Dynamics and Model-Predictive Anti-Jerk Control of Connected Electric Vehicles

by

Mohit Batra

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	Dr. Fengjun Yan (Associate Professor)
Supervisor 1	Dr. John McPhee (Professor)
Supervisor 2	Dr. Nasser Lashgarian Azad (Associate Professor)
Internal Member 1	Dr. Gordon Savage (Professor)
Internal Member 2	Dr. Eihab Abdel-Rahman (Professor)
Internal/External Member	Dr. Soo Jeon (Associate Professor)

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 models and controllers presented in this thesis were developed by me. The experimental testing of the Rav4EV were also designed and conducted with the help of our laboratory engineer, Chris Shum.

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

- Mohit Batra, John McPhee and Nasser L Azad (2017). Real-time model predictive control of connected electric vehicles, Journal of Vehicle System Dynamics (Under Review).

  Contributions of Chris Shum: technical help with the experiments

- Mohit Batra, John McPhee and Nasser L Azad (2017). Anti-jerk dynamic modeling and parameter identification of an electric vehicle based on road tests, ASME Journal of Computational and Nonlinear Dynamics (Under Review).

  Contributions of Chris Shum: technical help with the experiments

- Mohit Batra, Anson Maitland, John McPhee and Nasser L Azad (2017). Non-Linear Model Predictive Anti-Jerk Cruise Control for Electric Vehicles with Slip-Based Constraints. American Control Conference, American Control Conference, IEEE, June 27–29, 2018, Wisconsin Center, Milwaukee, USA (Under Review).

  Contributions of Anson Maitland: Development of the NMPC solver

- Mohit Batra, John McPhee and Nasser L Azad (2017). Anti-Jerk Model Predictive Cruise Control for Connected EVs with Changing Road Conditions. 2017 Asian Control Conference, 17 - 20 December 2017, Gold Coast, Australia.

  Contributions of Chris Shum: technical help with the experiments

- Sindhura Buggaveeti, Mohit Batra, John McPhee and Nasser L Azad (2017). Longitudinal Vehicle Dynamics Modeling and Parameter Estimation for Plug-in Hybrid Electric Vehicle, SAE International Journal of Vehicle Dynamics, Stability, and NVH 1(2):289-297, 2017, doi.org/10.4271/2017-01-1574.

  Contributions of Chris Shum: technical help with the experiments

- Mohit Batra, John McPhee and Nasser L Azad (2016). Parameter Identification for a Longitudinal Dynamics Model based on Road tests of an Electric Vehicle,

ASME. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 18th International Conference on Advanced Vehicle Technologies, Charlotte, North Carolina, USA, August 21–24, 2016, doi:10.1115/DETC2016-59732.

Contributions of Chris Shum: technical help with the experiments

- S Tajeddin, Mohit Batra, John McPhee, Nasser L Azad and R.A. Fraser (2015). An Adaptive and Fast Control Strategy for Antilock Braking System. ASME 2015, Dynamic Systems and Control Conference, Columbus, Ohio, USA, October 28–30, 2015, doi:10.1115/DSCC2015-9779.

  Contributions of S Tajeddin: adaptive strategy for control development

Abstract

Electric Vehicles (EVs) develop high torque at low speeds, resulting in a high rate of acceleration. However, the rapid rise in torque of an electric motor creates undesired torsional oscillations, with vehicle jerk arising as a result of wheel slip or flexibility in the half-shaft. These torsional oscillations in the halfshaft lead to longitudinal oscillations in the wheels, thus reducing comfort and drivability.

In this research, we have designed an anti-jerk longitudinal dynamics controller that damps out driveline oscillations and improves the drivability of EVs with central-drivetrain architecture. The anti-jerk longitudinal dynamics controller has been implemented for both traction and cruise control applications. We have used a model predictive control (MPC) approach to design the controller since it allows us to deal with multiple objectives in an optimal sense. The major scope of this research involves modeling, parameter identification, design and validation of the longitudinal dynamics controller. The real-time implementation has been demonstrated using hardware-in-the-loop experiments utilizing fast MPC solvers.

The MapleSim software, which utilizes symbolic computation and optimized-code generation techniques to create models that are capable of real-time simulation, has been used to develop the longitudinal dynamics plant model. Road tests have been conducted on our test vehicle, a Toyota Rav4 electric vehicle (Rav4EV), to identify the parameters for the longitudinal dynamics model. Experimental data measured using a vehicle measurement system (VMS), global-positioning system (GPS), and inertial measurement unit (IMU) was used for parameter identification. Optimization algorithms have been used to identify the model parameters. A control-oriented model of the EV, which includes a flexible halfshaft and effect of wheel-slip transients, has been developed with the aim of controlling driveline oscillations.

The MPC-based anti-jerk traction controller regulates the motor torque corresponding to the accelerator pedal position, to serve the dual objectives of traction and anti-jerk control. The performance of this controllers has been compared to that of other controllers in the literature. Since most traction controllers are on-off controllers and are only activated when wheel slip exceeds a desired limit, they are not effective in anti-jerk control. The MPC-based anti-jerk controller is able to serve multiple objectives related to anti-jerk as well as traction, and is therefore superior to other controllers.

A unified design combining the upper and lower level MPC-based cruise controller has also been formulated to meet the anti-jerk objective during cruise control. The cruise controller has been designed such that it is adaptive to changes in road friction conditions.

The efficacy of both traction and cruise controllers has been demonstrated through model-in-the-loop simulation, and the real-time capability has been demonstrated through hardware-in-the-loop experiments.

Acknowledgements

I would like to thank.....

my supervisors, Professor John McPhee and Professor Nasser Lashgarian Azad for their unwavering support and valuable guidance during the course of my PhD at University of Waterloo. They not only provided me with a healthy learning environment to grow professionally, but also encouraged me constantly to perform better.

our laboratory engineers, Chris Shum for his support in setting up the experimental measurement system on the Rav4EV and Stefanie Bruinsma for helping with testing in the Green and Intelligent Automotive (GAIA) laboratory.

the members of my committee: Prof Gordon Savage, Prof Soo Jeon, Prof Eihab Abdel-Rahman and Prof Fengjun Yan from McMaster University for taking the time to read my thesis and participate in my final defense.

all members of the Motion Research Group (MoRG), especially Dr Joydeep Banerjee and Amer Keblawi for helping me with MapleSim whenever I needed and providing a friendly and intellectually stimulating environment within our lab.

and, last but not least, my parents, Mohinder and Geeta for their constant encouragement, and my family, Ruchika, Adrit and Mahi Batra for their tremendous support during the course of my PhD.

to my wife Ruchika

Table of Contents

List of Figures

List of Tables

Nomenclature

Characters

Symbol	Description	Unit
J_v	Vehicle jerk	m/s^3
v	Vehicle speed	m/s
v_{op}	Vehicle operating speed	m/s
ω_w	Angular speed of wheel	rad/s
θ_w	Wheel rotation angle	rad
Wh_{sp}	Wheel speed	m/s
λ	Wheel slip ratio	–
λ_t	Transient wheel slip ratio	–
μ	Normalized longitudinal force on tire	–
T_m	Motor torque	Nm
ω_m	Motor speed	rad/s
ω_{m_b}	Motor base speed	rad/s
θ_m	Motor rotation angle	rad
T_t	Transmission torque of drivetrain	Nm
η_t	Transmission efficiency of drivetrain	%
gr	Gear ratio	–
T_w	Wheel torque	Nm
J_m	Motor inertia	$Kg.m^2$
J_w	Wheel inertia	$Kg.m^2$
M	Vehicle Mass	Kg

I_y	Pitch inertia of vehicle	Kg.m^2
J_{veh}	Rotary inertia of vehicle	Kg.m^2
F_x, F_{xf}, F_{xr}	Longitudinal traction force: Total, front and rear wheels	N
F_a	Longitudinal traction force measured by VMS at the wheel center	N
F_r, F_{rf}, F_{rr}	Rolling resistance force: Total, front and rear wheels	N
f_{rr}	Rolling resistance force	N
F_d	Aerodynamic drag force	N
F_z, F_{zf}, F_{zr}	Total normal load, load on the front and rear wheels	N
h	Height of CG	N
h_d	Height from ground where the Aerodynamic drag force is assumed to act	N
L	Wheel base - longitudinal distance between front and rear wheels	m
l_f, l_r	Longitudinal distance of front and rear wheel to CG	m
g	Acceleration due to gravity	m/s^2
ρ	Density of air	Kg/m^3
C_d	Coefficient of aerodynamic drag	–
A_f	Frontal area of the vehicle	–
r_{eff}	Dynamic radius of wheel	m
Rl	Longitudinal relaxation length	m
B, C, D, E	Parameters of Pacejka's magic tire formula model	–
C_x	Longitudinal Stiffness of tire in linear model	N/slip
C_f, C_r	Damping of front and rear suspension	Ns/m
K_f, K_r	Stiffness of front and rear suspension	N/m
K_t	Stiffness of tire	N/m
M_s, I_s	Sprung mass in vehicle suspension model	Kg
I_s	Sprung mass inertia in vehicle suspension model	Kg.m^2
z_f, z_r	Vertical displacement of the front and rear sprung mass	m
z_{ft}, z_{rt}	Vertical displacement of the front and rear tire mass	m
u_f, u_r	Vertical displacement based on road profile input	m
M_{ft}, M_{rt}	Mass of the front and rear wheels (unsprung mass)	m
θ	Angle representing pitch motion of sprung mass	deg
T_{hs}	Halfshaft torque	Nm

k	Torsional stiffness of halfshaft	Nm/rad
c	Torsional damping of halfshaft	Nm/rad/s
J_{hs}	Polar Moment of inertia for halfshaft	m^4
L_{hs}	Halfshaft length	m
G	Modulus of rigidity	N/m^2
u_{th}	Throttle pedal position	%
t_d	Motor time-delay constant	s
k_1, k_2	Motor model torque gain and flux weakening constants	–
C_x	Longitudinal stiffness of tire in linear model	N/slip
α	Relaxation factor-MTTE control	–
K_{in}	Integral gain of ISM controller	–
K	Tuning parameter of ISM controller	–
ϕ	Boundary layer thickness of slip	–

List of Abbreviations

Acronym	Description
AC	Alternating current
ACC	Adaptive cruise control
ADAS	Advanced driver assistance aystems
ASR	Acceleration-slip-regulation
CC	Cruise control
CAN	Controller area network
CG	Center of gravity
BLDC	Brushless direct-current
ECU	Electronic control unit
EV	Electric vehicle
FT-MPC	Frozen-time MPC
GAIA	Green and intelligent automotive
GM	Gain margin
GPS	Global position sensor
HIL	Hardware-in-the-loop
IAJ	Integral action of jerk
IA	Integral action
ICE	Internal combustion engine
IMU	Inertial measurement unit
ISM	Integral sliding mode
ITS	Intelligent transportation System
LA-MPC	Look-ahead MPC

LGS	Laser ground sensor
LDV	Laser doppler velocitimeter
MIL	Model-in-the-loop
MPC	Model predictive control
MPCC	Model predictive cruise controller
MTTE	Maximum torque transmissible
NMPCC	Non-linear model predictive cruise controller
NHTSA	National highway traffic safety administration
ODEs	Ordinary differential equations
RTC	Real-time computer
RMS-TE	RMS of tracking-error
RMS-STE	Root-mean-square of slip tracking error
RMS-VTE	Root-mean-square of velocity tracking error
TMMC	Toyota motor manufacturing Canada
UDDS	Urban dynamometer driving schedule
VMS	Vehicle measurement system
VTI	Vehicle-to-infrastructure
VTV	Vehicle-to-vehicle
WFS	Wheel force sensor
WPS	Wheel position sensor

Chapter 1

Introduction

1.1 History of Electric Vehicles

The history of electric propulsion dates back to 1827, when a Slovak-Hungarian priest, Anyos Jedik, powered a tiny car using a crude electric motor [4]. In 1835, Thomas Davenport built a toy electric locomotive. This was followed by Scottish inventor Robert Davidson developing the first full size electric locomotive, 'Galvani' in 1842. The later electric locomotive developed by Davidson in 1873 is often referred to as the first working electric road vehicle. By 1900, only 22 percent of cars were powered by gasoline, while 40 percent were electric and the remaining 38 percent ran on steam [5]. Eventually, improvements in internal combustion engines and the invention of the electric starter made gasoline-powered cars a cheaper option. Growth of gasoline-powered cars manufactured by companies like Ford and General Motors helped lower the prices of these vehicles to almost half that of their electric counterparts. By the 1930s, gasoline- powered cars had taken over the market, with electric cars disappearing from the marketplace.

Electric vehicles regained interest in the late 20^{th} century owing to growing concerns about pollution from gasoline-powered cars. In 1976, the US launched the Electric and Hybrid Vehicle Research Development and Demonstration Act, Public Law 94-413 [6]. Another big program was launched in the USA in 1993, when several government agencies and automotive manufacturers formed a partnership for producing electric vehicles (EVs). The program ended in 2001 with automotive manufacturers having demonstrated but not launched EVs. Rising fuel costs and emission regulations in the 21^{st} century are now mounting pressure on automotive manufacturers to increase production of eco-friendly EVs. Government laws like the Zero Emission Vehicle Mandate passed in California in 1990 are

forcing automotive manufacturers to produce and sell EVs [7].

The market share of EVs is not significant today due to high costs, short driving distance and long charging duration. However, with rapidly reducing cost of batteries [8], the costs of EVs are likely to go down in the future. Constantly depleting energy resources and increasing emission regulations are also likely to attract more interest in EVs. Pure battery-electric vehicles have been available in the market since the 2010s and according to recent data published [9], EV sales grew 60% in 2016 over the year before. Furthermore, all major car manufacturers offer EVs. Modeling and simulation is likely to play an important role in the success of EVs [6].

Electric motors used in EVs have graduated from brushless direct-current (BLDC) motors to alternating current (AC) synchronous machines, and presently, induction motors are being utilised in most EVs. Induction motors are most popular because they are cheap, reliable and robust and can develop high torque in less time with good efficiency as compared to IC engine (ICE) driven vehicles. Motors fitted in EVs have the ability to produce high torque over a range of speeds, and require a relatively simpler powertrain architecture [10]. A representation of the powertrain of a front wheel driven EV, directly coupled to the wheels through a reduction gearbox and halfshafts, is shown in Fig 1.1.

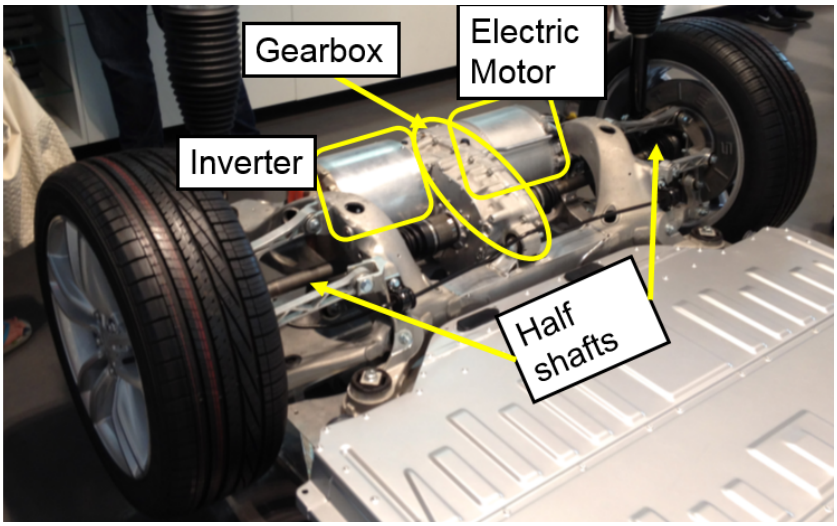

Figure 1.1: Powertrain of front wheel driven EV [1]

1.2 Model-Based Design

Model-based design is a tool that eases the process of selection of components by allowing early design of powertrain components and their control systems in the development cycle. Model-based design is also used effectively for validation of control systems. With widespread use of electronic control, a malfunction in torque control [11] could be a potential hazard to the driver and the vehicle. Therefore, it is critical that control systems are validated before implementation on the vehicle. Further, as model-based design and testing is a collaborative work between testers, system engineers, and programmers, the development process helps in establishing a common framework for communication through the design process supporting the development cycle, which all engineers can comprehend easily.

Automotive manufacturers have now moved to virtual prototyping where software are used to physically model the real component, and hardware is available for testing and validating the control design. Virtual prototyping not only lowers the expenses during design stage, but provides an environment for testing and improving the quality of the physical component before it is constructed. Several software packages such as Adams, MapleSim, Dymola and CarSim have been developed for modeling and simulation of dynamic systems.

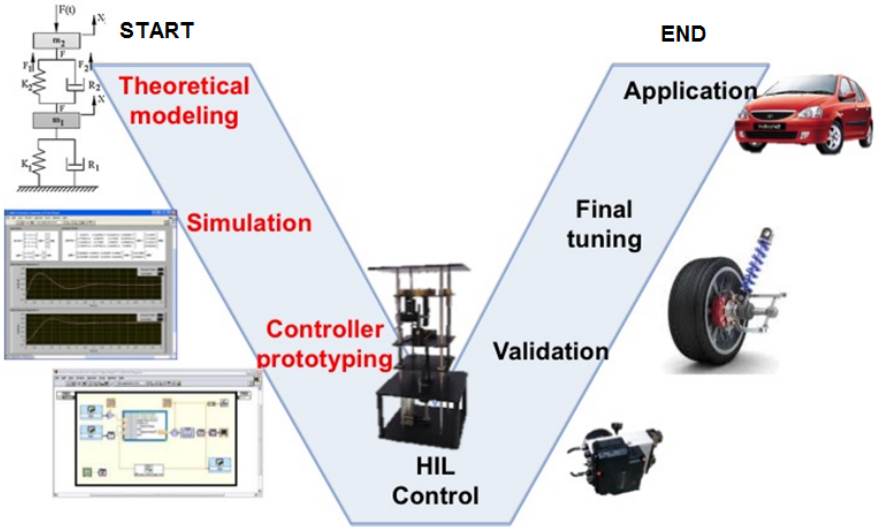

Figure 1.2: Model based design cycle for automotive control systems [2]

A representation of the model based development cycle [2] can be seen in Fig 1.2. The figure shows a common representation of the model development process for complex automotive control systems. The 'V' shape outlines the two broad groupings of phases

connected through Hardware in-the-Loop (HIL) testing. The left arm of the 'V' involves modeling, simulation and developing a controller prototype. The right arm of the 'V' involves development, tuning and optimization of the controller on the vehicle where it is to be fitted.

Acausal Modeling

In this research, MapleSim, a software based on graph-theoretic modeling [12] is used to create a longitudinal dynamics plant model of the Rav4EV. MapleSim is a multi-domain modeling software that can be used to develop and integrate dynamic models from different domains such as mechanical, electrical and hydraulic. Models developed in MapleSim are acausal, which implies that unlike traditional input/output blocks, the connected components are adaptable to the direction of data flow. Acausal modeling is more suitable for modeling complex systems such as vehicle powertrains [13].

