhancing trip energy economy.

The second chapter was devoted to a review of the existing works in the literature on control of connected vehicles specifically in the area of Eco-ACC, Eco-CACC, heterogeneous platooning and HIL experiments for automotive applications. This chapter addressed the successful attempts in design and implementation of autonomous cruise controllers for car-following and platooning and concluded the necessary research for practical application of these systems.

In the third chapter, an adaptive and robust model-based approach was proposed to the control design of Eco-ACC. NMPC's application to ACC has already been discussed, in previous research works. Kamal et. al. [21, 118] analyzed advanced driver assistance systems and showed that using a predictive controller can improve the efficiency of ACC systems significantly. They used a multi-objective NMPC to minimize distance error and fuel consumption by utilizing future driving information. They also defined a cost function to minimize traffic jams. Employing NMPC enabled them to achieve different objectives by defining a multi-objective cost function while handling the defined safety, comfort and actuation constraints. Vajedi and Azad [11] took a similar approach to design an ECO-ACC for PHEVs based on NMPC method. They defined a cost function to minimize distance

error and trip energy cost, which interacted with the energy management system though receiving power ratio parameter. They evaluated their method on a high-fidelity vehicle model and showed a significantly enhanced trip energy cost. These studies have shown significant results. However, these results were based on assuming accurate, constant vehicle models, perfect data with no delay or imperfection and also a known environment. Also, their evaluation steps had been done on perfect evaluation models with no uncertainty. In practice, however, these methods cannot be very useful as existed uncertainties in the system would deteriorate their performance and may result in unsafe driving. To maintain a safe and efficient performance in the presence of uncertainties, in this research, we have developed the AT-NMPC method for design of Eco-ACC systems. In fact, AT-NMPC method and similar approaches had never been used before in the context of Eco-ACC. In this method, we translated the uncertainties in the system into an additive disturbance term and then generated disturbance invariant sets based on this disturbance. Based on the generated sets, we defined the NMPC problem with tightened constraints on nominal states to ensure boundedness of the real states of the system. This way the ACC system was robust against uncertainties and external disturbances. Moreover, to ensure that the optimal point of the defined cost function matches the actual optimal point of the system, a parameter adaption method was used to adapt to the changes in the system. This way the ACC system was robust against uncertainties while being adapted to them in order to maintain driving performance.

A major problem with NMPC is its high computational demand, which makes it very hard to be implemented in real-time. Therefore, to be able to implement NMPC in practice, control engineers have to decrease the complexity of the problem by simplifying their models and also use more expensive hardwares with higher computational power. Although oversimplified models would reduce the computational burden of NMPC, the unmodeled dynamics may lead to infeasibility, which in the case of close car-following could mean unsafe driving and also probable accidents. As an alternative, this PhD research adapted a fast Newton/GMRES solver to solve the arising optimal control problem, in order to reduce the computational burden of the controller. Furthermore, HIL experiments were performed to show the real-time performance of the controller on a prototype vehicle ECU. The proposed controller in the third chapter enabled the baseline PHEV to perform safe and efficient car-following, which could also be used for the control of a platoon's leader. The HIL experiments showed the real-time implementation capability of the presented controller. Turnaround time of the controller in these experiments were below $700\mu s$ which means that it can be implemented on a real vehicle ECU with no concern about computational constraints.

The fourth chapter was devoted to an important issue in vehicle platooning, which is the simultaneous satisfaction of frequency-domain criteria, namely string stability, and time-domain criteria that is the satisfaction of the safety, comfort and actuation constraints. CACC is the enabling technology of vehicle platooning, that could offer many benefits to transportation systems. Most of the studies on CACC consider string stability as their main concern to achieve a safe platooning system. While string stability is an important factor, it is certainly not enough for guaranteed safe platooning. To achieve a safe vehicle platooning system, it is essential to guarantee the string stability of the vehicle platoon while handling the safety, comfort, and performance constraints. This research presented a distributed RG approach to the constraint handling of vehicle platoons equipped with CACC. This method and similar approaches had never been used in this context before. First, a string stable platoon was designed based on a frequency-domain approach. Second, an RG was designed that sits behind the controlled system and keeps the output inside the defined constraints. RG does not change the behavior of the controlled system; therefore, the platoon remains string stable based on the frequency-domain design. Only when there is a possibility of violating the defined constraints, the RG would intervene to push the system back into the constraints. This way, both the frequency-domain and time-domain requirements are satisfied. Moreover, this research developed a novel type of RG that can handle alternating references and void output admissible sets, by introducing a predictive version of RG.

To improve the platoons energy economy, an ecological predictive controller was used for controlling the leader, using the NMPC method, assuming it is a plug-in hybrid electric vehicle. It was shown here, that string stability will result in lower energy consumption for the platoon and, in fact, unlike what many other research works suggest, there is no need to design an ecological predictive controller for each individual vehicle in the platoon.

Evaluations performed with a platoon model, constructed using high-fidelity models of the baseline vehicle, showed that the proposed method is able to simultaneously maintain the string stability and platooning constraints while improving the total energy economy of the entire platoon. Moreover, the results of the hardware-in-the-loop testing demonstrated the performance of the proposed controller in the real-time application and on a prototype vehicle ECU. The turnaround time of the controller in HIL experiments was below 1 ms at all times, which means that this controller can be implemented on a real vehicle ECU for practical real-time control of connected PHEVs.