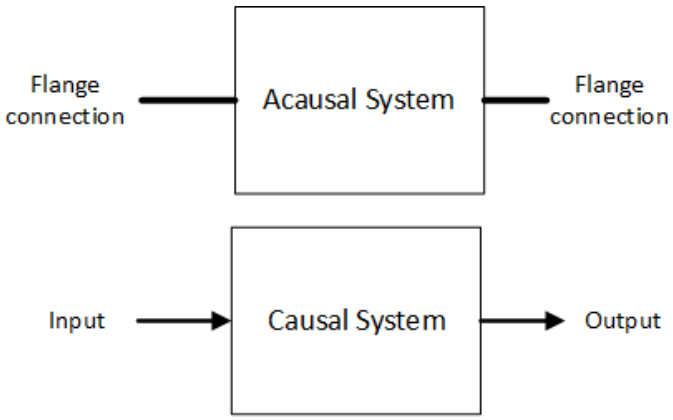

Figure 1.3: Comparison of acausal and causal system component

Fig 1.3 brings out the difference between causal and acausal system components. It can be seen that as acausal components have only flange ports, they can be modeled independent of the direction of signal flow. Output signals can be obtained by connecting sensors in series or in parallel with the components. Another feature of the MapleSim software is that it can generate symbolic equations for the model that can be used for parameter optimization, model reduction and sensitivity analysis. It also has the ability to generate a highly optimized C-code that can be used in Matlab/Simulink for controls development. The development of controllers for Model-in-the-loop (MIL) simulation and HIL experiments is undertaken in Matlab/Simulink environment in this thesis. Matlab and Simulink are software developed by

MathWorks that provide a numerical computing environment for developing controllers and simulating models. In this research, the different controllers developed in Matlab/Simulink environment have been used to control the models developed using MapleSim.

1.3 Driveline Oscillations in Electric Vehicles

Electric motors develop peak torque almost instantaneously, and over a wide range of speeds, have a high efficiency, and a high power to weight ratio, and require smaller size powertrains, which makes the powertrain design much simpler as compared to ICE driven vehicles. However, these characteristics also make EVs more prone to driveline oscillations as compared to ICE driven vehicles. In this section, we define driveline oscillations, discuss factors that cause driveline oscillations, and compare design factors that make EVs more susceptible to driveline oscillations as compared to ICE driven vehicles.

1.3.1 Driveline Oscillation

Drivetrain oscillations is a phenomenon of fluctuating vehicle speed caused by torsional vibrations in the vehicle powertrain. These oscillations are also called shunt/shuffle vibrations. Shunt can be described as the initial response or jerk that arises due to high rate of change of driveline torque, while shuffle is a low frequency vibration that follows shunt [14]. The low frequency shuffle vibration is typically in the range of 1-10 Hz and corresponds to the first natural frequency of the driveline [15].

1.3.2 Causes of Driveline Oscillation

Driveline oscillations are mainly caused by elasticity in vehicle powertrain components such as halfshafts and by wheel-slip dynamics [16]. The driveline oscillations are excited during tip-in and tip-out maneuvers (push-in and release of the accelerator pedal), clutching in or shifting gears, increased backlash in gears and wheel-slip in the tires. EVs fitted with central drive powertrain configurations are especially susceptible to torsional oscillations, resulting in jerk when subjected to high rate of acceleration especially during tip-in and tip-out maneuvers [17].

Fig. 1.4 shows the wheel speed, torque and slip oscillations measured experimentally on our test vehicle, the Rav4EV. It can be seen that the Rav4EV develops peak torque in $\approx 1\ s$

which is accompanied by wheel-slip and torque oscillations.

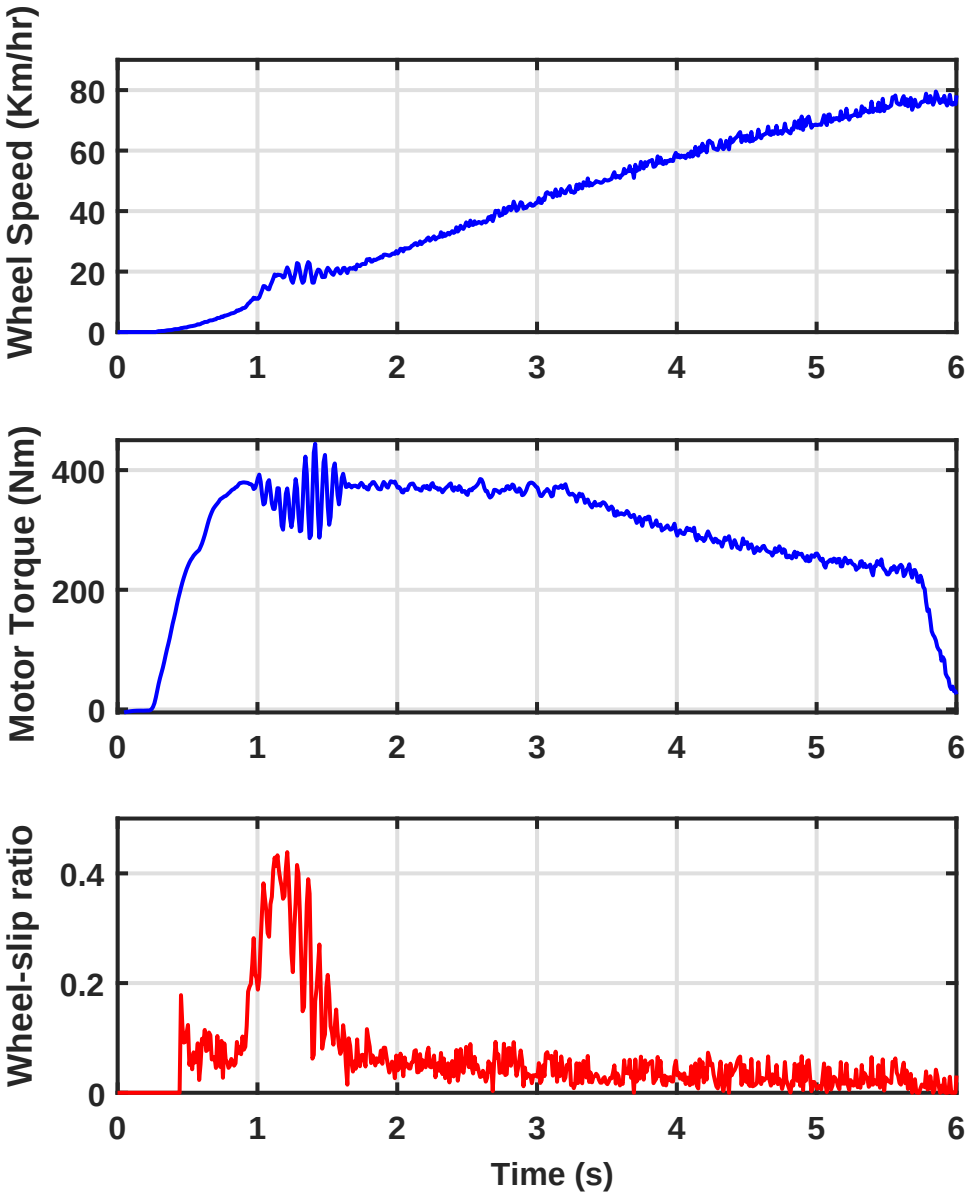

Figure 1.4: Wheel speed, torque and slip oscillations measured experimentally on the Rav4EV

1.3.3 EVs more susceptible to Oscillations

In this section, we compare some of the design characteristics, that make EVs more susceptible to driveline oscillations as compared to ICE driven vehicles [18]. We make this

comparison considering the Lotus Evora platform [3], that is powered by both, an Induction motor (Lotus Evora 414E) as well as an ICE (Lotus Evora S). The following differences exist in the characteristics of an EV and an ICE powered vehicle:

(a) Induction motors develop instantaneous torque, and the torque characteristics remain constant till it reaches its base speed (approx 3000 rpm), followed by a flux weakening region, where torque decay is observed. However, the ICE vehicle starts at 2000 rpm (idling) and torque is approximately constant over the speed range. Therefore, ICE driven Evora S requires to use a reduction gearbox so that torque can be transmitted at speeds lower than 2000 rpm or above 7000 rpm as shown in Fig. 1.6. The comparative torque-speed curves for Lotus Evora 414E EV and an ICE driven Evora S are shown in Fig 1.5.

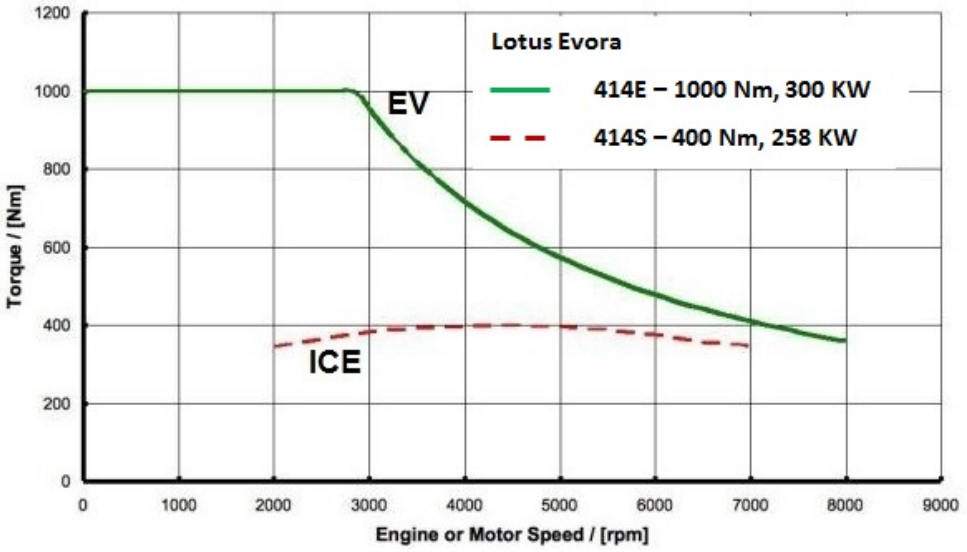

Figure 1.5: Torque requirement of Evora 414E EV vs a ICE driven Evora S [3]

(b) The capability of induction motors to develop torque over a range of speeds makes the powertrain design simpler and smaller as compared to ICEs. Generally, most EVs are fitted with small single-stage reduction gearboxes as compared to relatively heavy multi-stage reduction gearboxes fitted in ICE driven vehicles. The torque characteristics of Evora 4141E electric and an Evora S ICE with multi-stage reduction gearbox can be seen in Fig. 1.6.

(c) The EV design generally does not cater for a clutch or torque converter and therefore the powertrain is rigidly connected to the wheels through a reduction gear. Not including

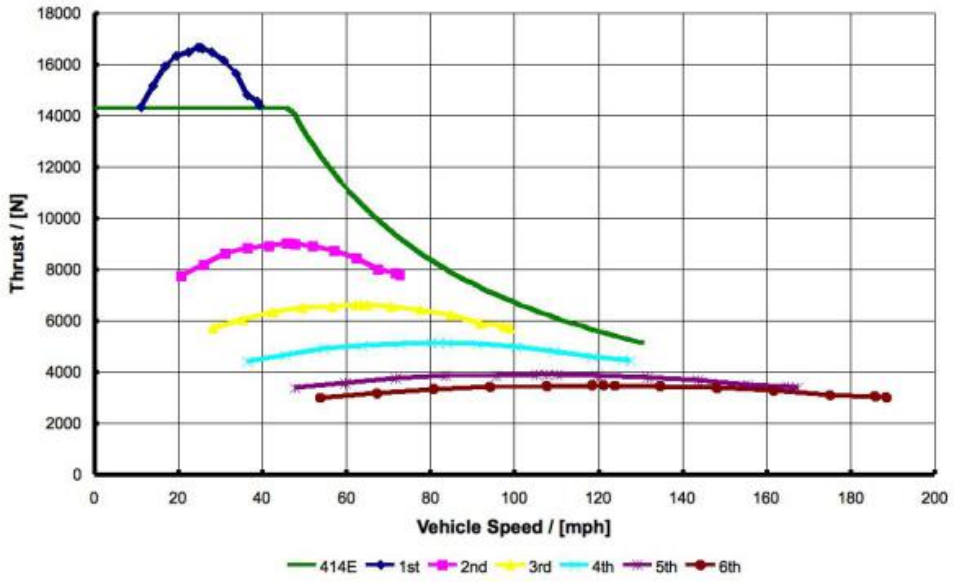

Figure 1.6: Evora 4141E electric drivetrain vs Evora S ICE with gearbox [3]

a clutch in the powertrain design, as in the case of an ICE driven vehicle, leads to low damping in the EV drivetrain. Low driveline damping can lead to sustained driveline oscillations in EVs [19].

It can be seen from the above example that EVs have lower powertrain inertia, and lower damping as compared to ICE vehicles. Further, ICEs are usually equipped with dampers to damp out the high torque oscillations occurring at the firing frequency of the engine [18]. These factors make EVs more susceptible to driveline oscillations.

1.4 Drivability

The low frequency longitudinal oscillations are unpleasant to the human body as resonant frequencies of some of the internal organs are in the same range [20], thus affecting drivability. Drivability as defined in [21] is a subjective perception of the interactions between the driver and vehicle. A literature survey in the area of drivability has indicated that, although there are no standards prescribed, it is commonly evaluated from a driver's perspective. Although drivability was associated with fuel consumption and emissions during the 1970s, car manufacturers in the last decade started focusing on several vehicle characteristics like responsiveness, operating smoothness and driving comfort. Drivability by most car

manufacturers is evaluated subjectively by having their experienced test drivers fill out form sheets [21].

In the past few years, the focus has shifted to developing objective methods to assess drivability. AVL DRIVE [22], a software based measurement system uses more than 100 criteria to evaluate drivability objectively. The typical operation modes used by AVL DRIVE to evaluate drivability include acceleration, braking, tip-in and gear change while assessment criteria include measurement of vibration response such as shuffle, jerk, oscillations and overshoots.

The driveline oscillations are quantified in terms of jerk index [21] on a scale of 1 to 10 to assess drivability objectively. Vehicle jerk can be expressed mathematically as the derivative of acceleration or the second derivative of velocity:

$$J_v = \frac{d}{dt}(\dot{v}) = \frac{d^2}{dt^2}(v) \tag{1.1}$$

It is not only important to assess jerk with a driver operation perspective, but also for advanced driver assistance systems (ADAS) such as cruise control. According to a survey report by the National Highway Traffic Safety Administration (NHTSA) [23], high jerk in cruise controllers is preventing users from using cruise-control; NHTSA since formed an important criteria for evaluating the performance of cruise control systems. In this research, the maximum acceptable longitudinal jerk for good drivability is considered as 2-3 m/s^3 [24, 25].

1.5 Motivation and Scope of Research

High power-to-weight ratios and high initial acceleration due to relatively flat torque-speed characteristics give EVs a significant edge over the ICEs. However, rapid build up of torque by an electric motor creates undesired torsional oscillations, leading to problems with regard to comfort and drivability. Not only is the driver comfort affected, torque oscillations can lead to wheel slip and increased fatigue of drivetrain components.

In the past, driveline oscillation control did not attract much attention as ICE driven vehicles are less prone to these oscillations as compared to EVs. Further, motor controllers are typically designed to produce torque independently of the mechanical state of the system and do not actively damp out the oscillations [18]. However, as the number of EVs is increasing, a number of torque control strategies are being researched to improve the

drivability of EVs.

In this research, we have designed torque management strategies for traction and cruise control applications, to damp out driveline oscillations actively, and to improve drivability of EVs. A model-based design approach has been adopted in this research since it can be used effectively for designing, developing and prototyping of controllers. Accordingly, the scope of this research falls under 3 categories: modeling, control design and control evaluation.

1.5.1 Modeling

As a first step in model-based design, two types of system models have been designed: longitudinal dynamics plant model and control-oriented model.

Longitudinal dynamics plant model

A longitudinal dynamics model of an electric vehicle, with central drive powertrain architecture, has been developed in MapleSim, a software based on graph-theoretic modeling [12]. The model has been exported to Matlab/Simulink as an optimized C code or S-function to support development of controllers. Specifically, this high-fidelity model of a Rav4EV has been utilized to support the following tasks:

- Parameter identification of the longitudinal dynamics model of the EV.
- Validation of control-oriented model.
- Testing the anti-jerk traction and cruise controllers through model-in-the-loop simulation.
- Testing and evaluation of the anti-jerk controllers through hardware-in-the-loop tests.

Control-oriented model

Since we use a model-predictive approach for control design, the control-oriented model is built into the structure of the model predictive controller (MPC). The control-oriented model is required to be accurate enough to represent the plant dynamics with reasonable accuracy, and fast enough to enable real-time implementation. The novel control-oriented model discussed in Section 3.2.2 enables wheel-slip control in addition to conventionally used halfshaft torque control models.

1.5.2 Anti-Jerk Control Design

An MPC approach is used to design the controllers, since it has the ability to deal with multiple objectives of tracking wheel-slip, velocity and anti-jerk in an optimal sense, while handling constraints at the same time. The anti-jerk traction controller is designed to track a reference wheel-slip, while maintaining jerk-free traction in response to a driver pedal input. A novel anti-jerk cruise control is designed to meet the multiple objectives of velocity tracking and anti-jerk on any road surface. The system is subject to constraints that include the maximum motor torque, rate of torque and maximum wheel-slip. In this research, the following tasks will be undertaken to achieve the control design:

- Create a control-oriented model inherent to the design of the MPC controller.
- Design an MPC-based anti-jerk traction controller.
- Design an MPC-based anti-jerk cruise controller.

1.5.3 Control Evaluation

In this research, the following process [26] will be used to validate the performance of the anti-jerk traction and cruise controllers:

- Model-in-the-loop (MIL) simulation to evaluate the performance of the MPC controllers.
- Comparison of the developed MPC controllers to other controllers presented in the literature.
- Hardware-in-the-loop (HIL) testing to evaluate the real-time implementation capability of the controllers.

1.6 Thesis Organization

The introduction to this thesis includes a brief historical background and motivation to design anti-jerk controllers for EVs. A comparison has been made with conventional ICE vehicles to bring out the significance of driveline oscillations in EVs. A problem statement indicating

the scope of this research, categorized under modeling, control design, and evaluation, has been explained.

Chapter 2 presents the dynamic modeling, parameter identification and validation of the longitudinal dynamics plant model. The longitudinal dynamics plant model is developed using standard components from the multibody library. An experimental set up for simultaneous data collection from an array of sensors like Vehicle Measurement System (VMS), Global Position Sensor (GPS), Inertial Measurement Unit (IMU) and vehicle Controller Area Network (CAN) using a CAN integration device has been discussed. The parameters for the longitudinal dynamics model were identified experimentally based on a two-stage process. In stage 1, component level parameters were identified based on sub-system models. Thereafter, the parameters identified during stage 1 were optimized using the full vehicle longitudinal dynamics model in Stage 2. The longitudinal dynamics model has been validated by comparing against experimental data measured on the vehicle.

Chapter 3 presents the design of control-oriented models and the model-predictive controller. Two different control-oriented models, one based on only halfshaft torsion control and the other on halfshaft torsion and wheel slip control, have been formulated. The parameters of the control-oriented model have been validated using experimental data. The advantages of using halfshaft torsion and wheel slip based models in the design of MPC-based anti-jerk traction and cruise controllers has been discussed. A criterion for performance assessment of controllers through MIL and HIL experiments is also discussed.

Chapter 4 presents the design of the MPC-based anti-jerk traction controller to satisfy the multiple objectives of slip-tracking performance and enhanced drivability at the same time. This controller has been compared against other controllers in the literature, all tuned for anti-jerk performance. The performance was assessed by comparing the test results of slip-tracking, jerk and integral action of jerk in 5 different scenarios. The real-time implementation has been demonstrated through HIL testing.

Chapter 5 presents the design of an anti-jerk cruise controller to satisfy the multiple objectives of velocity tracking and enhanced drivability, while adapting to changes in road conditions. The significance of incorporating wheel-slip constraints in addition to halfshaft torsion in the control design to enhance anti-jerk performance has been demonstrated. The real-time implementation has been demonstrated through HIL testing.

Conclusions, summary of contributions and future work is discussed in Chapter 6.

Chapter 2

Vehicle Dynamic Modeling and Validation

2.1 Introduction and Literature Review

Advances in vehicle control systems and reduced development time have necessitated quick and accurate development of vehicle models. A significant savings in development time are possible, if an accurate model is available. The accuracy of these models depends on the parameters used, which are identified through parameter identification techniques.

In this Chapter, an 18 degree of freedom (DOF) high-fidelity longitudinal dynamics plant model has been developed for the Rav4EV using MapleSim, a software based on graph-theoretic modeling [27]. MapleSim uses optimized code-generation techniques to build models capable of real-time simulation.

In general, vehicle parameters are identified through standard tests, undertaken in dedicated test facilities such as [28]. Standard tests such as null point method, weight balance method and pendulum tests [29] have been developed for measuring center of gravity of a vehicle. Tire parameters are identified as per ASTM J199801, which involves tire testing on a belt-type flat surface machine. However, the tire behaviour during these standard laboratory tests is not as good as on-road tests [30] due to differences in texture of road surface, ambient temperature, road crown and road alignment. Vehicle manufacturers are researching methods [31] to get test bench test data as close as possible to that obtained during on-road tests. Moreover, standardized test methods are expensive [32] and time-consuming.

This chapter explains how on-road performance measurement systems and numerical

parameter identification methods can substantially reduce the need for standardized testing on test beds for vehicles. Experimental data was gathered on our test vehicle, Toyota Rav4EV, using an integrated measurement system designed to simultaneously collect data by integrating an array of sensors including a Vehicle Measurement System (VMS), Global Position Sensor (GPS), Inertial Measurement Unit (IMU), and the vehicle Controller Area Network (CAN).

A two-stage parameter optimization procedure is proposed to identify the parameters of the longitudinal dynamics plant model. First, component level or sub-system models were considered to identify the individual parameters of the plant model. The parameters identified using this approach were then optimized together using the complete plant model developed in MapleSim to obtain the final set of parameters. A MATLAB/Simulink based non-linear least square parameter estimator with a trust region reflective algorithm [33] was used to identify the parameters by minimizing the difference between experimental and simulated data.

The MapleSim longitudinal dynamics model is presented in Section 2.2. Details of the experimental measurement system and vehicle tests are described in Section 2.3. The details of component level parameter identification are discussed in Section 2.4 and vehicle level parameter identification and model validation are presented in Section 2.5.

2.2 Plant Model for Longitudinal Dynamics

In this research, MapleSim is used to create a longitudinal dynamics model of the Rav4EV, since it has the ability to generate highly optimized simulation code useful for real-time implementation. MapleSim provides a library of standard components such as rigid bodies, springs, dampers, joints and tires which can be combined to create highly customized models for a user-application. The developed models are acausal, which means that they are adaptable to the direction of data flow unlike traditional input/output blocks.

A full-car chassis model of the front wheel drive Rav4EV with 18 DOF has been developed in MapleSim. The chassis has 6 DOF, wheels 4 DOF, suspension 4 DOF and halfshaft 4 DOF. The tires have been modeled using Pacejka's Magic formula relaxation length model so that the effect of wheel slip transients are included. The model consists of a rigid body representing the car mass, connected to both front and rear tires through suspensions. Both suspensions are simplistically modeled as linear spring dampers which allow only vertical displacement. The motor torque is amplified in a single-reduction gearbox

and applied to the forward wheels through a halfshaft. The halfshaft is modeled as a torsional spring and damper system of negligible mass (being small). The halfshaft is connected on both the wheel and the gearbox sides with universal joints with an additional prismatic joint on the wheel side to cater for any vertical movement of the front wheel. A representation of the full-car longitudinal dynamics model in MapleSim is shown in Fig. 2.1.

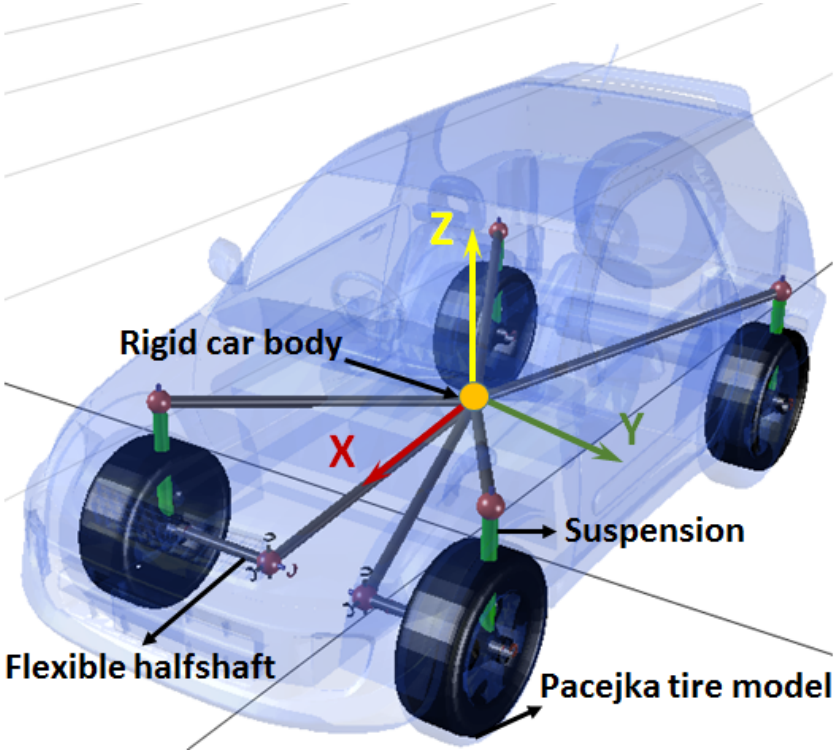

Figure 2.1: Full-car longitudinal dynamics model of Rav4EV in MapleSim

The various model components are explained in detail as follows:

(a) **Vehicle chassis**. In the MapleSim model, the vehicle chassis is modeled as rigid body with mass, CG and inertia parameters specified for the rigid body. The rigid car body is connected to the front and rear suspensions through a rigid body frame that defines the position and orientation relative to the center of gravity.

(b) **Suspension components**. Suspension elements connect the car mass to the front and rear wheels. The suspension elements are modeled using prismatic joints that exhibit linear stiffness and damping characteristics. The prismatic joint allows relative translational motion between the wheel and vehicle chassis only in the vertical direction. The prismatic joint is assumed to have an initial displacement due to the weight of

chassis acting on the spring. Each prismatic joint is connected to the wheel through a revolute joint, thus allowing the wheel one rotational degree of freedom. The stiffness and damping parameters of this revolute joint are assumed to be zero to simulate an ideal joint.

(c) **Tire components**. Each tire consists of two sub-components; a standard tire body and a tire model. The standard tire component computes the kinematic parameters such as slip angle, slip ratio and dynamic tire radius based on vertical tire stiffness and load changes on application of a normal load on the tire. The parameters of the standard tire component are tire mass, tire inertia, vertical stiffness, damping and unloaded tire radius. MapleSim has a library of different tire models (Linear, Fiala, Calspan, Pacejka, and user-defined) to calculate the forces and moments acting at the contact patch. In this research, we have used a user-defined tire model. The longitudinal force to be input in the tire model are computed by developing custom components based on the Pacejka tire model as given in equation (2.4) while the normal force is computed directly from the standard tire component.

(d) **Half shaft components**. Half shaft is modeled as a torsional spring and damper system. The halfshaft has universal joints at each end. One end of the halfshaft is connected to the powertrain while the other end is connected to the wheel. An ideal prismatic joint is also added between the halfshaft and wheel to allow for any lateral displacement of the wheel.

The model accepts motor torque (T_m) as an input and outputs halfshaft (wheel) torque (T_{hs}), vehicle speed (v), and wheel speed ($Wh_{sp} = r_{eff}\omega_w$). The model was exported as an optimized S-function so that it could be used for simulation and control in Matlab/Simulink.

2.3 Experimental Testing

2.3.1 Experimental Measurement System

Experimental data was recorded by integrating an array of sensors that includes a Vehicle Measurement System (VMS), Global Position Sensor (GPS), Inertial Measurement Unit (IMU), and the vehicle Controller Area Network (CAN). A CAN integration device from 'Vector Informatik GmbH' was used to integrate the various signals. The system architecture

for integrating the signals from the various devices is shown in Fig. 2.2. A brief description of the various sensors is as follows:

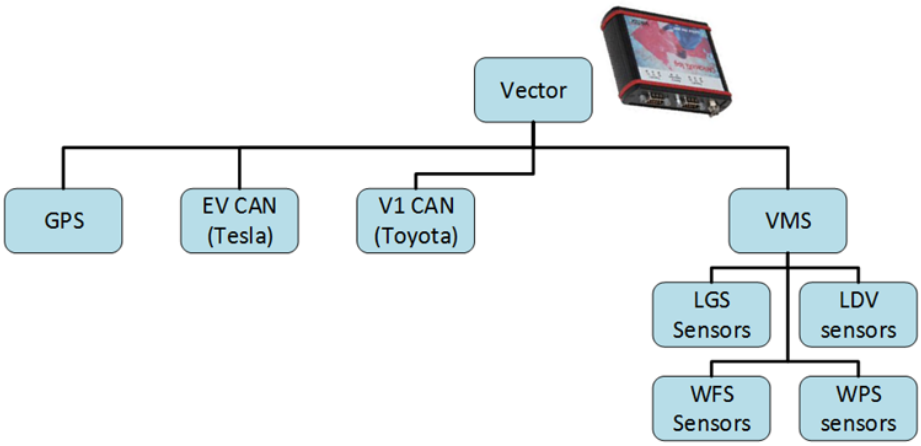

Figure 2.2: Integration of sensors through Vector CAN box

Vehicle Measurement System (VMS). The VMS manufactured by '*AnD Technologies*' consists of a sensor set mounted on each of the four wheels to record signals during on-road testing. Each VMS sensor set mounted on a wheel consists of the following five sensors:

(a) **Wheel Force Sensor (WFS) and Rotary encoder.** The WFS is a strain gauge based sensor mounted on a custom wheel hub to record forces (F_x, F_y, F_z) and moments (M_x, M_y, M_z) on the wheel. The wheel hub also houses a rotary encoder to measure the angular speed of wheel.

(b) **Wheel Position Sensor (WPS).** The WPS consists of five digital encoders mounted on a frame that connects the wheel to the chassis. The sensor measures the relative displacement and rotation of the wheel about the longitudinal, lateral and vertical axis with respect to the vehicle body. A picture of WFS and WPS sensors installed on the vehicle is shown in Fig. 2.3.

(c) **Laser Ground Sensors (LGS) and Laser Doppler Velocimeters (LDV).** The LGS and LDV sensors are mounted together as a single unit on the wheel hub as shown in Fig. 2.4. The LGS comprises three laser sensors that compute the dynamic radius of the wheel, or distance of the wheel center from the ground. The two LDV sensors are used to compute the ground speed of the tire in longitudinal and lateral directions. In addition to measuring the dynamic radius, the LGS and LDV sensors are interfaced

(a) WFS sensor

(b) WPS sensor

Figure 2.3: WFS and WPS sensors mounted on wheel hub

to compute the slip angle, camber angle, pitch angle and roll angle of the tire by processing the measured data from the sensor set.

Global Position Sensor (GPS) and Inertial Measurement Unit (IMU). An integrated GPS and IMU sensor unit manufactured by Racelogic was used to measure the pitch, roll and yaw rate of the vehicle using rate gyroscopes, and longitudinal, lateral and vertical accelerations using accelerometers. The IMU sensor (Model RLVBIMU04) has an accuracy of yaw rate resolution of $0.014^0/s$ and acceleration resolution of $0.15\ mg$. The GPS and IMU sensors are co-located on the roof of the car with the help of a magnetic base as shown in Fig. 2.5.

Vehicle Data using Controlled Area Network (CAN bus). A number of signals were also obtained by tapping into the vehicle CAN through the onboard data port of the Rav4EV. Since the vehicle is powered by the Tesla powertrain, the vehicle CAN network is an integration of V1-CAN from Toyota and EV-CAN from Tesla Motors. The V1-CAN consists of signals related to the vehicle such as accelerator pedal position and vehicle speed while EV-CAN

Figure 2.4: LGS and LDV sensor set mounted on wheel hub

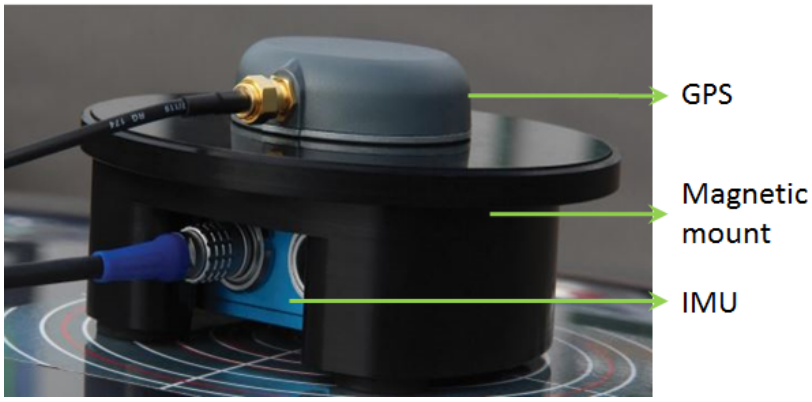

Figure 2.5: GPS and IMU sensor set co-located with the magnetic mount

consists of powertrain related signals such as motor speed, motor torque, DC voltage and battery state-of-charge (SOC).

2.3.2 Vehicle Tests

Experimental testing of the Toyota Rav4EV was conducted on the Toyota Motor Manufacturing Canada (TMMC) test tracks. Fig. 2.6a shows the Rav4EV equipped up with the VMS. The Rav4EV was tested at TMMC test track as shown in Fig. 2.6b. Several straight

line driving maneuvers, which included hard acceleration and hard braking, acceleration cruise and braking, coast down, and driving over a speed bump, were undertaken to excite the longitudinal dynamics of the vehicle. The VMS system was calibrated prior to testing. While few parameters such as wheel base, front and rear track widths are obtained directly from physical measurements and weight of the Rav4EV on each of the four wheels identified through weight scales at TMMC, all other model parameters were estimated based on track tests with the vehicle.

(a) Rav4EV instrumented with measurement system

(b) TMMC test track

Figure 2.6: Rav4EV rigged with vehicle measurement system on TMMC test track

During coast down tests, wind speed was also recorded using a weather station. A brief summary of the road tests undertaken and parameters identified are given in Table 2.1.

Test type	**Vehicle Speed range**	**Parameters Identified**
Speed-bump test	Constant 15 Km/hr	Suspension parameters
Hard acceleration tests	0-100 Km/hr	Tire model parameters, Motor inertia, Wheel inertia, Motor model parameters
Acceleration and cruise tests	0-80 Km/hr 0-100 Km/hr	Position of center of gravity, Vehicle pitch inertia
Coast-down tests	0-70 Km/hr	Coefficient of drag and rolling resistance

Table 2.1: Experimental tests conducted

2.4 Parameter Identification - Component Level

On-road testing was found to be a quick and reliable methodology for parameter identification. A two-stage optimization process was followed for to identify the paramaters of the longitudinal dynamics model. In this section, different component level sub-system models were considered to identify the individual parameters of the longitudinal plant model. The parameters identified from component level testing was used as an initial guess for vehicle level estimation discussed in section 2.5. The data collected from different maneuvers has been processed to estimate the required parameters.

For component level parameter identification, we consider a 3 DOF (longitudinal, pitching and heave) longitudinal dynamics model of the Rav4EV. In this model, we assume that the mass is lumped at its center of gravity (CG), and there is no relative motion between the wheels and chassis. The forces acting on the Rav4EV during longitudinal motion are shown in Fig. 2.7. The dynamic equation for the longitudinal motion of the vehicle is expressed as:

$$M\dot{v} = F_x - F_r - F_d \tag{2.1}$$

where M represents the total mass of the vehicle, $F_x = 2F_{xf}$ is the sum of longitudinal traction/braking forces acting on the each of the front wheels ($F_{xr} = 0$ for a front wheel

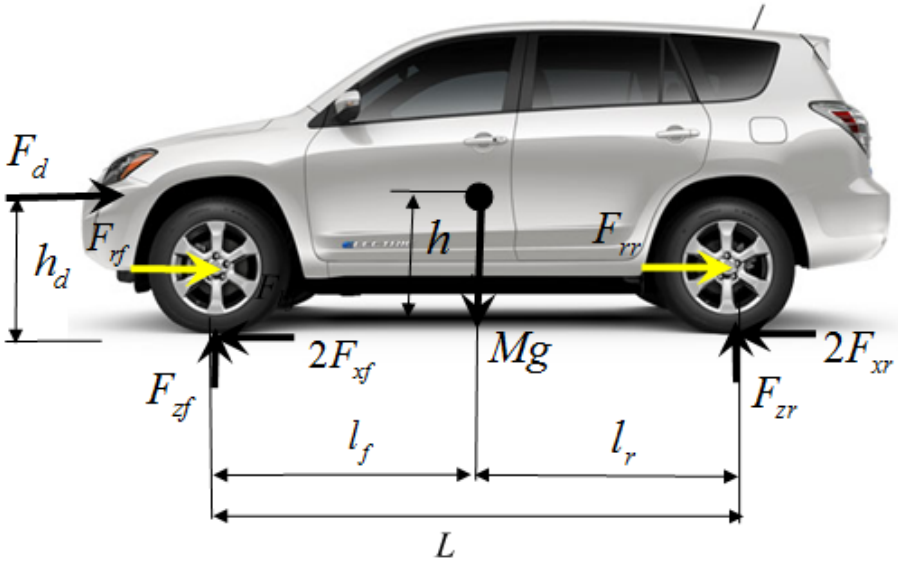

Figure 2.7: Longitudinal forces acting on Rav4EV

driven vehicle), $F_r = F_{rf} + F_{rr}$ is the rolling resistance force on the front and rear wheels and F_d is the aerodynamic drag force acting at the center of pressure h_d. F_{zf} is the sum of normal forces on the front wheels and F_{zr} is the sum of normal forces on the rear wheels and L is the wheel base indicating the distance between the front and rear wheels. The longitudinal distance of CG from front and rear wheels are l_f and l_r respectively, while the vertical distance of CG from ground is represented by h.