The fifth chapter explained a direct model reference adaptive control (MRAC) approach heterogeneous platooning. Most of the studies on vehicle platooning consider homogeneous

platoons for their control design and synthesis. This is due to the overall complexity of the CACC problem, which can be reduced by assuming homogeneity in the platoon. Although the homogeneity assumption would reduce the complexity of CACC design, to have a more practical platooning system, it is also necessary to consider its interaction with heterogeneous platoons. This research proposed an adaptive control approach to the platooning of heterogeneous vehicles. First, a nonlinear dynamical model was presented that explained a heterogeneous platoon subjected to external disturbances. Next, a direct model reference adaptive controller (MRAC) was designed that enforced a string stable behavior on the vehicle platoon despite different dynamical models of the platoon members and the external disturbances acting on the systems. The proposed method estimated the controller coefficients online to adapt to the disturbances such as wind, changing road grade and also to different vehicle dynamic behaviors.

To address the real-time capability issue, HIL experiments were performed to validate the performance of the controllers on a prototype vehicle ECU. The proposed controller was evaluated with a high-fidelity heterogeneous platoon model with injected communication and radar delays to ensure that this controller can maintain string stability despite dynamic differences and alternating environment. Also, different communication topologies was considered to show the performance of this method for different types of CACC. The turnaround time of this controller was below $50\mu s$ at all times during the HIL experiments.

The contributions of this research can be summarized as follows:

- Developed an AT-NMPC with two separate models to guaranty robust constraints handling while adapting to changes to maintain performance.
- Employed the AT-NMPC method to design a robust and ecological adaptive autonomous cruise control system for PHEV.
- Adapted a fast newton/GMRES solver to solve the AT-NMPC nonlinear optimization problem in real-time and performed HIL experiments to show the real-time implementation capability of this controller.
- Developed a real-time distributed predictive RG that can handle alternating references.
- Employed the RG method to design a consternated string stable platoon with the simultaneous satisfaction of the frequency-domain and time-domain platooning criteria.

- Performed HIL experiment on a high-fidelity vehicle platoon model to show the real-time implementation capability of the designed Eco-CACC.
- Proposed a direct model reference adaptive control approach to heterogeneous platooning in order to achieve string stability despite dynamic differences and external disturbances.
- Performed HIL experiment to evaluate the designed MRAC based heterogeneous platoon controller on a high-fidelity heterogeneous platoon model.

6.2 Recommended future directions

This thesis proposed robust and adaptive controllers for safe and efficient driving of connected PHEV. However, further research is still required to expand this study. This section provides some recommended future directions for this work.

1. Ecological Autonomous Cruise Control for car-following
    - Employ robust calibration methods to find the optimum values of the cost function's weightings for the uncertain environment and vehicle model.
    - Study the effect of different spacing policies on the car-following performance especially nonlinear spacing policies.
    - Employ traffic light data from V2I communications to improve car-following performance.
2. Ecological Cooperative Autonomous Cruise Control for platooning
    - Combine the designed platoon with an AT-NMPC controlled leader instead of an NMPC controlled leader.
    - Study the effect of different spacing policies on the platooning performance.
    - Study the fault tolerance capability of the RG controlled platoon especially in the presence of packet loss and connection loss.
    - Employ a traffic simulator to validate the designed controller in different complex traffic scenarios.

- Perform field tests to validate the designed controller on a real vehicle and in interaction with a real-world environment.

3. Direct MRAC for heterogeneous platooning

   - Study the effect of different spacing policies on the heterogeneity of vehicle platoons.
   - Perform control synthesis to find the optimum network topology for a vehicle platoon.
   - Develop platoon formation algorithms in order to allow vehicles to leave or join the vehicle platoon.

References

[1] Ottmar Edenhofer, Ramón Pichs-Madruga, Youba Sokona, E Farahani, S Kadner, K Seyboth, A Adler, I Baum, S Brunner, and P Eickemeier. Climate change 2014: Mitigation of climate change. *Working group III contribution to the fifth assessment report of the Intergovernmental Panel on Climate Change. UK and New York*, 2014.

[2] Assad Alam. *Fuel-efficient heavy-duty vehicle platooning*. PhD thesis, KTH Royal Institute of Technology, 2014.

[3] Eoin Ó Broin and Céline Guivarch. Transport infrastructure costs in low-carbon pathways. *Transportation Research Part D: Transport and Environment*, 55:389–403, 2017.

[4] Nadine Unger, Tami C Bond, James S Wang, Dorothy M Koch, Surabi Menon, Drew T Shindell, and Susanne Bauer. Attribution of climate forcing to economic sectors. *Proceedings of the National Academy of Sciences*, 107(8):3382–3387, 2010.

[5] Terry Dinan and David H Austin. *Reducing gasoline consumption: Three policy options*. Congressional Budget Office, 2002.

[6] Ali Emadi. *Advanced electric drive vehicles*. CRC Press, 2014.

[7] Aneesh Paul, Rohan Chauhan, Rituraj Srivastava, and Mriganka Baruah. Advanced driver assistance systems. Technical report, SAE Technical Paper, 2016.

[8] Joyoung Lee and Byungkyu Park. Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment. *IEEE Transactions on Intelligent Transportation Systems*, 13(1):81–90, 2012.

[9] Sindhura Buggaveeti. Dynamic Modeling and Parameter Identification of a Plug-in Hybrid Electric Vehicle. sep 2017.

[10] C. E.Sandy Thomas. *Greenhouse gases by sector*, volume 35. 2017.

[11] Mahyar Vajedi and Nasser L Azad. Ecological adaptive cruise controller for plug-in hybrid electric vehicles using nonlinear model predictive control. *IEEE Transactions on Intelligent Transportation Systems*, 17(1):113–122, 2016.