Through the parameter identification procedure, we aim to accurately estimate the traction force (F_x), resistance forces (F_r and F_d), and torque transmitted through the powertrain of the Rav4EV. The traction force on the Rav4EV is governed by road-load friction characteristics as defined by the Pacejka tire model. The friction characteristics are further influenced by normal load distribution on the wheels, which in turn is governed by suspension dynamics. The resistance forces are estimated from the coefficient of drag and rolling resistance identified through coast down tests. The inertial characteristics of the drivetrain (motor, gearbox and drivetrain) and stiffness/damping characteristics of the halfshaft, govern the torque (and torsional oscillation) characteristics of the powertrain system. The procedure followed for parameter identification is explained in the following sections.

2.4.1 Tire Model Parameters

The longitudinal traction force generated at each tire is a function of longitudinal slip ratio and the normal force applied on the tires. For a front wheel drive vehicle, the traction force on the two front wheels is represented as:

$$F_x = 2F_{xf} = 2\mu(\lambda)F_{zf} \tag{2.2}$$

where μ is the normalized longitudinal force governed by the Pacejka tire model [34] which is a function of the slip ratio λ. The slip ratio for an accelerating wheel is defined as:

$$\lambda = \frac{r_{eff}\omega_w - v}{r_{eff}\omega_w} \tag{2.3}$$

where ω_w is the angular velocity of the wheel, r_{eff} is the dynamic radius of the wheel and v is the vehicle speed. The relationship between μ and λ is represented by Pacejka's Magic formula tire model as:

$$\mu = D\sin\left[C\tan^{-1}\left\{B\lambda - E\left(B\lambda - \tan^{-1}B\lambda\right)\right\}\right] \tag{2.4}$$

The parameters of Pacejka's Magic formula tire model in equation (2.4) are determined by plotting μ with the longitudinal slip ratio λ. Using a non-linear curve fit, the parameters of the Magic formula tire model as seen in Fig.2.8 have been estimated as B=49, C=1.37, D=1.25 and E=0.01. The model parameters B, C, D and E in equation (2.4) are:
B: Stiffness factor. It determines the slope at the origin
C: Shape factor. It determines the range of the sine function
D: Peak value. It represents the peak value of the tire force.
BCD: The product of BCD represents the longitudinal stiffness of the tire.
E: Curvature factor. It is used to represent the curvature near the peak of the curve.

2.4.2 Pitch Inertia and Position of Center of gravity

The pitch inertia (I_y), position of longitudinal center of gravity (CG) (l_r or l_f) and height of CG (h) are identified by exciting the pitching motion of the vehicle through rapid acceleration and hard braking tests. The data gathered on front and rear tires during acceleration/braking maneuvers is processed to obtain these parameters. The equation governing the pitch motion

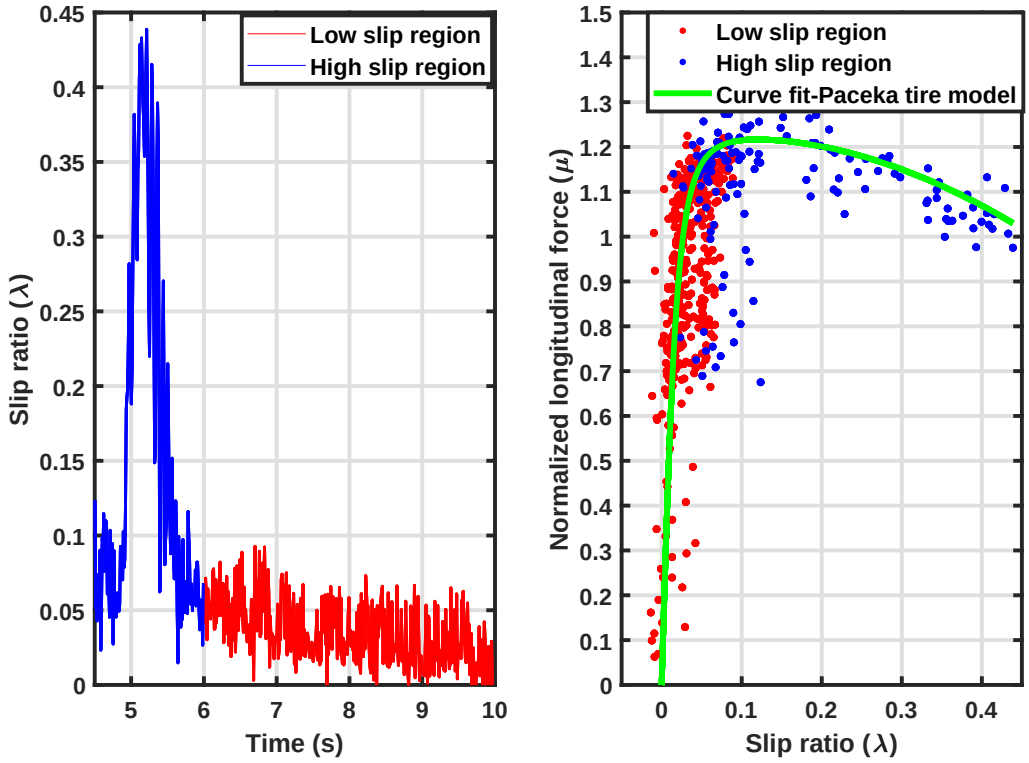

Figure 2.8: (a) Slip ratio λ vs time (b) Normalized longitudinal force μ vs λ

of the vehicle is expressed as:

$$I_y\ddot{\theta} = -F_{zf}l_f + F_{zr}l_r - M\dot{v}h \tag{2.5}$$

where, I_y is the pitch inertia of the vehicle and $\dot{v}$ is the acceleration at the center of gravity of the vehicle. Rearranging the terms to obtain normal force on the front wheels:

$$F_{zf} = \frac{Mgl_r}{L} - \frac{Mh\dot{v}}{L} - \frac{I_y\ddot{\theta}}{L} \tag{2.6}$$

Since the GPS is not positioned at the CG of the vehicle but at the roof of the car, the acceleration recorded by the GPS needs to be transformed to the CG of the vehicle. Neglecting the small centripetal terms, the longitudinal vehicle acceleration at CG is expressed as:

$$\dot{v} = \dot{v}_{GPS} + \ddot{\theta}(h - H) \tag{2.7}$$

where H is the height at which GPS is mounted from the ground.

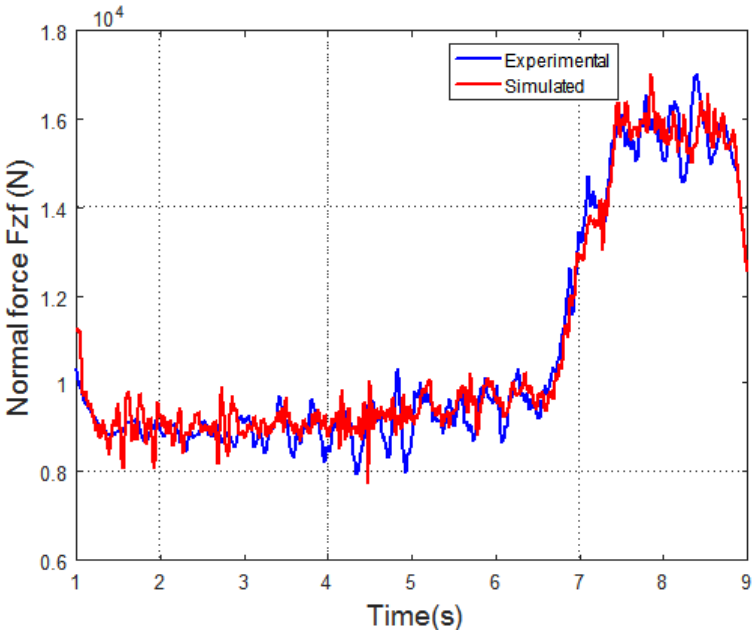

Figure 2.9: Experimental vs estimated normal force

Substituting equation (2.7) into (2.6), we have:

$$F_{zf} = \frac{Mgl_r}{L} - \frac{Mh(\dot{v}_{GPS} + \ddot{\theta}(h - H))}{L} - \frac{I_y\ddot{\theta}}{L} \tag{2.8}$$

where the vehicle acceleration in longitudinal direction ($\dot{v}_{GPS}$) is measured using the GPS, and pitch acceleration ($\ddot{\theta}$) measured by taking a derivative of the pitch rate measured by the IMU.

By minimizing the difference between the experimental and simulated normal force on the front wheels (F_{zf}), as shown in Fig 2.9, the parameters have been estimated as $I_y = 3052\ kgm^2$, $l_r = 1.42\ m$ and $h = 0.62\ m$.

2.4.3 Suspension Parameters

The front suspension of the Rav4EV is a Mcpherson strut while the rear suspension is of double wishbone type. To include load transfer effects due to pitching motion of the vehicle during acceleration and braking maneuvers, we have considered a simplistic approach of modeling the front and rear suspensions as linear spring and damper elements and assumed that the suspension elements are symmetric about the longitudinal axis for left/right wheels. We consider a four-DOF half car model as shown in Fig. 2.10, with the aim to estimate the

stiffness parameters (K_f, K_r) and damping parameters (C_f, C_r) of front and rear suspensions.

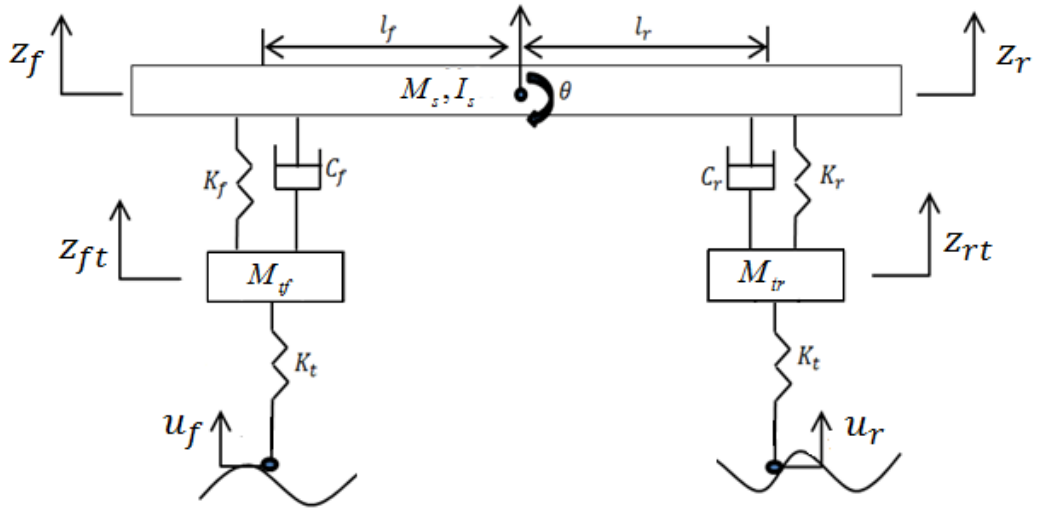

Figure 2.10: 4-DOF car model for identification of suspension parameters

The model allows vertical motion (z direction) and pitch motion (θ) of the car chassis (sprung mass) and vertical motions of the front and rear wheels (z_{ft}, z_{ft}). The sprung mass (M_s) is calculated by subtracting the half car mass from those of the two wheels (M_{tf}, M_{tr}) and the pitch inertia (I_s) of the sprung mass is calculated by subtracting half the vehicle inertia ($\frac{I_y}{2}$) of the from the inertia of the unsprung mass (I_u) about the CG as follows:

$$\begin{aligned} M_s &= \frac{M}{2} - (M_{tf} + M_{tr}) \\ I_s &= I_y - I_u \end{aligned} \tag{2.9}$$

where $I_y = F_{zf} l_f^2 + F_{zr} l_r^2$ and $I_u = M_{tf} l_f^2 + M_{tr} l_r^2$. The tires are modelled as linear springs with stiffness K_t. Since the damping of the tires is small as compared to that of the suspension elements, it is neglected. The equations of motion for M_s and I_s for a vehicle driven at a constant speed v are:

$$\begin{aligned} M_s \ddot{z} &= -\overbrace{K_f \left\{z_f - z_{tf}\right\} - C_f \left\{\dot{z}_f - \dot{z}_{tf}\right\}}^{F_{zf}} - \overbrace{K_r \left\{z_r - z_{tr}\right\} - C_r \left\{\dot{z}_r - \dot{z}_{tr}\right\}}^{F_{zr}} \\ I_s \ddot{\theta} &= l_f \left[K_f \left\{z_f - z_{tf}\right\} + C_f \left\{\dot{z}_f - \dot{z}_{tf}\right\}\right] - l_r \left[K_r \left\{z_r - z_{tr}\right\} + C_r \left\{\dot{z}_r - \dot{z}_{tr}\right\}\right] \end{aligned} \tag{2.10}$$

The equations of motion for unsprung masses (M_{tf}) and (M_{tr}) consisting of the wheel and

suspension mass can be written as:

$$\begin{aligned} M_{tf}\ddot{z}_{tf} &= K_f\left\{z_f - z_{tf}\right\} + C_f\left\{\dot{z}_f - \dot{z}_{tf}\right\} - K_t(z_{tf} - u_f) = 0 \\ M_{tr}\ddot{z}_{tr} &= K_r\left\{z_r - z_{tr}\right\} + C_f\left\{\dot{z}_r - \dot{z}_{tr}\right\} - K_t(z_{tr} - u_r) = 0 \\ z_f &= z + l_f\cos\theta \qquad \text{and} \qquad z_r = z - l_r\cos\theta \end{aligned} \tag{2.11}$$

The input to the model is a speed bump of a known profile has been used to excite pitch and heave motions. The road profile input u_i to the model has been considered to be a modified cosine function [35, 36] expressed as:

$$u_i = \frac{A}{2}\left[\begin{cases} 0 & t < \frac{D_i}{v} \\ 1 - \cos(\frac{2\pi}{\lambda_{bump}}D_i) & \frac{D_i}{v} \le t \le \frac{D_i+\lambda_{bump}}{v} \\ 0 & t > \frac{D_i+\lambda_{bump}}{v} \end{cases}\right] \tag{2.12}$$

where $A = 7.2\ cm$ is the height of the speed bump, $\lambda_{bump} = 49.5\ cm$ is the width of the speed bump and D_i is the distance taken by the front or rear tire to travel to the speed bump and $i = f, r$ refers to the front and rear tire respectively. The speed profile given in (2.12) is considered appropriate for approximating the suspension forces as compared to the standard sine function given by:

$$u_i = A\sin\left(\frac{\pi}{\lambda_{bump}}\right) \tag{2.13}$$

A plot comparing the speed profile approximated using the cosine function in (2.12) and the sine function (2.13) is given in Fig.2.11.

The suspension parameters are determined by minimizing the cost function J expressed as:

$$J = \int_0^t \left[w_1 {J_1}^2 + w_2 {J_2}^2 + w_3 {J_3}^2\right] dt \tag{2.14}$$

where J_1, J_2, and J_3 are the costs with respect to normal forces (F_{zf} and F_{zr}) on the front and rear wheels and pitch rate ($\dot{\theta}$) of the unsprung mass, expressed as:

$$\begin{aligned} J_1 &= F_{zf}^* - \overbrace{(M_{sf} + M_{tf})g + K_t(z_{tf} - u_f)}^{F_{zf}} \\ J_2 &= F_{zr}^* - \overbrace{(M_{sr} + m_{tr})g + K_t(z_{tr} - u_r)}^{F_{zr}} \\ J_3 &= \dot{\theta}^* - \dot{\theta} \end{aligned} \tag{2.15}$$

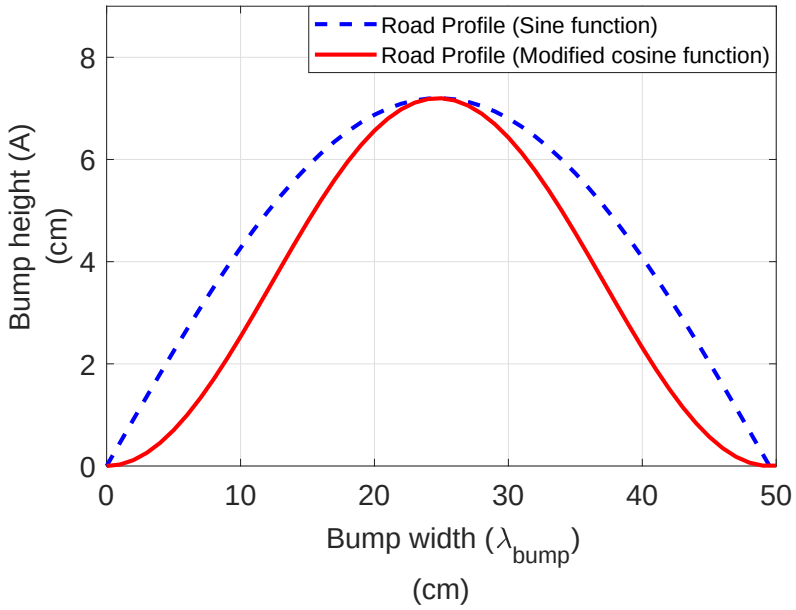

Figure 2.11: Profile of the speed bump

where F^*_{zf}, F^*_{zr} and $\dot{\theta}^*$ are the experimentally measured normal force on the front wheel, normal force on the rear wheel and pitch rate, and w_1, w_2, and w_3 are the corresponding weights. A comparison of the experimental and simulated suspension forces is given in Fig. 2.12.

Using the non-linear least square algorithm, the suspension parameters have been identified as $K_f = 54,370\ N/m$, $K_r = 35,540\ N/m$, $C_f = 1980\ Ns/m$ and $C_r = 1795\ Ns/m$. The vertical tire stiffness (K_t) is assumed to be $253,000\ N/m$ as given in the manufacturer's specifications.

2.4.4 Resistance force Parameters

The forces resisting the longitudinal motion of the vehicle are the aerodynamic drag force and the rolling resistance force. Aerodynamic drag force (F_d) is the force due to the air resistance acting on the frontal area of the vehicle. This can be represented as:

$$F_d = \frac{1}{2}\rho C_d A_f v^2 \tag{2.16}$$

where ρ is the density of air, assumed as $1.2Kg/m^3$. The parameters estimated are the aerodynamic drag coefficient C_d and the frontal area of the vehicle A_f. The rolling resistance

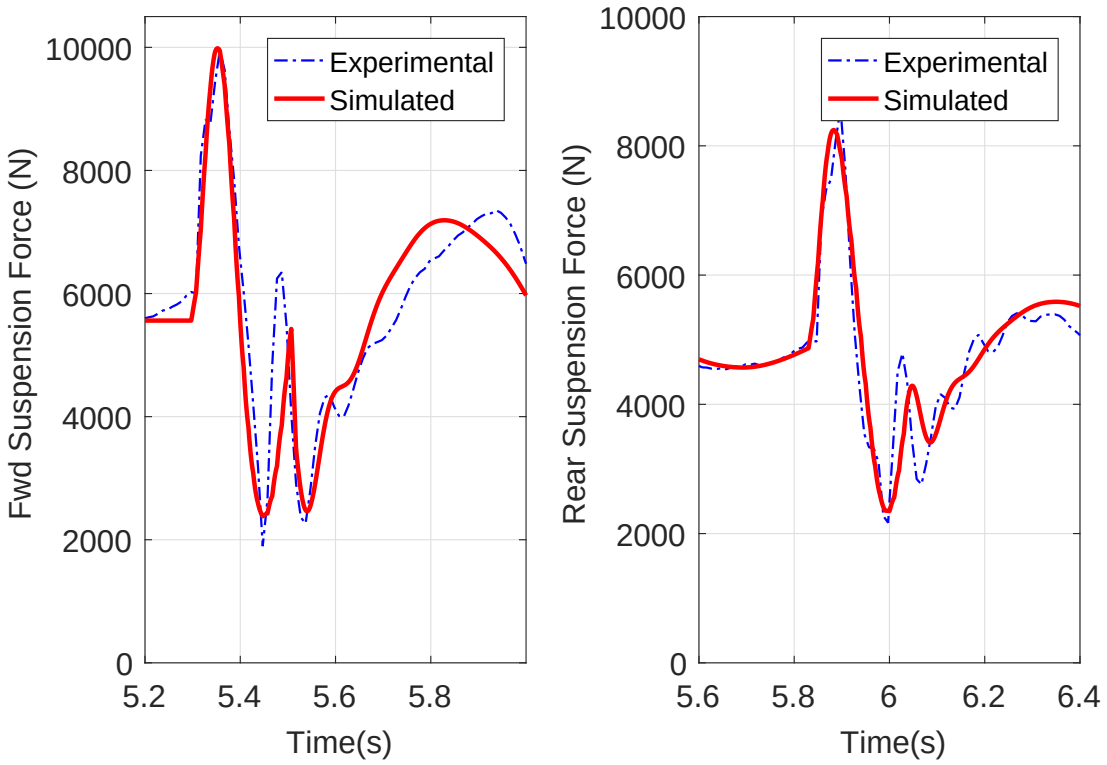

Figure 2.12: Normal force on the front left and rear left wheels

force is caused by a non-symmetric distribution of normal tire load over the contact patch. The rolling resistance force F_r at the tire center can be modeled as:

$$F_r = f_{rr}(F_{zf} + F_{zr}) \tag{2.17}$$

where f_{rr} is the rolling resistance coefficient to be estimated.

2.4.4.1 Frontal Area and Center of Pressure

Frontal area of a vehicle is the orthogonal projection of the vehicle on a plane perpendicular to the longitudinal axis of the vehicle. Frontal area is an important parameter that determines the aerodynamic drag, thus affecting the power consumption of the vehicle. In this research work, the frontal area (A_f) is estimated using an image processing technique [37]. The frontal image of the Rav4EV was processed to fill the car area with black color and background with white color. Then '**im2bw**' function in in the Image Processing Toolbox of MATLAB is used to convert the grey scale image to binary image. The grey scale image's darker pixels are replaced by 0 and lighter pixels are replaced by 1 in the final binary image as shown

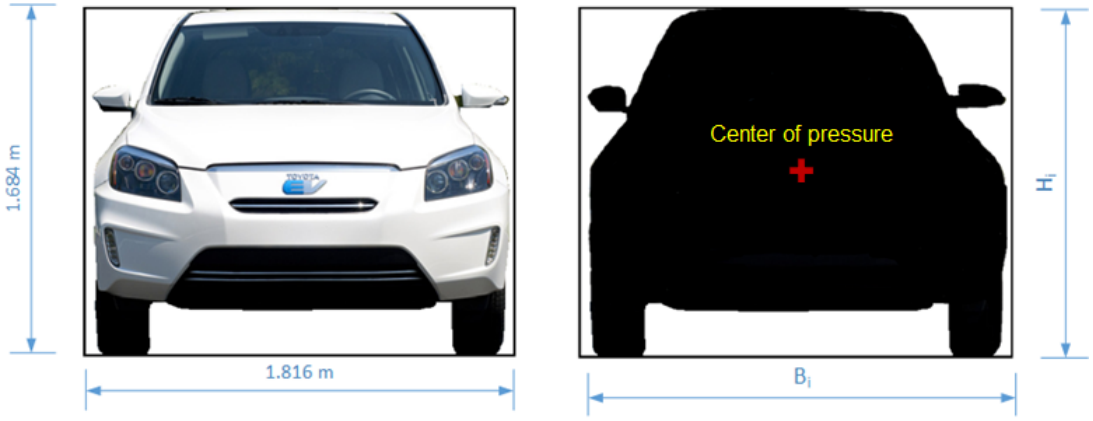

Figure 2.13: Frontal area of Rav4EV and processed image with black and white pixels

in Fig 2.13. Counting the pixels ($B_i \times H_i$) of the black and white color, the actual frontal area of Rav4EV has been calculated as 2.464 m^2 which is 80.6% of the total area of the box (3.058 m^2).

Further, assuming uniform air pressure distribution over the frontal area, the centroid of the figure could represent the center of pressure where the aerodynamic drag force acts. The height of center of pressure (h_d) has been estimated as 0.689 m.

2.4.4.2 Coefficient of Drag and Rolling Resistance

The aerodynamic drag coefficient (C_d) is determined from a coast-down test in accordance with the methodology followed by White and Krost [38]. The Rav4EV was accelerated to about 65 Km/hr and then the throttle released so that the vehicle slows down under the effects of aerodynamic drag and rolling resistance. The vehicle speed recorded with and against the wind can be seen in Fig. 2.14.

During coast-down, the traction force (F_x) on the vehicle is small and therefore neglected. The longitudinal dynamics equation (2.1) reduces to:

$$\begin{aligned} M\dot{v} &= \frac{1}{2}\rho C_d A_f v^2 + F_r \\ &or \\ -\frac{dv_x}{\frac{\rho C_d A_f v_x^2}{2M} + \frac{F_r}{M}} &= dt \end{aligned} \tag{2.18}$$

where F_r is assumed to be constant for speeds below 70 Km/hr [38]. Integrating equation (2.18), assuming an initial velocity of v_i, the coast down time t for vehicle speed to drop

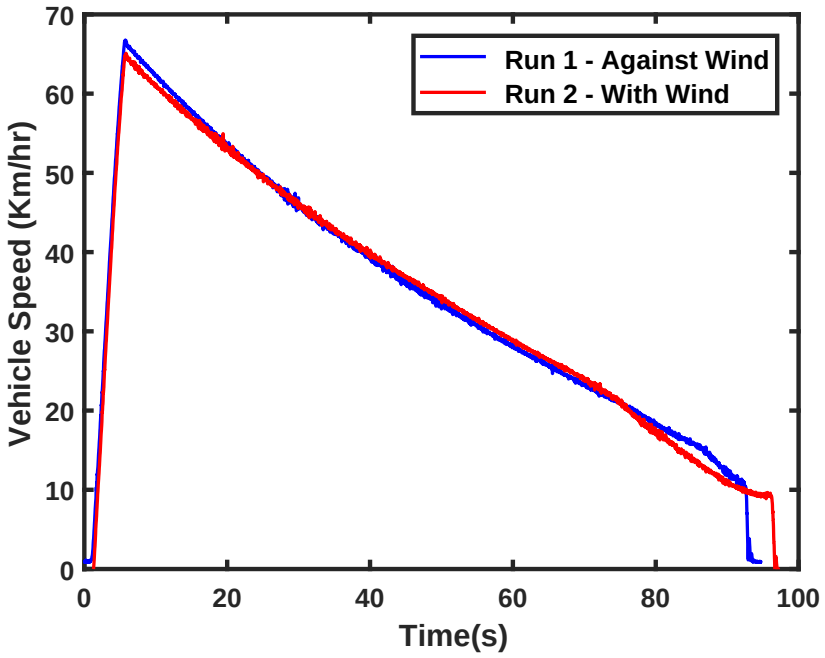

Figure 2.14: Vehicle speed recorded during coast down testing of Rav4EV

from v_i to v_x can be expressed as:

$$\frac{t}{T} = 1 - \frac{\tan^{-1}\left[v_x\left(\frac{\rho C_d A_f}{2F_r}\right)^{\frac{1}{2}}\right]}{\tan^{-1}\left[v_i\left(\frac{\rho C_d A_f}{2F_r}\right)^{\frac{1}{2}}\right]} \tag{2.19}$$

Solving for velocity v_x from equation (2.19) yields:

$$v_x = \left(\frac{2F_r}{\rho C_d A_f}\right)^{\frac{1}{2}} \tan\left\{\left(1 - \frac{t}{T}\right)\tan^{-1}\left[v_i\left(\frac{\rho C_d A_f}{2F_r}\right)^{\frac{1}{2}}\right]\right\} \tag{2.20}$$

Simplifying further,

$$\frac{v_x}{v_i} = \frac{1}{\beta}\tan\left[\left(1 - \frac{t}{T}\right)\tan^{-1}\beta\right] \tag{2.21}$$

where

$$\beta = v_i\left(\frac{\rho C_d A_f}{2F_r}\right)^{\frac{1}{2}} \tag{2.22}$$

The parameter β is estimated from (2.21) by plotting non-dimensional velocity ($\frac{v_x}{v_i}$) against non-dimensional time ($\frac{t}{T}$) as shown in Fig. 2.15.

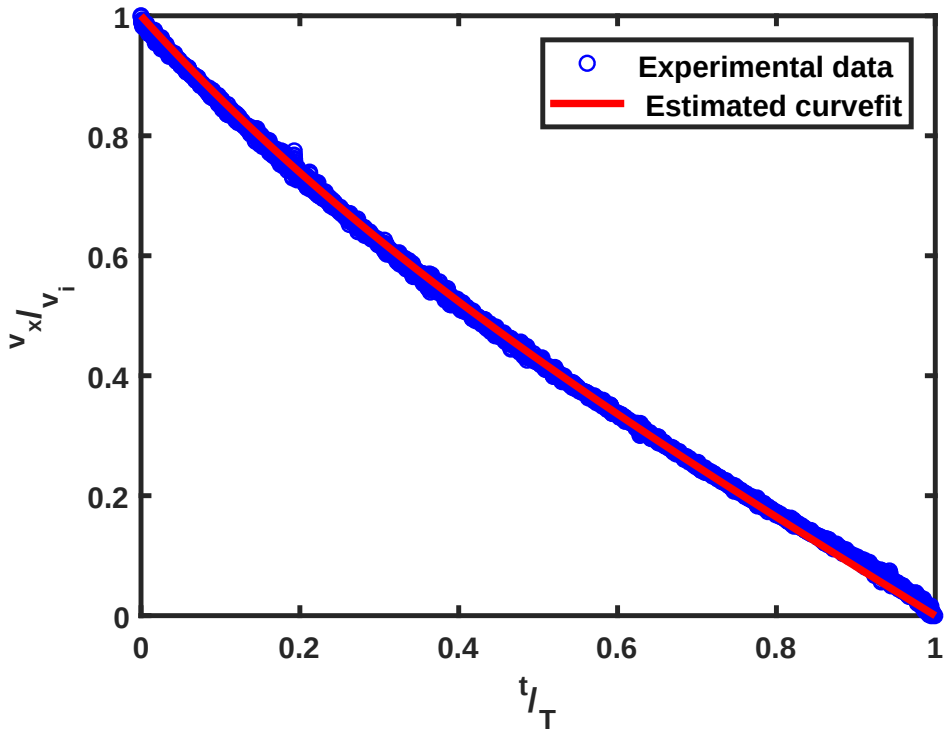

Figure 2.15: Plot of $\frac{v_x}{v_0}$ against $\frac{t}{T}$ to estimate β

The coefficient of drag C_d is obtained as:

$$C_d = \frac{2M\beta\tan^{-1}\beta}{v_i T \rho A_f} \tag{2.23}$$

and coefficient of rolling resistance f_{rr} is obtained as:

$$f_{rr} = \frac{v_i\tan^{-1}\beta}{\beta T g} \tag{2.24}$$

From data collected over 5 runs, the average coefficient of drag (C_d) was estimated as 0.308 which is close to 0.3 specified in the Toyota technical manual. The average coefficient of rolling resistance (f_{rr}) was estimated as 0.0015.

2.4.5 Drivetrain Inertia

The powertrain of the Rav4EV consists of a central motor driving the wheels through a single reduction gear. A differential connects the two front wheels on either side of the

motor. The dynamics of the motor and gearbox, and those of gearbox and halfshaft can be represented as:

$$J_m \dot{\omega}_m = \frac{T_m}{\eta_t} - \frac{T_t}{gr} \tag{2.25a}$$

$$J_t \dot{\omega}_t = T_t - T_w \tag{2.25b}$$

where J_m represents motor inertia, J_t transmission inertia, η_t efficiency of transmission, gr gear ratio of transmission and T_t transmission torque. Combining equations (2.25a) and (2.25b), the equation for drivetrain dynamics can be written in terms of measured motor torque (T_m) and wheel torque (T_w) as:

$$\underbrace{\left(J_m + \frac{J_t}{gr^2}\right)}_{J_d} \dot{\omega}_m = \frac{T_m}{\eta_t} - \frac{T_w}{gr} \tag{2.26}$$

which was used to simulate angular speed of motor (ω_m) for measured inputs T_m and T_w.

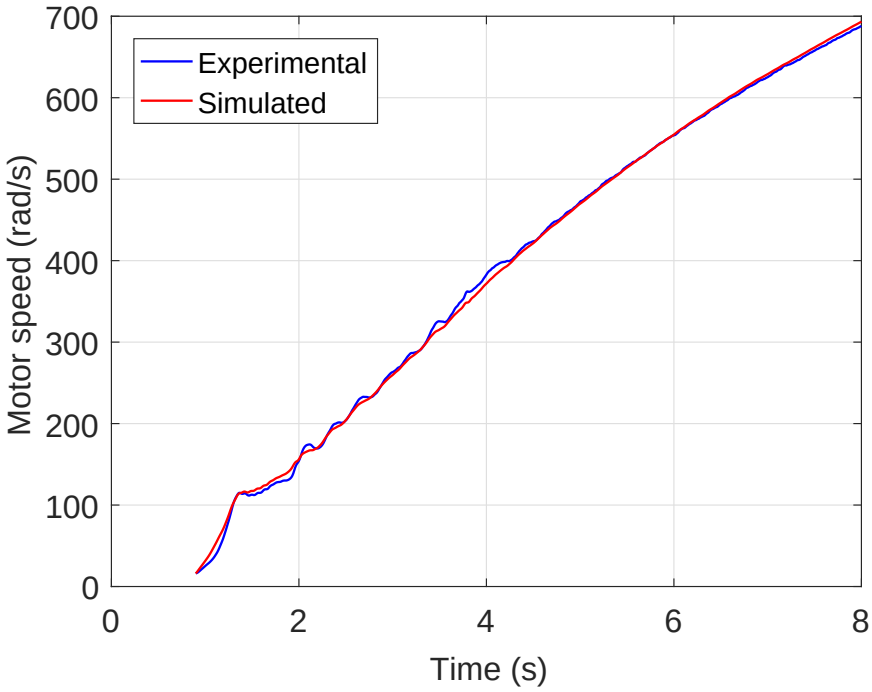

Figure 2.16: Experimental vs simulated motor angular speed (ω_m)

By comparing the experimental and simulated motor speed (ω_m) as shown in Fig. 2.16, the combined drivetrain inertia (J_d) has been estimated as 0.25 kgm^2 and drivetrain efficiency

as 94.6%.

2.4.6 Wheel Inertia

A free body diagram (FBD) of the forces acting on the driven wheel is shown in Fig. 2.17. The dynamic equation governing the driven wheel dynamics can be expressed as:

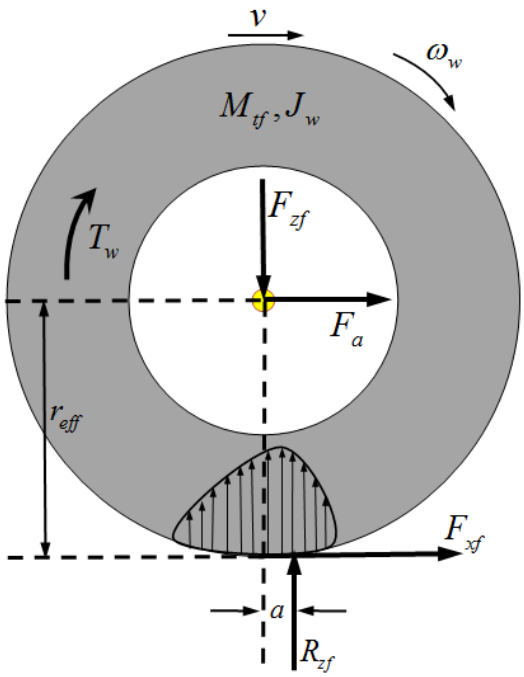

Figure 2.17: Free-body diagram showing forces on the front (driven) wheel

$$J_w \dot{\omega}_w = T_w - r_{eff} F_{xf} - M_{rr} \tag{2.27}$$

where $\dot{\omega}_w$ is the angular acceleration of wheel, r_{eff} is the measured dynamic radius of the wheel, and rolling resistance moment is $M_{rr} = aR_{zf} = f_{rr}F_{zf}$. The rolling resistance coefficient (f_{rr}) is estimated as 0.0015 (section 2.4.4.2) and mass of the wheel (M_{tf}) including rims with tire specification $225/65\ R17$ was measured as 22.7 kg. The relation between the longitudinal force (F_{xf}) at the ground and force F_a measured by the VMS at the centre of the wheel hub is expressed as:

$$F_{xf} = M_{tf}\dot{v} - F_a \tag{2.28}$$

By comparing the experimental and simulated wheel speed (ω_w) as shown in Fig. 2.18, the wheel inertia (J_w) has been estimated as 2.72 kgm^2.