[12] T Hummel, M Kühn, J Bende, and A Lang. Advanced driver assistance systems: an investigation of their potential safety benefits based on an analysis of insurance claims in Germany. *German Insurance Association Insurers Accident Research, Research Report FS*, 3, 2011.

[13] Bart Van Arem, Member Ieee, Cornelie J G Van Driel, and Ruben Visser. The impact of Cooperative Adaptive Cruise Control on traffic flow characteristics. *Intelligent Transportation Systems, IEEE*, 7(4):429–436, 2006.

[14] David Schrank., Bill Eisele., Tim Lomax., and Jim Bak. 2015 Urban Mobility Scorecard. Technical Report August, aug 2015.

[15] Petros A. Ioannou and C. C. Chien. Autonomous intelligent cruise control. *IEEE Transactions on Vehicular Technology*, 42(4):657–672, nov 1993.

[16] Datta Godbole, Natalia Kourjanskaia, Raja Sengupta, and Marco Zandonadi. Breaking the highway capacity barrier: Adaptive cruise control-based concept. *Transportation Research Record: Journal of the Transportation Research Board*, (1679):148–157, 1999.

[17] Arne Kesting, Martin Treiber, Martin Schönhof, and Dirk Helbing. Adaptive cruise control design for active congestion avoidance. *Transportation Research Part C: Emerging Technologies*, 16(6):668–683, 2008.

[18] Seungwuk Moon, Ilki Moon, and Kyongsu Yi. Design, tuning, and evaluation of a full-range adaptive cruise control system with collision avoidance. *Control Engineering Practice*, 17(4):442–455, 2009.

[19] Petros Ioannou, Zhigang Xu, S Eckert, D Clemons, and T Sieja. Intelligent cruise control: theory and experiment. In *Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on*, pages 1885–1890. IEEE, 1993.

[20] Chi-Ying Liang and Huei Peng. Optimal adaptive cruise control with guaranteed string stability. *Vehicle system dynamics*, 32(4-5):313–330, 1999.

[21] Md Abdus Samad Kamal, Jun-ichi Imura, Tomohisa Hayakawa, Akira Ohata, and Kazuyuki Aihara. Smart driving of a vehicle using model predictive control for improving traffic flow. *IEEE Transactions on Intelligent Transportation Systems*, 15(2):878–888, 2014.

[22] Bijan Sakhdari, Mahyar Vajedi, and Nasser L. Azad. Ecological Adaptive Cruise Control of a plug-in hybrid electric vehicle for urban driving. In *Intelligent Transportation Systems (ITSC), 2016 IEEE 19th International Conference on*, pages 1739–1744. IEEE, 2016.

[23] Li-hua Luo, Hong Liu, Ping Li, and Hui Wang. Model predictive control for adaptive cruise control with multi-objectives: comfort, fuel-economy, safety and car-following. *Journal of Zhejiang University SCIENCE A*, 11(3):191–201, 2010.

[24] Shengbo Li, Keqiang Li, Rajesh Rajamani, and Jianqiang Wang. Model predictive multi-objective vehicular adaptive cruise control. *IEEE Transactions on Control Systems Technology*, 19(3):556–566, 2011.

[25] M Wang, W Daamen, S P Hoogendoorn, and B Van Arem. Driver assistance systems modeling by model predictive control. In *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 1543–1548. IEEE, 2012.

[26] Ralph H. Rasshofer, M. Spies, and H. Spies. Influences of weather phenomena on automotive laser radar systems. *Advances in Radio Science*, 9:49–60, 2011.

[27] Alebel Arage Hassen. *Indicators for the signal degradation and optimization of automotive radar sensors under adverse weather conditions.* PhD thesis, Technische Universität, 2007.

[28] Zhen Jia, Arjuna Balasuriya, and Subhash Challa. Sensor fusion-based visual target tracking for autonomous vehicles with the out-of-sequence measurements solution. *Robotics and Autonomous Systems*, 56(2):157–176, 2008.

[29] Balázs Németh and Péter Gáspár. Model-based H2/H [infinity] control design of integrated vehicle tracking systems. *Periodica Polytechnica. Transportation Engineering*, 40(2):87, 2012.

[30] Daniele Corona, Ion Necoara, Bart De Schutter, and Ton van den Boom. Robust hybrid MPC applied to the design of an adaptive cruise controller for a road vehicle.

In *Decision and Control, 2006 45th IEEE Conference on*, pages 1721–1726. IEEE, 2006.

[31] Graham C Goodwin, He Kong, Galina Mirzaeva, and Mar\'\ia M Seron. Robust model predictive control: reflections and opportunities. *Journal of Control and Decision*, 1(2):115–148, 2014.

[32] D Limon, I Alvarado, T Alamo, and E F Camacho. Robust tube-based MPC for tracking of constrained linear systems with additive disturbances. *Journal of Process Control*, 20(3):248–260, 2010.

[33] Andrew Gray, Yiqi Gao, J Karl Hedrick, and Francesco Borrelli. Robust predictive control for semi-autonomous vehicles with an uncertain driver model. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 208–213. IEEE, 2013.

[34] Yiqi Gao, Andrew Gray, H Eric Tseng, and Francesco Borrelli. A tube-based robust nonlinear predictive control approach to semiautonomous ground vehicles. *Vehicle System Dynamics*, 52(6):802–823, 2014.

[35] Bijan Sakhdari, Ebrahim Moradi Shahrivar, and Nasser L. Azad. Robust tube-based MPC for automotive adaptive cruise control design. In *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, volume 2018-March, pages 1–6. IEEE, oct 2018.

[36] Linjun Zhang, Jing Sun, and Gábor Orosz. Hierarchical Design of Connected Cruise Control in the Presence of Information Delays and Uncertain Vehicle Dynamics. *IEEE Transactions on Control Systems Technology*, 2017.

[37] Ondrej Santin, Jaroslav Pekar, Jaroslav Beran, Anthony D'Amato, Engin Ozatay, John Michelini, Steven Szwabowski, and Dimitar Filev. Cruise controller with fuel optimization based on adaptive nonlinear predictive control. *SAE International Journal of Passenger Cars-Electronic and Electrical Systems*, 9(2016-01-0155):262–274, 2016.

[38] Ji-Wook Kwon and Dongkyoung Chwa. Adaptive bidirectional platoon control using a coupled sliding mode control method. *IEEE Transactions on Intelligent Transportation Systems*, 15(5):2040–2048, 2014.

[39] Chih-Min Lin and Chiu-Hsiung Chen. Car-following control using recurrent cerebellar model articulation controller. *IEEE Transactions on Vehicular Technology*, 56(6):3660–3673, 2007.

[40] Hyeongcheol Lee and Masayoshi Tomizuka. Adaptive vehicle traction force control for intelligent vehicle highway systems (IVHSs). *IEEE Transactions on Industrial Electronics*, 50(1):37–47, 2003.

[41] Darbha Swaroop, J. Karl Hedrick, and S. B. Choi. Direct adaptive longitudinal control of vehicle platoons. *IEEE Transactions on Vehicular Technology*, 50(1):150–161, jan 2001.

[42] Bijan Sakhdari and Nasser L. Azad. Adaptive Tube-based Nonlinear MPC for Ecological Autonomous Cruise Control of Plug-in Hybrid Electric Vehicles. *eprint arXiv:1803.05942*, mar 2018.

[43] Anil Aswani, Humberto Gonzalez, S Shankar Sastry, and Claire Tomlin. Provably safe and robust learning-based model predictive control. *Automatica*, 49(5):1216–1226, 2013.

[44] Anil Aswani, Patrick Bouffard, and Claire Tomlin. Extensions of learning-based model predictive control for real-time application to a quadrotor helicopter. In *American Control Conference (ACC), 2012*, pages 4661–4666. IEEE, 2012.

[45] Chris J Ostafew, Angela P Schoellig, and Timothy D Barfoot. Learning-based nonlinear model predictive control to improve vision-based mobile robot path-tracking in challenging outdoor environments. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 4029–4036. IEEE, 2014.

[46] Chris J Ostafew, Angela P Schoellig, Timothy D Barfoot, and Jack Collier. Learning-based Nonlinear Model Predictive Control to Improve Vision-based Mobile Robot Path Tracking. *Journal of Field Robotics*, 33(1):133–152, 2016.

[47] Bart Van Arem, Cornelie J G Van Driel, and Ruben Visser. The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Transactions on Intelligent Transportation Systems*, 7(4):429–436, 2006.

[48] Ali Touran, Mark A Brackstone, and Mike McDonald. A collision model for safety evaluation of autonomous intelligent cruise control. *Accident Analysis & Prevention*, 31(5):567–578, 1999.

[49] Neville A Stanton and Mark S Young. Driver behaviour with adaptive cruise control. *Ergonomics*, 48(10):1294–1313, aug 2005.

[50] Cem Hatipoglu, U Ozguner, and Martin Sommerville. Longitudinal headway control of autonomous vehicles. In *Control Applications, 1996., Proceedings of the 1996 IEEE International Conference on*, pages 721–726. IEEE, 1996.

[51] Arnab Bose and Petros Ioannou. Evaluation of the environmental effects of intelligent cruise control vehicles. *Transportation Research Record: Journal of the Transportation Research Board*, (1774):90–97, 2001.

[52] Li-hua Luo, Hong Liu, Ping Li, and Hui Wang. Model predictive control for adaptive cruise control with multi-objectives: comfort, fuel-economy, safety and car-following. *Journal of Zhejiang University SCIENCE A*, 11(3):191–201, 2010.

[53] Mahyar Vajedi and Nasser L. Azad. Ecological Adaptive Cruise Controller for Plug-In Hybrid Electric Vehicles Using Nonlinear Model Predictive Control. *Intelligent Transportation Systems, IEEE Transactions on*, PP(99):1–10, 2015.

[54] Assad Al Alam, Ather Gattami, and Karl Henrik Johansson. An experimental study on the fuel reduction potential of heavy duty vehicle platooning. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 306–311. IEEE, sep 2010.

[55] Jay Wright Forrester. Industrial dynamics. *Journal of the Operational Research Society*, 48(10):1037–1041, 1997.

[56] M Cantoni, E Weyer, Y Li, S K Ooi, I Mareels, and M Ryan. Control of Large-Scale Irrigation Networks. *Proceedings of the IEEE*, 95(1):75–91, jan 2007.

[57] Pin-Ho Lin, David Shan-Hill Wong, Shi-Shang Jang, Shyan-Shu Shieh, and Ji-Zheng Chu. Controller design and reduction of bullwhip for a model supply chain system using z-transform analysis. *Journal of Process Control*, 14(5):487–499, 2004.

[58] Alex A Kurzhanskiy, Pravin Varaiya, and Others. Ellipsoidal Toolbox, EECS Dept. *Univ. of California, Berkeley, CA, Tech. Rep. UCB/EECS-2006-46*, 2006.