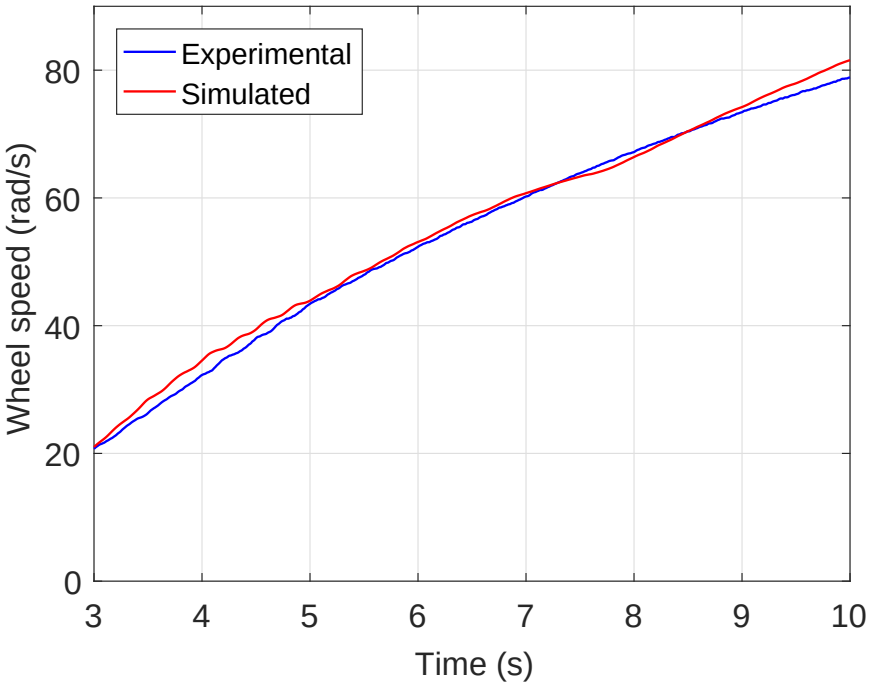

Figure 2.18: Experimental and simulated wheel spin (ω_w)

2.4.7 Halfshaft Parameters

Halfshafts are used to transmit the transmission torque from the gearbox to the wheel. The halfshaft is modeled as a torsional spring and damper system (1 DOF) negligible inertia. The stiffness of these shafts was derived from the physical and material properties of the halfshaft obtained from the manufacturer drawings. Torsional stiffness (torque required per unit twist) of the shaft is expressed as:

$$k = \frac{GJ_{hs}}{L_{hs}} \tag{2.29}$$

where G is the modulus of rigidity, J_{hs} is the polar moment of inertia and L_{hs} is the length of shaft. As the halfshaft has three different diameters over its length, the equivalent stiffness k is calculated by summing the longitudinal stiffness of each section. The torsional stiffness k of the halfshaft is calculated as 22555 Nm/rad. The damping parameter of the halfshaft c is chosen as 200 $Nm/(rad/s)$ as in [39], which is typical for halfshafts of similar length and diameters.

2.5 Parameter Identification - Vehicle Level and Model Validation

The parameters identified through component level identification procedures were used as an initial guess for vehicle level estimation. Vehicle level estimation can also be considered as model validation, since it involved comparison of model outputs to experimental data during on-road testing.

A motor torque required to achieve a constant vehicle speed of 100 Km/hr was used as reference input to validate the plant model. Fig. 2.19 shows the motor torque measured experimentally on the vehicle.

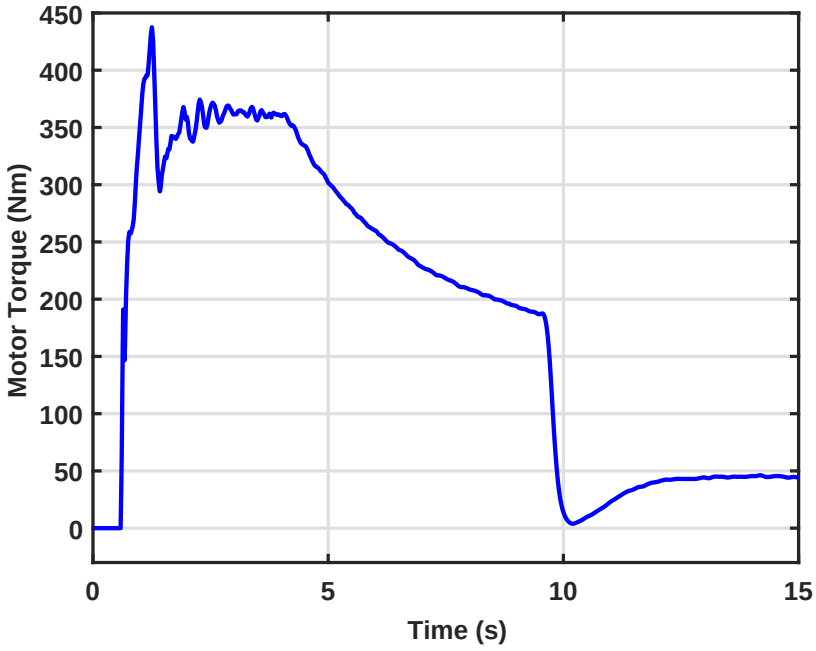

Figure 2.19: Motor torque input measured experimentally

A comparison of the plant outputs with the experimental data can be seen in Fig. 2.20. A good co-relation can be seen in the wheel torque, wheel and vehicle speeds. Small differences between experiments and models can be attributed to the following:

(a) Differences in effective tire radius. The effective tire radius in the plant model is calculated by the custom tire component based on dynamic changes in load as discussed in section 2.2 while the control-oriented model assumes a constant tire radius.

(b) Small differences in parameters (B, C, D and E) identified for the Pacejka tire model (2.4) and the relaxation length model (3.9).

A comparison has been drawn between the acceleration recorded experimentally by the IMU and that simulated by the plant model. Since the acceleration data recorded by the IMU was noisy, a second order Savitzky-Golay filter [40] was used to filter the noise. In the absence of a sensor which can directly measure jerk, both experimental and model jerks were calculated by numerically differentiating the acceleration signal. A comparison between model and experimental acceleration and jerk can be seen in Fig. 2.21.

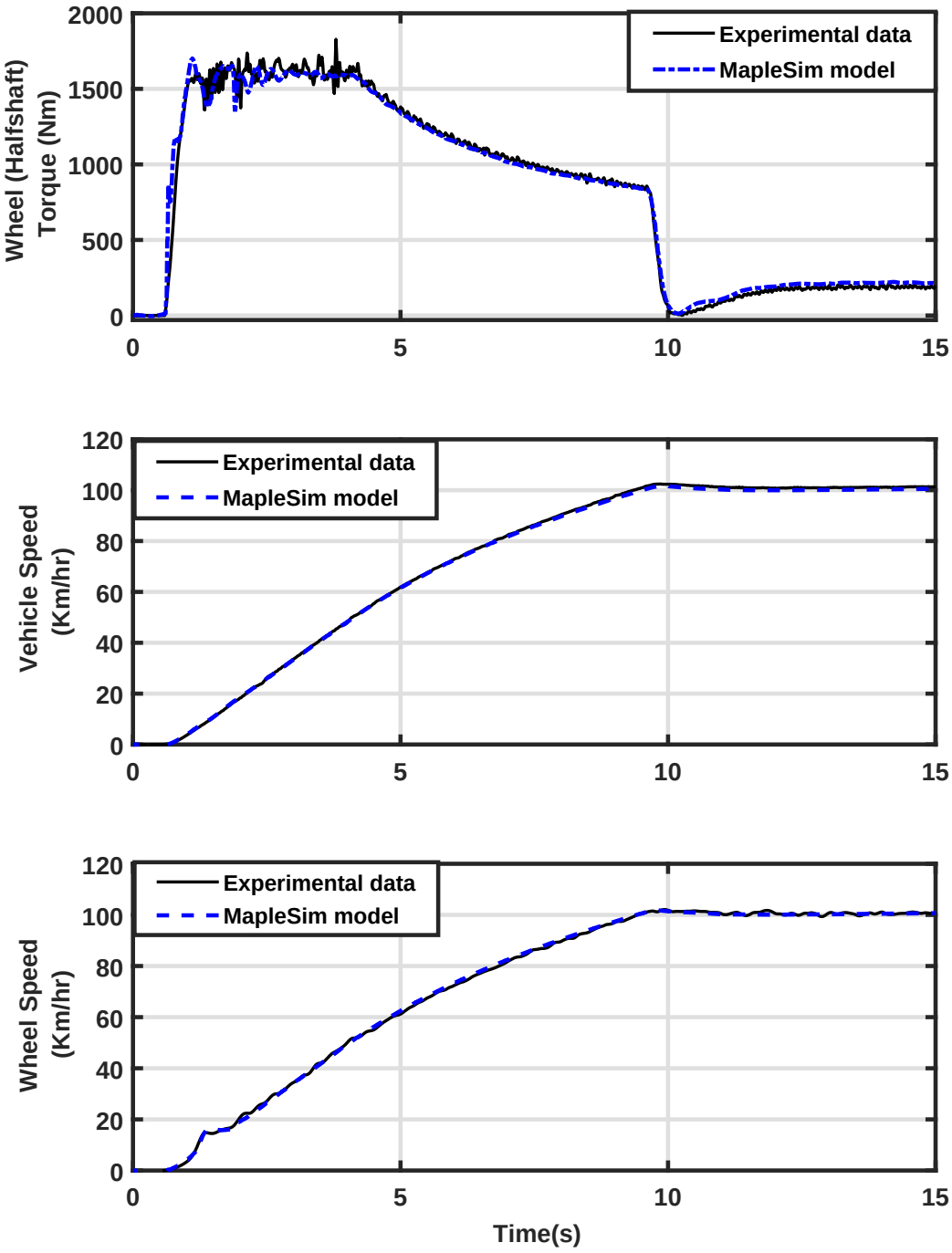

Figure 2.20: Comparison of wheel torque, vehicle speeds and wheel speeds - plant model and experimental data

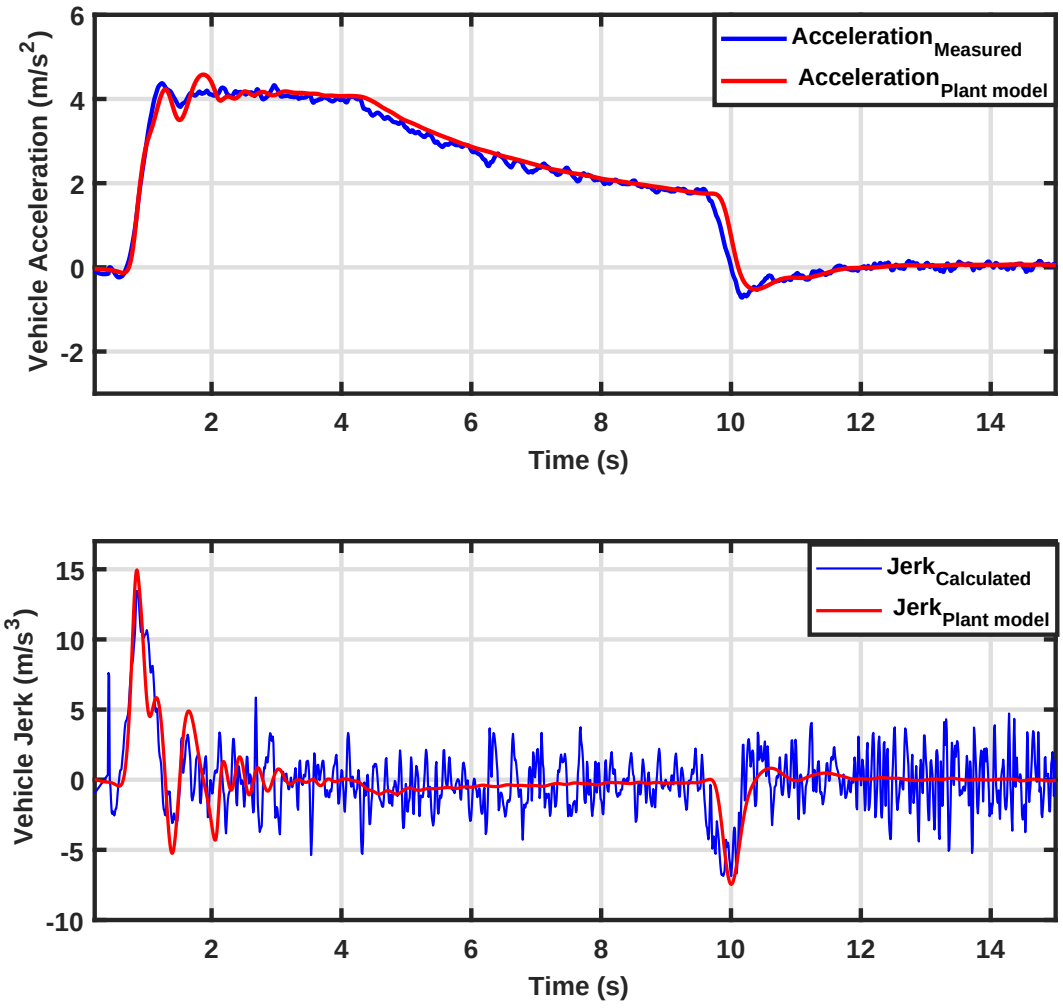

Figure 2.21: Comparison of acceleration and jerk - plant model and experimental data

2.6 Summary

This chapter illustrates a quick and accurate method for identifying parameters of a longitudinal dynamics model of an EV based on road testing. A real-time implementable, 18 DOF longitudinal dynamics plant model is presented. The models have been validated against experimental data collected from the vehicle. A list of all the parameters identified for the plant are given in Table 1.

Chapter 3

Model Predictive Control Design

3.1 Introduction and Literature Review

Model Predictive Control (MPC) is an optimal control strategy, which computes the optimal future control input, taking into account the constraints on system variables. The MPC algorithm consists of a prediction model, cost function and constraints. At the heart of MPC resides a control-oriented model (predictor) that predicts the future behavior of the system over a time frame. These future predictions are evaluated in an optimizer based on a cost function, and is subjected to constraints. The optimal control input, or control action, computed by the optimizer is applied to the plant model for one timestep, after which the process repeats. A block diagram representation of the MPC implementation scheme is shown in Fig 3.1.

In MPC, the optimization is performed over a moving horizon window. The length of the moving horizon window is called the prediction horizon (N_p), which determines the extent to which the future is predicted. The MPC problem, at each time instant, computes a vector of future optimal control inputs (Δu), with the aim to minimize the difference between the reference and predicted outputs over N_p steps. The length of optimal control vector Δu is called the control horizon (N_c). Out of the vector Δu, only the first control input is applied, while neglecting the rest. This principle is called a Receding Horizon Control [41].

The MPC torque control strategy has a great potential for control of automotive drivelines [42, 43]. The main advantages of MPC are its ability to deal with multiple input and multiple output systems in an optimal sense, and its ability to handle multiple constraints, while computing the optimal control input. Further, the capability of MPC to influence the transient response characteristics [44] by adjusting the weights of the objective function, and

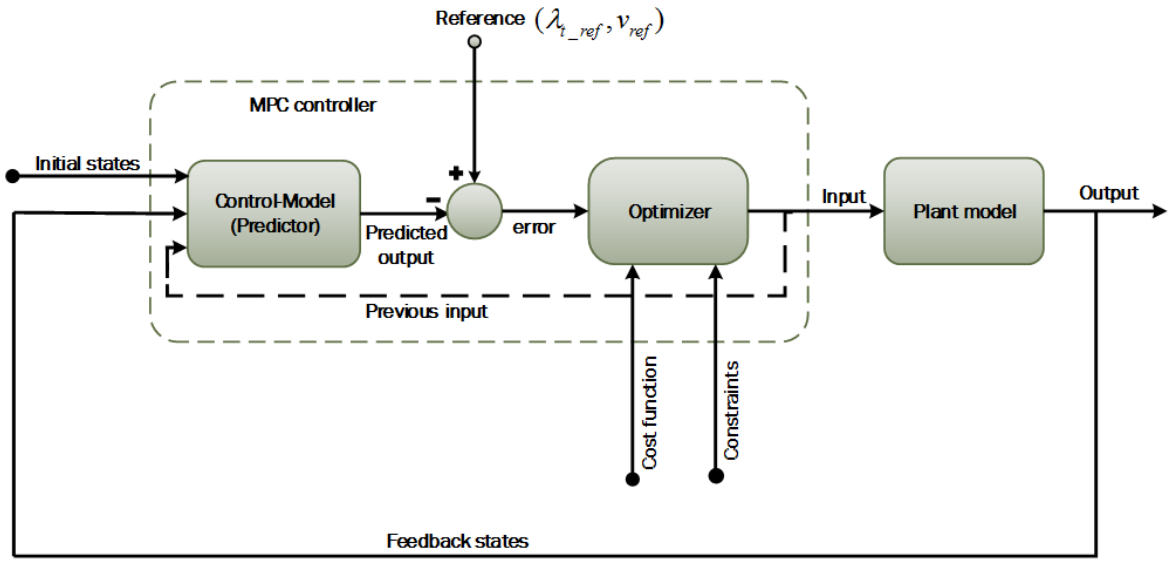

Figure 3.1: Block diagram representation of a MPC controller

to implement constraints at each time step makes it an apt choice for the design of anti-jerk controllers, where driveline oscillations occur as transients during the driving maneuvers. With the ability of MPC to compute the future optimal control input at the current time step, it is an ideal choice for connected and autonomous vehicles [45], considering that such vehicles may have access to future information. Also, MPC can used for linear as well as non-linear systems and has the ability to reject disturbances [46].

The accuracy and speed of an MPC controller depends on the control-oriented model built into its structure. In this chapter, we discuss the design of two different control-oriented models for the design of anti-jerk controllers. First, we discuss the design of common anti-jerk control-oriented models, mainly designed for ICE driven vehicles, which aim to damp oscillations through halfshaft torsion control [18, 47, 48]. Since slipping of wheels often occurs in EVs due to sudden changes in driver torque demand [16, 49], we have developed a more realistic anti-jerk control-oriented model, which includes tire-friction effects in addition to halfshaft torsion. We have used the linear tire as well as the non-linear Pacejka tire to model the tire-road friction. While adding non-linearity to the MPC makes the control design more accurate, it also makes the MPC computationally expensive, thus affecting the real-time implementation ability.

The design of control-oriented models and the procedure for identifying parameters for the models is discussed in section 3.2. The design of MPC-based linear and non-linear traction and cruise control applications is discussed in section 3.3. The HIL setup used to

assess the real-time performance of the MPC traction and cruise controllers is discussed in section 3.4.

3.2 Control-Oriented Models

The accuracy and speed of an MPC controller depends on the control-oriented model built into its structure. In this section, we design a control-oriented model of an EV that includes a flexible halfshaft and effects of wheel slip.