[59] Mike Huang, Hayato Nakada, Ken Butts, and Ilya Kolmanovsky. Robust rate-based Model Predictive Control of diesel engine air path. In *American Control Conference (ACC), 2014*, pages 1505–1510. IEEE, 2014.

[60] Roozbeh Kianfar, Paolo Falcone, and Jonas Fredriksson. A control matching model predictive control approach to string stable vehicle platooning. *Control Engineering Practice*, 45:163–173, 2015.

[61] Jeroen Ploeg, Nathan van de Wouw, and Henk Nijmeijer. Lp String Stability of Cascaded Systems: Application to Vehicle Platooning. *IEEE Transactions on Control Systems Technology*, 22(2):786–793, 2014.

[62] William B Dunbar and Derek S Caveney. Distributed receding horizon control of vehicle platoons: Stability and string stability. *IEEE Transactions on Automatic Control*, 57(3):620–633, 2012.

[63] Roozbeh Kianfar, Paolo Falcone, and Jonas Fredriksson. Reachability analysis of cooperative adaptive cruise controller. In *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 1537–1542. IEEE, 2012.

[64] Assad Alam, Ather Gattami, Karl H Johansson, and Claire J Tomlin. Guaranteeing safety for heavy duty vehicle platooning: Safe set computations and experimental evaluations. *Control Engineering Practice*, 24:33–41, 2014.

[65] Roozbeh Kianfar, Paolo Falcone, and Jonas Fredriksson. A distributed model predictive control approach to active steering control of string stable cooperative vehicle platoon. volume 7, pages 750–755, 2013.

[66] Uros Kalabic. Reference governors : Theoretical extensions and practical applications PhD defense. 2015.

[67] Ilya Kolmanovsky, Emanuele Garone, and Stefano Di Cairano. Reference and command governors: A tutorial on their theory and automotive applications. In *American Control Conference (ACC), 2014*, pages 226–241. IEEE, 2014.

[68] Christophe Bonnet and Hans Fritz. Fuel Consumption Reduction in a Platoon: Experimental Results with two Electronically Coupled Trucks at Close Spacing. In *trid.trb.org*, 2000.

[69] Debojyoti Mitra and Asis Mazumdar. Pollution control by reduction of drag on cars and buses through platooning Pollution control by reduction of drag on cars and buses. *Int. J. Environment and Pollution*, 30(1):90–96, 2007.

[70] Valerio Turri, Bart Besselink, and Karl H. Johansson. Cooperative Look-Ahead Control for Fuel-Efficient and Safe Heavy-Duty Vehicle Platooning. *IEEE Transactions on Control Systems Technology*, PP(99):1–17, 2016.

[71] Baisravan HomChaudhuri, Ardalan Vahidi, and Pierluigi Pisu. A fuel economic model predictive control strategy for a group of connected vehicles in urban roads. pages 2741–2746, jul 2015.

[72] Baisravan HomChaudhuri, Ardalan Vahidi, and Pierluigi Pisu. Fast Model Predictive Control-Based Fuel Efficient Control Strategy for a Group of Connected Vehicles in Urban Road Conditions. *IEEE Transactions on Control Systems Technology*, PP(99):1–8, 2016.

[73] Mohammad Pirani, Ehsan Hashemi, John W. Simpson-Porco, Baris Fidan, and Amir Khajepour. Graph Theoretic Approach to the Robustness of k -Nearest Neighbor Vehicle Platoons. *IEEE Transactions on Intelligent Transportation Systems*, 18(11):3218–3224, nov 2017.

[74] Fu Lin, Makan Fardad, and Mihailo R Jovanovic. Optimal Control of Vehicular Formations With Nearest Neighbor Interactions. *IEEE Transactions on Automatic Control*, 57(9):2203–2218, sep 2012.

[75] Bijan Sakhdari and Nasser L. Azad. A Distributed Reference Governor Approach to Ecological Cooperative Adaptive Cruise Control. *IEEE Transactions on Intelligent Transportation Systems*, PP(99):1–12, 2017.

[76] Sinan Öncü, Jeroen Ploeg, Nathan van de Wouw, and Henk Nijmeijer. Cooperative Adaptive Cruise Control: Network-Aware Analysis of String Stability. *IEEE Transactions on Intelligent Transportation Systems*, 15(4):1527–1537, aug 2014.

[77] Darbha Swaroop and J. Karl Hedrick. Constant spacing strategies for platooning in automated highway systems. *Journal of dynamic systems, measurement, and control*, 121(3):462–470, 1999.

[78] Yihang Zhang, Elias B. Kosmatopoulos, Petros A. Ioannou, and C. C. Chien. Using front and back information for tight vehicle following maneuvers. *IEEE Transactions on Vehicular Technology*, 48(1):319–328, 1999.

[79] Feyyaz Emre Sancar, Bari Fidan, and Jan P. Huissoon. Deadzone switching based cooperative adaptive cruise control with rear-end collision check. In *2015 International Conference on Advanced Robotics (ICAR)*, pages 283–287, 2015.

[80] Cong Wang and Henk Nijmeijer. String Stable Heterogeneous Vehicle Platoon Using Cooperative Adaptive Cruise Control. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 1977–1982. IEEE, sep 2015.

[81] Xiang Gui Guo, Jian Liang Wang, Fang Liao, and Rodney Swee Huat Teo. String stability of heterogeneous leader-following vehicle platoons based on constant spacing policy. In *IEEE Intelligent Vehicles Symposium, Proceedings*, volume 2016-Augus, pages 761–766, 2016.

[82] Yang Zheng, Shengbo Eben Li, Dongsuk Kum, and Feng Gao. Robust control of heterogeneous vehicular platoon with uncertain dynamics and communication delay. *IET Intelligent Transport Systems*, 10(7):503–513, 2016.

[83] Mario Di Bernardo, Alessandro Salvi, and Stefania Santini. Distributed consensus strategy for platooning of vehicles in the presence of time-varying heterogeneous communication delays. *IEEE Transactions on Intelligent Transportation Systems*, 16(1):102–112, feb 2015.

[84] Mauro Fusco, Elham Semsar-Kazerooni, Jeroen Ploeg, and Nathan Van De Wouw. Vehicular platooning: Multi-Layer Consensus Seeking. In *IEEE Intelligent Vehicles Symposium, Proceedings*, volume 2016-Augus, pages 382–387. IEEE, jun 2016.

[85] Elaine Shaw and J. Karl Hedrick. Controller design for string stable heterogeneous vehicle strings. In *Proceedings of the IEEE Conference on Decision and Control*, pages 2868–2875, 2007.

[86] Elaine Shaw and J. Karl Hedrick. String stability analysis for heterogeneous vehicle strings. In *Proceedings of the American Control Conference*, pages 3118–3125, 2007.

[87] Youssef Abou Harfouch, Shuai Yuan, and Simone Baldi. An Adaptive Switched Control Approach to Heterogeneous Platooning with Inter-Vehicle Communication Losses. *IEEE Transactions on Control of Network Systems*, 5870(c):1–10, 2017.

[88] Petros Ioannou and Bari Fidan. *Adaptive Control Tutorial*. Society for Industrial and Applied Mathematics (SIAM), 2006.

[89] Amir Taghavipour. Real-time Optimal Energy Management System for Plug-in Hybrid Electric Vehicles. 2014.

[90] Mahyar Vajedi. *Real-Time Optimal Control of a Plug-in Hybrid Electric Vehicle Using Trip Information*. PhD thesis, 2016.

[91] Amir Taghavipour, Nasser L Azad, and John McPhee. Real-Time predictive control strategy for a plug-in hybrid electric powertrain. In *Mechatronics*, volume 29, pages 13–27, 2015.

[92] Hao Hu, Guoqing Xu, and Yang Zhu. Hardware-In-the-Loop Simulation of Electric Vehicle Powertrain System. *2009 Asia-Pacific Power and Energy Engineering Conference*, pages 1–5, mar 2009.

[93] Robin Asby and Emil Wolf. Model-free Optimal Control based Intelligent Cruise Control with Hardware-in-the-loop Demonstration. *IEEE Computational Intelligence Magazine*, 12(2):56–69, may 2017.

[94] Jia Liu, Liyan Zhang, Qihong Chen, Shuhai Quan, and Rong Long. Hardware-in-the-loop test bench for vehicle ACC system. In *2017 Chinese Automation Congress (CAC)*, pages 1006–1011. IEEE, oct 2017.

[95] Dietmar Winkler and Clemens Gühmann. Hardware-in-the-Loop simulation of a hybrid electric vehicle using Modelica/Dymola. In *22nd International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium*, pages 1054–1063, 2006.

[96] Ari Hentunen, Jussi Suomela, Antti Leivo, Matti Liukkonen, and Panu Sainio. Full-scale hardware-in-the-loop verification environment for heavy-duty hybrid electric vehicles. *World Electric Vehicle Journal*, 4(1):119–127, sep 2011.

[97] Parisa Golchoubian and Nasser L. Azad. Real-Time Nonlinear Model Predictive Control of a Battery-Supercapacitor Hybrid Energy Storage System in Electric Vehicles. *IEEE Transactions on Vehicular Technology*, 66(11):9678–9688, nov 2017.

[98] Amir Taghavipour, Ramin Masoudi, Nasser L Azad, and John McPhee. High-fidelity modeling of a power-split plug-in hybrid electric powertrain for control performance evaluation. In *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages V001T01A008—-V001T01A008. American Society of Mechanical Engineers, 2013.

[99] Jinming Liu, Huei Peng, and Z. Filipi. Modeling and Control Analysis of Toyota Hybrid System. *Proceedings, 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics.*, pages 24–28, 2005.

[100] Mahyar Vajedi, Amir Taghavipour, and Nasser L Azad. Traction-motor power ratio and speed trajectory optimization for power split PHEVs using route information. pages 301–308, ASME 2012 International Mechanical Engineering Congress and Exposition, 2012. American Society of Mechanical Engineers.

[101] Richard H. Middleton and Julio H. Braslavsky. String Instability in Classes of Linear Time Invariant Formation Control With Limited Communication Range. *IEEE Transactions on Automatic Control*, 55(7):1519–1530, 2010.

[102] Ilya Kolmanovsky and Elmer G Gilbert. Theory and computation of disturbance invariant sets for discrete-time linear systems. *Mathematical problems in engineering*, 4(4):317–367, 1998.

[103] Luigi Chisci, J Anthony Rossiter, and Giovanni Zappa. Systems with persistent disturbances: predictive control with restricted constraints. *Automatica*, 37(7):1019–1028, 2001.

[104] Carl T Kelley. *Iterative methods for optimization*. SIAM, 1999.

[105] Sadegh Tajeddin. *Automatic Code Generation of Real-Time Nonlinear Model Predictive Control for Plug-in Hybrid Electric Vehicle Intelligent Cruise Controllers*. PhD thesis, University of Waterloo, 2016.