For an EV fitted with a central drive powertrain, the central motor connects to the two forward wheels through a gearbox and flexible shafts, which in turn connect to the wheels. A powertrain model for a front-wheel drive EV with a central drivetrain is shown in Fig. 3.2. Since the powertrain is symmetric about each of the forward wheels, the equations of motion have been derived for only one half of the powertrain. In this model, J_m and J_t represent

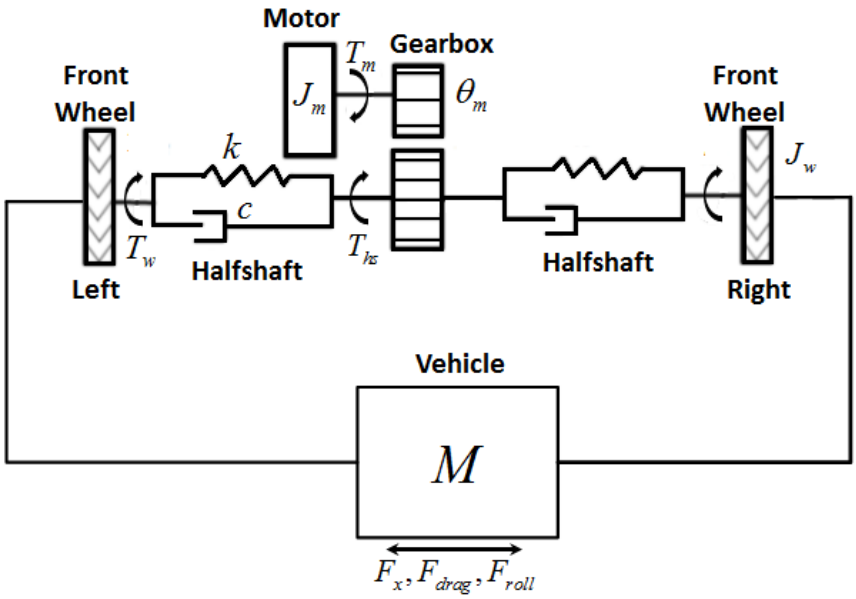

Figure 3.2: Powertrain model for EV with central drivetrain

half the inertia of the motor and transmission and J_w, the inertia of the wheel. The halfshaft is modeled as a torsional spring and damper with stiffness k and damping c. The inertia of the halfshaft is neglected since it is small as compared to the mass of the vehicle. The mass of the vehicle is represented as M, reduction ratio of the gearbox as gr, dynamic radius of the wheel as r_{eff} and torsion angle in the halfshaft as $\frac{\theta_m}{gr} - \theta_w$.

In general, powertrain models for anti-jerk control are based on either halfshaft torsion control or wheel slip ratio control. In this research, we have combined the two approaches for better anti-jerk performance.

3.2.1 Model based on Halfshaft Torsion Control

The vehicle can be modeled as a two-inertia system [50], where the inertia of the vehicle and the wheel is lumped on one side while the inertia of the motor and transmission is lumped on the other side. In this model, it is assumed that a tire is a rigid body, and its inertia is added to that of the vehicle. The halfshaft is modeled as a torsional spring and damper connecting the two inertias. A diagram representing the system is shown in Fig 3.3.

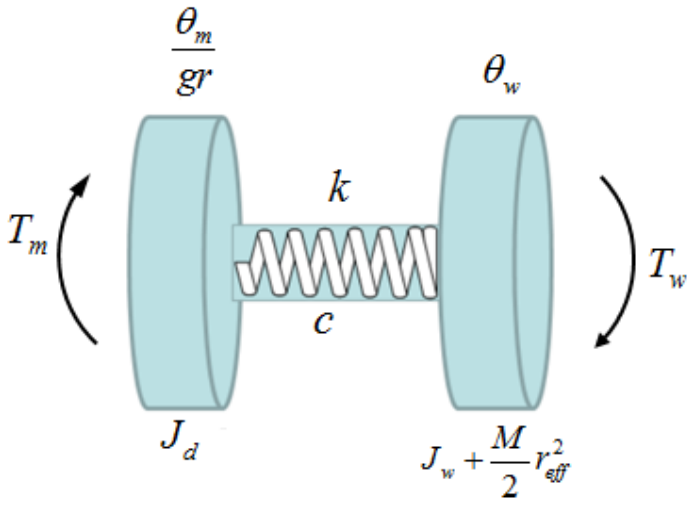

Figure 3.3: Diagram of the EV powertrain

The dynamic equation representing the motor and transmission dynamics for a single front wheel is:

$$J_d \dot{\omega}_m = \frac{T_m}{2} - \underbrace{\frac{k}{gr}\left(\frac{\theta_m}{gr} - \theta_w\right) - \frac{c}{gr}\left(\frac{\omega_m}{gr} - \omega_w\right)}_{\frac{T_{hs}}{gr} = \frac{T_w}{gr}} \tag{3.1}$$

where T_m is the motor torque, T_{hs} is the halfshaft torque which is the same as wheel torque (T_w) considering negligible halfshaft inertia, ω_m is the angular velocity of the motor, ω_w is the angular velocity of the wheel, and the drivetrain inertia is given by:

$$J_d = J_m + \frac{J_t}{gr^2} \tag{3.2}$$

The wheel dynamics can be represented as:

$$J_w\dot{\omega}_w = T_w - r_{eff}\frac{F_x}{2} - f_{rr}F_{zf} \tag{3.3}$$

where F_x the total longitudinal force on the vehicle, which is the sum of the force on the front wheels, f_{rr} is the rolling resistance constant, and F_{zf} is the normal load on each of the front wheels. The vehicle acceleration ($\dot{v}$) is given by:

$$M\dot{v} = F_x - F_d - F_r \tag{3.4}$$

where the aerodynamic drag $F_d = \frac{1}{2}D_c v^2$ where $D_c = \rho C_d A_f$. The parameter ρ is the density of air, C_d is the aerodynamic drag coefficient and A_f is the frontal area of the vehicle, and F_r is the force due to rolling resistance. Since the tire is assumed to be rigid, the vehicle acceleration can be related to angular wheel acceleration as $\dot{v} = r_{eff}\dot{\omega}_w$. The traction force ($F_x$) in equation (3.4) can therefore be represented as:

$$F_x = Mr_{eff}\dot{\omega}_w + F_d + F_r \tag{3.5}$$

Substituting for F_x in equation (3.3), the wheel and vehicle dynamics are given by:

$$J_2\dot{\omega}_w = T_w - r_{eff}\frac{(F_d + F_r)}{2} - f_{rr}F_{zf} \tag{3.6}$$

where $J_2 = J_w + \frac{M}{2}r_{eff}^2$. Combining equations (3.1) and (3.6), the system model representing the dynamics of the powertrain in state-space form is represented as:

$$\begin{aligned} \dot{X} &= Ax + Bu + Eu_d \\ Y &= Cx \end{aligned} \tag{3.7}$$

where

$$\begin{aligned} A &= \begin{bmatrix} 0 & \frac{1}{gr} & -1 \\ -\frac{k}{J_d gr} & \frac{-c}{J_d gr^2} & \frac{c}{J_d gr} \\ \frac{k}{J_2} & \frac{c}{J_2 gr} & \frac{-c}{J_2} \end{bmatrix}; \quad B = \begin{bmatrix} 0 \\ \frac{1}{2J_d} \\ 0 \end{bmatrix}; \quad E = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ r_{eff} & f_{rr} \end{bmatrix} \\ C &= \begin{bmatrix} k & \frac{c}{gr} & -c \\ 0 & 0 & r_{eff} \end{bmatrix}; \end{aligned} \tag{3.8}$$

with states as: $x = \begin{bmatrix} \frac{\theta_m}{gr} - \theta_w & \omega_m & \omega_w \end{bmatrix}^T$,

input $u = T_m$, disturbance input $u_d = \begin{bmatrix} 0 & \frac{-(F_d+F_r)}{2} & -F_{zf} \end{bmatrix}^T$, and outputs $Y = \begin{bmatrix} T_{hs} & v \end{bmatrix}^T$.

3.2.2 Model based on Halfshaft Torsion and Wheel Slip Control

The model discussed in section 3.2.1 considers the tire as a rigid body; thus it assumes that the longitudinal force applied to the wheel is transmitted to the tire (3.5). However, the longitudinal force acting on the tires is a function of the tire-road friction. Therefore, a more realistic vehicle model, which includes the tire flexibility and tire-road friction effects, has been formulated. This model will help control wheel-slip during sudden changes in driver torque demand, in addition to halfshaft torsion, or during changes in road condition.

The equations for motor/transmission dynamics (3.1) and (3.2), wheel dynamics (3.3) and vehicle acceleration (3.4) for this wheel-slip based model is the same as that for the halfshaft torsion model. However, the simplistic longitudinal force model (3.5) for the halfshaft torsion model is replaced with linear or non-linear tire models. Several tire models like Lugre [51], Burckhardt [52], Kiencke and Dias [53] and Pacejka 2002 [54] have been used to model tire-road friction. Out of the these, we have used the Pacejka tire model, since parameters have been identified for this model.

Slip-based transients play an important role in driveline oscillations especially during tip-in and tip-out maneuvers. Since our aim is to develop models for anti-jerk control, we have modeled slip-based transients based on Pacejka's relaxation length model [55]. The dynamic equation for transient slip λ_t can be expressed as:

$$\dot{\lambda}_t = \frac{r_{eff}\omega_w}{Rl} - \frac{v}{Rl} - \frac{v}{Rl}\lambda_t \tag{3.9}$$

where (Rl) is defined as the distance needed by the tire to reach a certain percentage of the steady state slip with a step change in slip. The longitudinal relaxation length (Rl) of the tire is assumed as 0.3 as in [56]. To avoid singularities close to 0 speed as discussed in [57], we linearize equation (3.9) about an operating speed v_{op}:

$$\dot{\lambda}_t = \frac{r_{eff}\omega_w}{Rl} - \frac{v}{Rl} - \frac{v_{op}}{Rl}\lambda_t \tag{3.10}$$

Therefore, including the effect of slip-based transients, the longitudinal force on each tire based on Pacejka non-linear tire model and linear tire model is:

(a) **Pacejka tire model**: Based on equation (2.4):

$$F_x = 2D \sin\left[C\tan^{-1}\left\{B\lambda_t - E\left(B\lambda_t - \tan^{-1}B\lambda_t\right)\right\}\right] F_{zf} \tag{3.11}$$

where parameters B, C, D, and E are experimentally identified as discussed in section 2.4.1. The normal load F_{zf} on each of the front wheels based on equation (2.6) is calculated as:

$$F_{zf} = \frac{M}{2L}(gl_r - h\dot{v}) \tag{3.12}$$

The pitch acceleration $\ddot{\theta}$ for a straight line maneuver is small and therefore neglected.

(b) **Linear tire model**: The longitudinal force on the vehicle is given by:

$$F_x = 2C_t\lambda_t \tag{3.13}$$

The transient longitudinal stiffness C_t is given by:

$$C_t = C_x Rl, \tag{3.14}$$

where the longitudinal stiffness $C_x = (BCD)F_{zf}$.

Combining equations (3.1) to (3.4) and (3.10) to (3.12)/(3.13), a model for anti-jerk or driveline oscillation control of EVs is formulated. The linear model in state-space form is represented as:

$$\begin{aligned} \dot{X} &= Ax + Bu + Eu_d \\ Y &= Cx \end{aligned} \tag{3.15}$$

where

$$A = \begin{bmatrix} -\frac{c}{J_d g r^2} & \frac{c}{J_d g r} & 0 & \frac{-k}{J_d g r^2} & \frac{k}{J_d g r} & 0 \\ \frac{c}{J_w g r} & \frac{-c}{J_w} & 0 & \frac{k}{J_w g r} & \frac{-k}{J_w} & \frac{-C_t r_{eff}}{J_w} \\ 0 & 0 & -\frac{D_c v_{op}}{M} & 0 & 0 & \frac{2C_t}{M} \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{r_{eff}}{Rl} & \frac{-1}{Rl} & 0 & 0 & -\frac{v_{op}}{Rl} \end{bmatrix} B = \begin{bmatrix} \frac{1}{2J_d} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} E = \begin{bmatrix} 0 & 0 \\ \frac{1}{J_w} & 0 \\ 0 & \frac{1}{M} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} \frac{c}{gr} & -c & 0 & \frac{k}{gr} & -k & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & r_{eff} & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.16}$$

with states: $x = \begin{bmatrix} \omega_m & \omega_w & v & \theta_m & \theta_w & \lambda_t \end{bmatrix}^T$,

input: $u = T_m$, disturbance inputs: $u_d = \begin{bmatrix} 0 & -f_{rr}F_{zf} & -F_d - F_r & 0 & 0 & 0 \end{bmatrix}^T$,

and outputs: $Y = \begin{bmatrix} T_{hs} & v & Wh_{sp} \end{bmatrix}^T$.

Difference in Pacejka and Linear Tire Models

The longitudinal force approximated using the Pacejka tire model (3.11) and that by the linear tire model (3.13) has been compared in Fig. 3.4. It can be seen that initially, there is no major difference in the rate of increase of longitudinal force with slip. However, at slip ratios higher than 0.035, the longitudinal force approximated by the linear tire model is higher than that recorded by the Pacejka tire model. It can also be seen that the longitudinal force for the Rav4EV saturates at 5300 N, when a slip ratio of 0.062 is reached. The control bounds for longitudinal force and slip should accordingly be defined below these limits.
While the Pacejka tire model estimates the longitudinal force more accurately than a linear tire model, it will result in a higher computation time. The control design is a choice between accuracy and computation time. Real-time implementation with both linear and non-linear Pacejka tire models is discussed in Chapters 4 and 5.

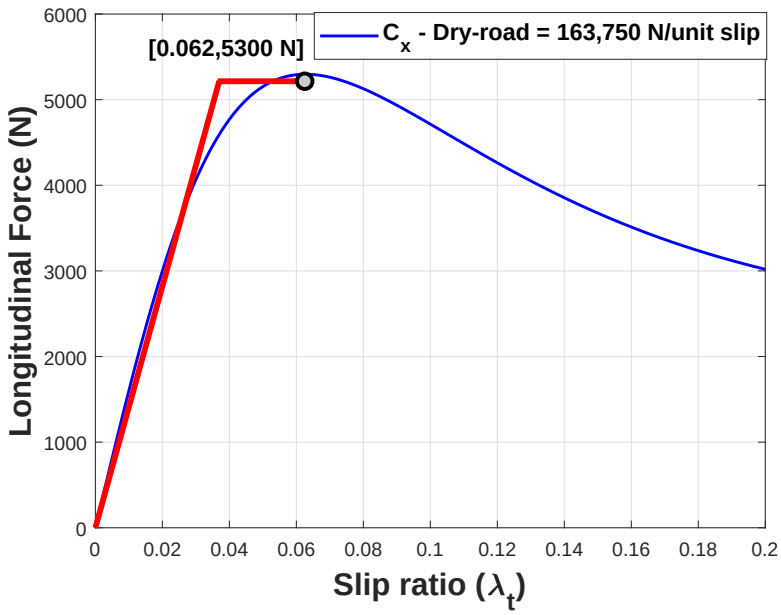

Figure 3.4: Comparison of longitudinal force vs wheel slip in linear and Pacejka tire models

3.2.3 Parameter Identification for Control-Oriented Model

Using an experimental motor torque input shown in Fig. 2.19, the optimization procedure as discussed in section 2.5 was used to identify the parameters of the control-oriented model. The parameters identified for the plant model (Table 1) were used as an initial guess. The list of parameters identified for the control-oriented models are given in Table 2. A comparison of outputs from the control-oriented model (halfshaft torque, vehicle speed and wheel speed) with the experimental data for a test, where the vehicle speed is increased from 0-100 Km/hr, is shown in Fig. 3.5.

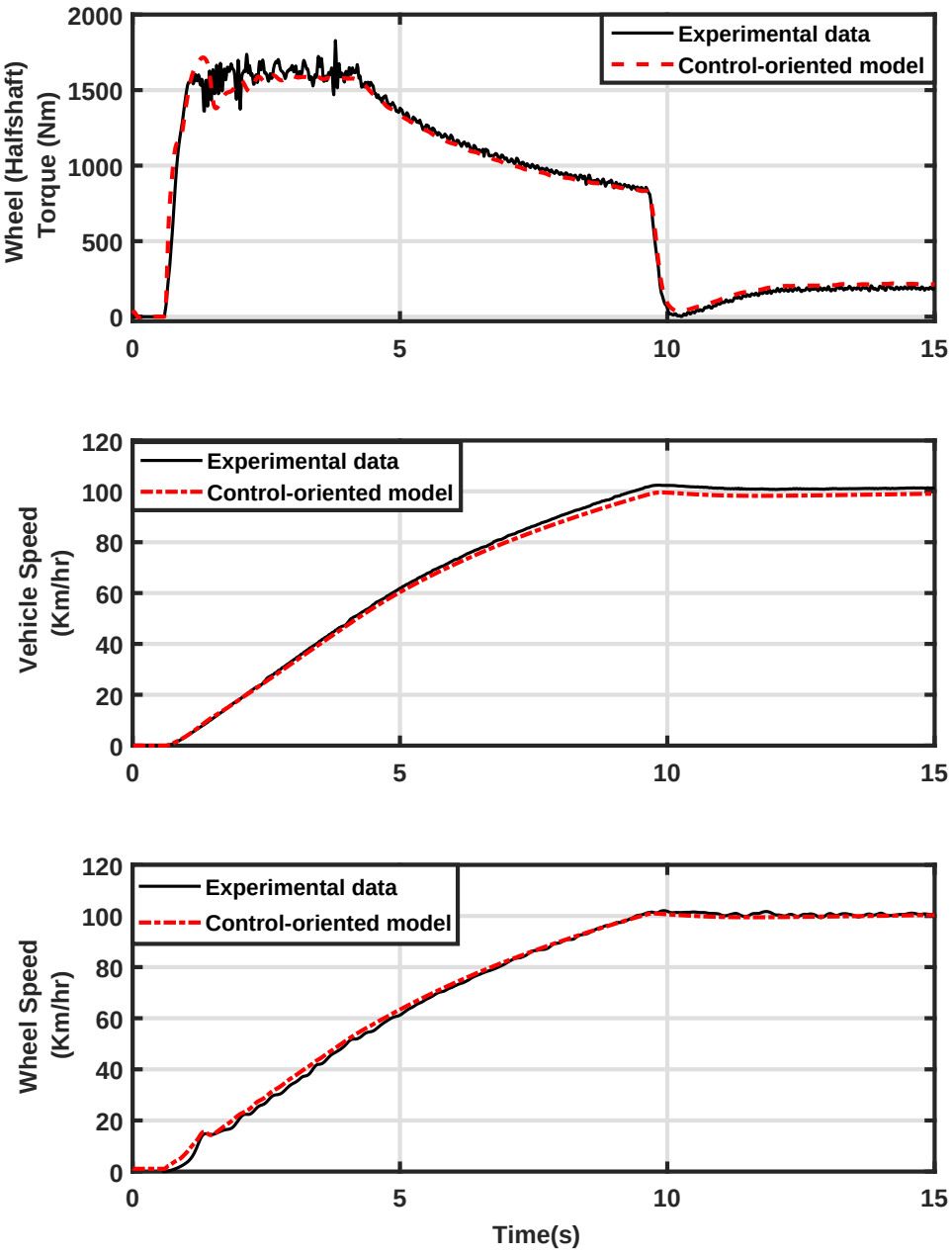

Figure 3.5: Comparison of wheel torque, vehicle speeds and wheel speeds - control-oriented model and experimental data

3.3 Design of MPC for Anti-Jerk Control

The MPC control effort is aimed at regulating the input motor torque to serve the multiple objectives of reference slip tracking ($y = \lambda_t$) in case of traction control or velocity tracking ($y = v$) in case of cruise control. It is assumed that the reference slip or reference vehicle velocities are available to the MPC ahead of time over the prediction horizon N_p. To achieve anti-jerk control, the halfshaft torsion is also minimised ($\frac{\theta_m}{gr} - \theta_w$) in the cost function. Further, to minimize the energy consumption and jerk and wheel/torque oscillations, the input torque $u = T_m$ in case of non-linear MPC, and $\Delta u = \Delta T_m$ for the linear MPC is also included in the cost function. The optimal control problem is subjected to constraints on wheel slip (λ_t) and motor torque (T_m).