[106] Sadegh Tajeddin and Nasser L Azad. Ecological Cruise Control of a Plug-in Hybrid Electric Vehicle: A comparison of different GMRES-based Nonlinear Model Predictive Controls. In *American Control Conference (ACC), 2017*, pages 3607–3612. IEEE, 2017.

[107] Kiminao Kogiso and Kenji Hirata. Reference governor for constrained systems with time-varying references. *Robotics and Autonomous Systems*, 57(3):289–295, 2009.

[108] Michael C Grant and Stephen P Boyd. Graph implementations for nonsmooth convex programs. In *Recent advances in learning and control*, pages 95–110. Springer, 2008.

[109] Michael Grant, Stephen Boyd, and Yinyu Ye. {CVX}: {Matlab} software for disciplined convex programming. 2008.

[110] Sadegh Tajeddin, Mahyar Vajedi, and Nasser L Azad. A newton/gmres approach to predictive ecological adaptive cruise control of a plug-in hybrid electric vehicle in car-following scenarios. *IFAC-PapersOnLine*, 49(21):59–65, 2016.

[111] Mahyar Vajedi, Maryyeh Chehrehsaz, and Nasser L Azad. Intelligent power management of plug–in hybrid electric vehicles, part I: real–time optimum SOC trajectory builder. *International Journal of Electric and Hybrid Vehicles*, 6(1):46–67, 2014.

[112] Mahyar Vajedi, Maryyeh Chehrehsaz, and Nasser L Azad. Intelligent power management of plug–in hybrid electric vehicles, part II: real–time route based power management. *International Journal of Electric and Hybrid Vehicles*, 6(1):68–86, 2014.

[113] Lennart Ljung. *System identification*. Wiley Online Library, 1999.

[114] Bogdan Marcu and Fred Browand. Aerodynamic forces experienced by a 3-vehicle platoon in a crosswind. Technical report, SAE Technical Paper, 1999.

[115] M Michaelian and F Browand. Field experiments demonstrate fuel savings for close-following, California PATH Program, Institute of Transportation Studies. *University of California at Berkeley*, 2000.

[116] Shengbo Eben Li, Yang Zheng, Keqiang Li, and Jianqiang Wang. An overview of vehicular platoon control under the four-component framework. In *IEEE Intelligent Vehicles Symposium, Proceedings*, volume 2015-Augus, pages 286–291. IEEE, jun 2015.

[117] Hossein Jeddi Tehrani. Study of Disturbance Models For Heavy-duty Vehicle Platooning Master ' s Thesis Study of Disturbance Models For Heavy-duty Vehicle Platooning. (September), 2010.

[118] Ralph H Rasshofer, M Spies, and H Spies. Influences of weather phenomena on automotive laser radar systems. *Advances in Radio Science*, 9(B. 2):49–60, 2011.

[119] Elmer G. Gilbert and Kok Tin Tan. Linear Systems with State and Control Constraints: The Theory and Application of Maximal Output Admissible Sets. *IEEE Transactions on Automatic Control*, 36(9):1008–1020, 1991.

APPENDICES

Appendix A

Output admissible set

As mentioned in chapter-4, a reference governor is a nonlinear controller that sits behind a controlled stable system and enforces the constraints by modifying the reference. RG works based on a system model and output admissible set that is calculated based on defined output constraints. Here, we will explain how we calculated the output admissible set based on the works presented in [3, 66, 119]. We assumed the following linear system with bounded disturbance:

$$\begin{aligned} x(t+1) =& A_k x(t) + B_k u(t) + B_w w(t) \\ y(t) =& Cx(t) + Du(t) + D_w w(t) \in \mathbb{Y} \ \forall t \in Z^+, \end{aligned} \tag{A.1}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the input or reference vector, $w(t) \in W$ is the bounded additive disturbance acting on the system, Z^+ is the set of nonnegative integers, y is the output and $\mathbb{Y}$ is the output constraint set. We are assuming that A_k is stable with eigenvalues inside the unit disk, the system (A.1) is observable and W and $\mathbb{Y}$ are compact convex sets and contain the origin inside them. The output admissible set can be defined as follows:

$$O_\infty = \{(x_0, \bar{u}) \mid y(t; x_0, \bar{u}) \in \mathbb{Y} \ \forall \ t \in Z^+\}, \tag{A.2}$$

where $y(t; x_0, \bar{u})$ is the system's output at time t, starting at point x_0 and with constant reference $\bar{u}$. O_∞ is the set of all initial states and constant references that, if given to the system, will never violate the output's constraint. Having the system model (A.1) with

the initial condition and a constant reference, we are able to calculate the output in each time step using the following equation:

$$y(t) = CA_k^t(x_0 - \Gamma\bar{u}) + (C\Gamma + D)\bar{u} + CA_k^t B_w w(0) + \cdots + CAB_w w(t-1) + D_w w(t), \quad \text{(A.3)}$$

where Γ is the steady-state response for the constant reference for the system A.1 in the absence of disturbance. Using (A.3), we can find the output admissible set:

$$CA_k^t x_0 + (C\Gamma + D)\bar{u} \subset \mathbb{Y} \sim CA_k^t B_w W \sim \cdots \sim CABW \sim D_w W. \quad \text{(A.4)}$$

The pairs $(x_0, \bar{u})$ that satisfy (A.4) are admissible points which together make the output admissible set. This enables us to find finite time admissible sets with the following iterations:

$$\begin{aligned} \mathbb{Y}_{t+1} &= \mathbb{Y}_t \sim CA_k^t B_w W, \\ O_{t+1} &= O_t \cap X_{t+1}, \end{aligned} \quad \text{(A.5)}$$

where $X_t = \{(x_0, \bar{u}) | CA_k^t x_0 + (C\Gamma + D)\bar{u} \in \mathbb{Y}_t\}$. This iteration calculates a finite admissible set for each time step. It continues until the calculated set remains constant. In the case that this set cannot be finitely determined, an inner approximation of the set can be calculated, as follows.

$$\Omega_d = \{\bar{u} \in \mathbb{R}^m \mid (C\Gamma + D)\bar{u} \in \mathbb{Y}_\infty\}. \quad \text{(A.6)}$$

This inner approximation has all the properties of the output admissible set and can be easily calculated specially for polytopic constraints.

Appendix B

Proof of stability and convergence for the proposed MRAC CACC

Here, we want to show that the proposed controller in Equation (5.6) results in bounded states and also convergence to the desired model Equations (5.2). The equation (5.1) represents the dynamical model of the system. If the desired controller coefficients in (5.7) are used to control this model, the desired behavior will be achieved. Therefore, by ignoring the small delay in actuation, the desired model can be written as follows:

$$\dot{x}_{m_i} = Ax_{m_i} + BK_i^*(t)\phi_i(t) + B_p a_{i-1} + B_g F_{r_i}. \tag{B.1}$$

Letting $e_{x_i} = x_i - x_{m_i}$ and $\epsilon_i = a_i - a_{m_i}$, where x_m is from the reference model, and subtracting (B.1) from (5.1), the tracking error equation is obtained:

$$\begin{aligned} \dot{e}_{x_i} &= Ae_{x_i} + B(u(t) - K_i^*(t)\phi_i(t)) \\ &= Ae_{x_i} + B\widetilde{K}_i(t)\phi_i(t), \\ \epsilon_i &= Ce_{x_i}. \end{aligned} \tag{B.2}$$

where $\widetilde{K} = K - K^*$. We define the following Lyapunov candidate function:

$$V(e_{x_i}, \widetilde{K}_i) = \frac{1}{2} e_{x_i}^T Q_c e_{x_i} + \frac{1}{2} \widetilde{K}^T P^{-1} \widetilde{K}, \tag{B.3}$$

where $P = P^T > 0$ and $Q_c = Q_c^T > 0$ is chosen such that it satisfies:

$$\begin{aligned} Q_c A + A^T Q_c &= -\psi\psi^T - \nu_c \zeta_c, \\ Q_c B &= C, \end{aligned} \tag{B.4}$$

where $C = [0\ 0\ 1]$ is the output from the vehicle model $a_i = Cx_i$, $\zeta_c = \zeta_c^T > 0$, $\nu_c > 0$ is a small constant and ψ is a vector, implied by the Meyer-Kalman-Yakubovich lemma [88]. Taking the derivative of $V(t, \widetilde{K})$ with respect to time along the solution of Equation (B.3):

$$\dot{V} = e_{x_i}^T Q_c \dot{e}_{x_i} + \widetilde{K}^T P^{-1} \dot{\widetilde{K}} - \frac{1}{2} \widetilde{K}^T P^{-1} \dot{P} P^{-1} \widetilde{K}. \tag{B.5}$$

By substituting (B.4) and (B.2) in (B.5), Lyapunov time derivative can be rewritten as:

$$\begin{aligned} \dot{V} = -\frac{1}{2} e_{x_i}^T \psi\psi^T e_{x_i} - \frac{1}{2} \nu_c e_{x_i}^T \zeta_c e_{x_i} + e_{x_i}^T Q_c B \widetilde{K}_i \phi + \\ \epsilon_i \widetilde{K}_i^T \phi_i - \frac{1}{2} \widetilde{K}_i P^{-1} \beta \widetilde{K}_i + \frac{1}{2} \frac{\widetilde{K}_i^T \phi_i \phi_i^T \widetilde{K}_i}{m_s^2}. \end{aligned} \tag{B.6}$$

considering that $\epsilon_i = -\frac{\widetilde{K}_i \phi_i}{m_s^2}$ and $Q_c B = C$ in (B.7) and also the fact the P is positive definite and bounded because of the projection, we will have:

$$\dot{V} = -\frac{1}{2} e_{x_i}^T \psi\psi^T e_{x_i} - \frac{1}{2} \nu_c e_{x_i}^T \zeta_c e_{x_i} - \frac{1}{2} \widetilde{K}_i P^{-1} \beta \widetilde{K}_i - \frac{3}{2} \epsilon_i^2 m_s^2 \leq 0. \tag{B.7}$$

which implies that V, e_{x_i} and $\widetilde{K} \in \mathcal{L}_\infty$ and because a_{m_i} and $K_i^* \in \mathcal{L}_\infty$, a_i and K_i will be bounded. Moreover, based on equation (B.2), the boundedness of e_{i_x} results in $\dot{e}_{i_x}, \epsilon_i, \dot{\epsilon}_i \in \mathcal{L}_\infty$. Moreover, since $\dot{V} \leq 0$ and V has a lower bound, from (B.7) it implies that $e_{x_i}, \epsilon_i, \widetilde{K} \in \mathcal{L}_2$ and, therefore, $\widetilde{K} \in \mathcal{L}_\infty \cap \mathcal{L}_2$, which implies that $\widetilde{K} \to 0$ as $t \to \infty$.