The traction control design has been implemented with a linear MPC, while the cruise control design has been implemented using both linear and non-linear controllers. The design of linear and non-linear controllers is explained in the following sections.

3.3.1 Design of Linear MPC

The design of linear MPC is based on linear tire model with wheel-slip transient effects discussed in section 3.2.2. The state-space model given in equation (3.16) is reformulated as in [58] by choosing the optimization variable as the rate of change of motor torque ($\Delta u = \Delta T_m$) instead of the absolute value of the control signal ($u = T_m$). Since ΔT_m is proportional to jerk, it results in a better anti-jerk performance of the controller. Consider the control-oriented system in discrete form as:

$$x_{k+1} = Ax_k + Bu_k + Eu_{d_k} = Ax_k + B\overbrace{u_{k-1}}^{x_{u_k}} + B\Delta u_k + Eu_{d_k} \tag{3.17a}$$

$$x_{u_{k+1}} = x_{u_k} + \Delta u_k \tag{3.17b}$$

$$y_k = Cx_k \tag{3.17c}$$

Therefore, the linear system model can be re-written in matrix form as:

$$\begin{bmatrix} x_{k+1} \\ x_{u_{k+1}} \end{bmatrix} = \underbrace{\begin{bmatrix} A & B \\ 0 & I \end{bmatrix}}_{\overline{A}} \begin{bmatrix} x_k \\ x_{u_k} \end{bmatrix} + \underbrace{\begin{bmatrix} B \\ I \end{bmatrix}}_{\overline{B}} \Delta u_k + \underbrace{\begin{bmatrix} E \\ 0 \end{bmatrix}}_{\overline{E}} u_{d_k} \tag{3.18a}$$

$$y_k = \underbrace{\begin{bmatrix} C & 0 \end{bmatrix}}_{\overline{C}} \begin{bmatrix} x_k \\ x_{u_k} \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \end{bmatrix}}_{\overline{D}} \Delta u_k \tag{3.18b}$$

where $x \in \mathbb{R}^n$, $\Delta u \in \mathbb{R}^m$, and $y \in \mathbb{R}^p$ are the state, input, and output variables of the linear system. Due to the principle of receding horizon control, where only the current information of the plant is required both for prediction and control, it is implicitly assumed that the input does not affect the output [59]. Therefore, $\overline{D} = 0$ in Equation 3.18.

The relation between the predicted output of the system inside the prediction window Y, the measured states at the time step x_k and the designed variation of the inputs ΔU can be represented as:

$$Y = \mathbf{F}x_k + \mathbf{\Phi}\Delta U + \mathbf{\Phi_d}\Delta U_d \tag{3.19}$$

where

$$\mathbf{F} = \begin{bmatrix} \overline{CA} \\ \overline{CA}^2 \\ \overline{CA}^3 \\ \dots \\ \overline{CA}^{N_p} \end{bmatrix}; \mathbf{\Phi} = \begin{bmatrix} \overline{CB} & 0 & 0 & \dots & 0 \\ \overline{CAB} & \overline{CB} & 0 & \dots & 0 \\ \overline{CA}^2\overline{B} & \overline{CAB} & \overline{CB} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \overline{CA}^{N_p-1}\overline{B} & \overline{CA}^{N_p-2}\overline{B} & \overline{CA}^{N_p-3}\overline{B} & \dots & \overline{CA}^{N_p-N_c}\overline{B} \end{bmatrix}$$

$$\mathbf{\Phi_d} = \begin{bmatrix} \overline{CE} & 0 & 0 & \dots & 0 \\ \overline{CAE} & \overline{CE} & 0 & \dots & 0 \\ \overline{CA}^2\overline{E} & \overline{CAE} & \overline{CE} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \overline{CA}^{N_p-1}\overline{E} & \overline{CA}^{N_p-2}\overline{E} & \overline{CA}^{N_p-3}\overline{E} & \dots & \overline{CA}^{N_p-N_c}\overline{E} \end{bmatrix} \tag{3.20}$$

and

$$
\begin{aligned}
Y &= \begin{bmatrix} y(k_i+1|k_i) & y(k_i+2|k_i) & \dots & y(k_i+N_p|k_i) \end{bmatrix}^T \\
\Delta U &= \begin{bmatrix} \Delta u(k_i) & \Delta u(k_i+1) & \dots & \Delta u(k_i+N_c-1) \end{bmatrix}^T \\
\Delta U_d &= \begin{bmatrix} \Delta u_d(k_i) & \Delta u_d(k_i+1) & \dots & \Delta u_d(k_i+N_c-1) \end{bmatrix}^T
\end{aligned} \tag{3.21}
$$

Over the prediction window, the optimal control vector U is defined in matrix form as:

$$
\underbrace{\begin{bmatrix} u_{k|k} \\ u_{k+1|k} \\ \dots \\ \dots \\ u_{k+N_c-1|k} \end{bmatrix}}_{U} = \underbrace{\begin{bmatrix} I_{m\times m} \\ I_{m\times m} \\ \dots \\ \dots \\ I_{m\times m} \end{bmatrix}}_{G_{1,\Delta u}} U_{k-1} + \underbrace{\begin{bmatrix} I_{m\times m} & 0 & 0 & \dots & 0 \\ I_{m\times m} & I_{m\times m} & 0 & \dots & 0 \\ I_{m\times m} & I_{m\times m} & I_{m\times m} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ I_{m\times m} & I_{m\times m} & I_{m\times m} & \dots & I_{m\times m} \end{bmatrix}}_{G_{2,\Delta u}} \underbrace{\begin{bmatrix} \Delta u_{k|k} \\ \Delta u_{k|k} \\ \dots \\ \dots \\ \Delta u_{k+N_c-1|k} \end{bmatrix}}_{\Delta U} \tag{3.22}
$$

Constraints can be added to the system in the form of variation to the input signal as follows:

$$
\underbrace{\begin{bmatrix} G_{\Delta u} \\ G_u \\ G_y \end{bmatrix}}_{\mathbf{G}} \Delta U \leqslant \underbrace{\begin{bmatrix} W_{\Delta u} \\ W_u \\ W_y \end{bmatrix}}_{\mathbf{W}} \tag{3.23}
$$

where

$$
G_{\Delta u} = \begin{bmatrix} -G_{2,\Delta u} \\ G_{2,\Delta u} \end{bmatrix};\ W_{\Delta u} = \begin{bmatrix} -U_{\min} + G_{1,\Delta u} U(k-1) \\ U_{\max} - G_{1,\Delta u} U(k-1) \end{bmatrix}
$$

$$
G_u = \begin{bmatrix} -I \\ I \end{bmatrix}; \qquad W_u = \begin{bmatrix} -\Delta U_{\min} \\ \Delta U_{\max} \end{bmatrix}
$$

$$
G_y = \begin{bmatrix} -\phi \\ \phi \end{bmatrix}; \qquad W_3 = \begin{bmatrix} -Y_{\min} + F x_0) \\ Y_{\max} - F x_0) \end{bmatrix}
$$

In Equation 3.23, the G_{Δ_u} and G_{Δ_u} matrices define the input constraints, G_u and W_u matrices define the input rate constraints, and G_y and W_y matrices define the output constraints.

Let the current optimization problem have the cost function:

$$J = \sum_{j=1}^{N_p-1} \left\{ y^T (k+j\,|k)\ \mathbf{Q}\ y\,(k+j\,|k) \right\} + \sum_{j=1}^{N_p-1} \left\{ \Delta u^T (k+j-1\,|k)\ \mathbf{R}\ \Delta u\,(k+j-1\,|k) \right. \tag{3.24}$$

where $\mathbf{Q}$ is the diagonal matrix of output weights and $\mathbf{R}$ is the input weight.
Substituting the prediction equation 3.19 in equation 3.24, the revised objective function can be written in quadratic form for a case with no terminal penalty as:

$$\begin{aligned} &J(k) = \frac{1}{2}\Delta U^T \mathbf{H} \Delta U + \mathbf{M}\Delta U^T \\ &\text{subject to} \\ &\mathbf{G}\Delta U \leqslant \mathbf{W} \end{aligned} \tag{3.25}$$

where

$$\begin{aligned} \mathbf{H} &= 2(\mathbf{R} + \phi^T \mathbf{Q}\phi + \phi_d^T \mathbf{Q}\phi_d) \\ \mathbf{M} &= 2(x_0^T \mathbf{F}^T - x_{ref}^T(k))\mathbf{Q}\phi + 2(x_0^T \mathbf{F}^T - x_{ref}^T(k))\mathbf{Q}\phi_d \end{aligned} \tag{3.26}$$

and G and W are specified by the constraints of Equation 3.25. The optimal control vector ΔU of length N_c is computed by solving:

$$\Delta U = -\mathbf{H}^{-1}\mathbf{M}^T \tag{3.27}$$

Note that since the first element of ΔU is used to determine the control signal $u(k)$ applied to the plant, the procedure is repeated at each sampling interval. The linear problem has been solved using the *mpcqp* solver [60] in Matlab. Since this solver supports generation of C-code, all HIL tasks could also be performed using this solver.

3.3.2 Design of Non-Linear MPC

The non-linear MPC has been design using the non-linear Pacejka tire model discussed in section 3.2.2. The model also includes the effect of wheel slip transients and load transfer between front and rear wheels. The non-linear MPC was solved using the sequential quadratic programming (SQP) solver using $fmincon$ function in Matlab/Simulink. A brief description of the Matlab based SQP solver can be found in [61]. However, since $fmincon$ does not

support generation of C-code required for undertaking HIL experiments, the problem was also solved using an NMPC solver developed by Maitland et al. [62].

NMPC solver

The NMPC was implemented using a direct single shooting approach with penalty functions to enforce inequality constraints. The Gradients and Hessians required for Newton's method were pre-computed symbolically using Maple. The recursive definition of the states over the prediction horizon found in single-shooting was handled by realizing the Lagrangian as a straight line program of symbolic expressions on which we could carry out automatic differentiation followed by code optimization. This design choice minimized the order of the optimization problem within the NMPC. In particular, since only the control inputs over the control horizon are included in the Lagrangian, and no multipliers or slack variables are introduced, it restricts the number of terms in the Lagrangian. The resultant Lagrangian is of the form:

$$\mathcal{L}(\overline{\mathbf{U}}) = \mathbf{J}(\overline{\mathbf{U}}) + \sum_{i=1}^{h} w_i \rho(g_i(\overline{\mathbf{U}})) \tag{3.28}$$

where $\overline{\mathbf{U}} \in \mathbb{R}^{N_c}$ is a vector of control inputs, $\mathbf{J}$ is the cost function as defined in equation (4.1) or (5.1), $g \in \mathbb{R}^h$ are the inequality constraints, w_i are weighting factors and $\rho(z)$ is a penalty function of inequality constraints represented as:

$$\rho(z) = \left(\frac{2z - (z^{\max} + z^{\min})}{z^{\max} - z^{\min}} \right)^p \tag{3.29}$$

where $z \in [z^{\min}, z^{\max}]$ defines the maximum and minimum of the state/control inputs $(\overline{\mathbf{X}}, \overline{\mathbf{U}})$ included in the constraints penalty function. We set $p = 8$ in our implementation. The behaviour of the function $\rho(z)$ is demonstrated in Fig. 3.6.

It can be seen that as z increases from -10 to 10, the function ρ approaches infinity in the form of a square well and can be defined as:

$$\lim_{p \to \infty} \rho = \begin{cases} 0 & z \in [(z_{\min} + z_{\max})/2] \\ 1 & z \in [z_{\min}, z_{\max}] \\ \infty & z \notin [z_{\min}, z_{\max}] \end{cases} \tag{3.30}$$

Newton's method has been used to optimize the Lagrangian $\mathcal{L}(\overline{\mathbf{U}})$. We consider a single state discrete dynamical system $x_{k+1} = g(x_k)$ then we can capture the recursively defined

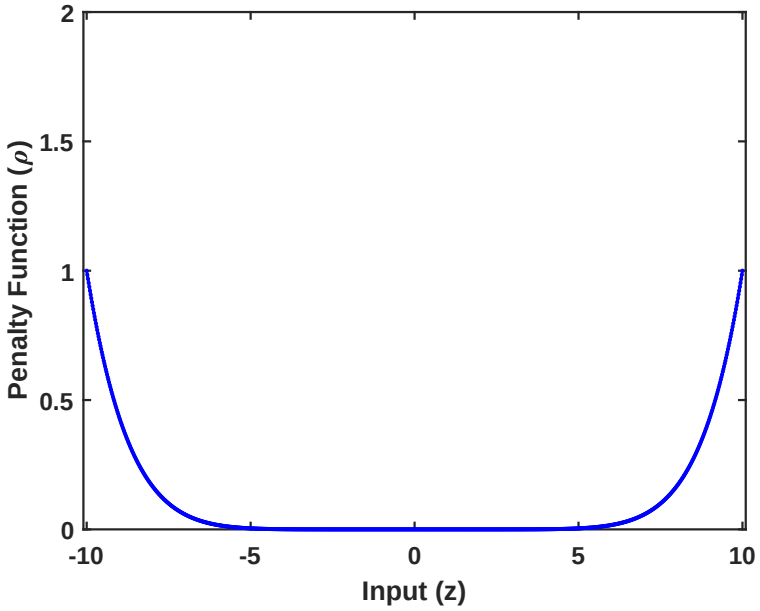

Figure 3.6: Penalty function ρ with any arbitrary input z varying from -10 to 10

state $x_3 = g(g(g(x_0)))$ by the straight line program

$$[\texttt{input} x_0 \tag{3.31}$$

$$t_1 = g(x_0), \tag{3.32}$$

$$t_2 = g(t_1), \tag{3.33}$$

$$x_3 = g(t_2), \tag{3.34}$$

$$\texttt{return} x_3]. \tag{3.35}$$

The Gradient ($\nabla\mathcal{L}$) and Hessian ($\mathbf{H}_{\mathcal{L}}$) of the Lagrangian $\mathcal{L}$ are pre-computed using symbolic computation and optimized for on-line evaluation. The use of symbolics in this manner negates the need for a second simulation loop common to numerical implementations of the direct single shooting method. Thus, the NMPC in this formulation solves for the optimum $\overline{\mathbf{U}}^*$ by finding a minimum of $\mathcal{L}$. This involves repeatedly solving the linear system:

$$\mathbf{H}_{\mathcal{L}}(\overline{\mathbf{U}})\Delta\overline{\mathbf{U}}_{NS} = -\nabla\mathcal{L}(\overline{\mathbf{U}})^T \tag{3.36}$$

while updating $\overline{\mathbf{U}} = \overline{\mathbf{U}} + \Delta\overline{\mathbf{U}}_{NS}$ until convergence, $\Delta\overline{\mathbf{U}}_{NS}$ being the Newton step.

The proposed method sets (3.36) to be of minimal dimension and uses precomputed and optimized functions to evaluate $\mathbf{H}_{\mathcal{L}}$ and $\nabla\mathcal{L}$, aimed at reducing controller turnaround times.

Since it minimizes the computational cost, we are able to forgo the globalization steps found within more advanced Newton based solvers like `fmincon`.

3.4 Performance Assessment of Controllers

Model-based design (MBD) process, cuts down the development costs and makes the development process faster and safer. With the simulation hardware available today, complex vehicle models that can depict the behaviour of a vehicle can be simulated in real-time. In this research, we have tested the MPC through Model-in-the-loop (MIL) and Hardware-in-the-loop (HIL) testing.

3.4.1 Model-in-the-loop Tests

MIL simulation has been undertaken by testing the controller on the MapleSim plant model using Matlab/Simulink. Since the objective of the anti-jerk controllers is to maximize the the slip or velocity tracking ability with minimum jerk, we have formed the following assessment criteria to evaluate the tracking and jerk performance of our controllers:

(a) The root mean square of the slip tracking error $e(t) = \lambda_{t_{ref}} - \lambda_t$ (RMS–STE) in case of traction controller and velocity tracking error $e(t) = v_{ref} - v$ (RMS–VTE) in case of cruise controller is given by:

$$\text{RMS–STE or RMS–VTE} = \sqrt{\frac{1}{t_f - t_i} \int_{t_i}^{t_f} e(t)^2 dt} \tag{3.37}$$

where t_i and t_f are initial and final time respectively over the reference input. The RMSE slip and velocity is considered as a measure of the tracking performance of the controller over the drive cycle. A low RMS-STE or RMS-VTE implies a good tracking performance.

(b) Measurement of longitudinal jerk. As discussed in section 1.4, a maximum acceptable jerk of 2-3 m/s^3 is considered for good drivability.

(c) The integral of absolute value of jerk (IAJ) corresponds to the cumulative sum of jerk

over the reference input and is defined as:

$$IAJ = \frac{1}{t_f - t_i} \int_{t_i}^{t_f} |J_v| \, dt \tag{3.38}$$

IAJ is a measure of the overall jerk performance of the controller over the drive cycle.

(d) The motor energy consumption (in Whr) during acceleration can be represented as the integral sum of the absolute value of motor power consumption and is given by:

$$Energy = \begin{cases} \frac{1}{t_f - t_i} \int_{t_i}^{t_f} |T_m \omega_m| \, dt & [rise - fall(T_m \leq 50Nm)] \\ 0 & [fall(T_m > 50Nm)] \end{cases} \tag{3.39}$$

A trigger based on '$rise$' and '$fall$' of power is used to calculate the overall energy consumed. '$rise$' implies a power increase across any time step while '$fall$' implies power decrease in a time step. Therefore, as long as the power increase is observed across a time step, it is added to the motor power consumption. However, if there is a decrease in power observed across a time step, it is not counted towards the energy consumed. Since EVs are equipped with regenerative braking, when energy is produced during '$fall$' in motor torque ($\leq 50Nm$ for the Rav4EV), it does not contribute towards the power consumed during acceleration and is therefore subtracted from the overall energy consumption. The remaining torque during deceleration has been neglected as it corresponds to mechanical braking and does not contribute towards motor energy consumption or generation.

3.4.2 Hardware-in-the-loop Experiments

Hardware-in-the-Loop (HIL) experiments, controlled through a host computer, have been performed to evaluate the the real-time performance of the MPC on an Electronic Control Unit (ECU). The HIL setup consists of the ECU, which is loaded with the designed controller, and a Real-Time computer (RTC) that runs the MapleSim plant model. A Control Area Network (CAN) bus connects the ECU to the RTC. A schematic diagram of the HIL architecture can be seen in Fig. 3.7. The specification of the ECU, RTC and host computer are shown in Table 3.1.

The MapleSim based longitudinal vehicle dynamics model has been converted to a C-code using the dSPACE real-time workshop compiler/code generator rti1006.tlc. Similarly,

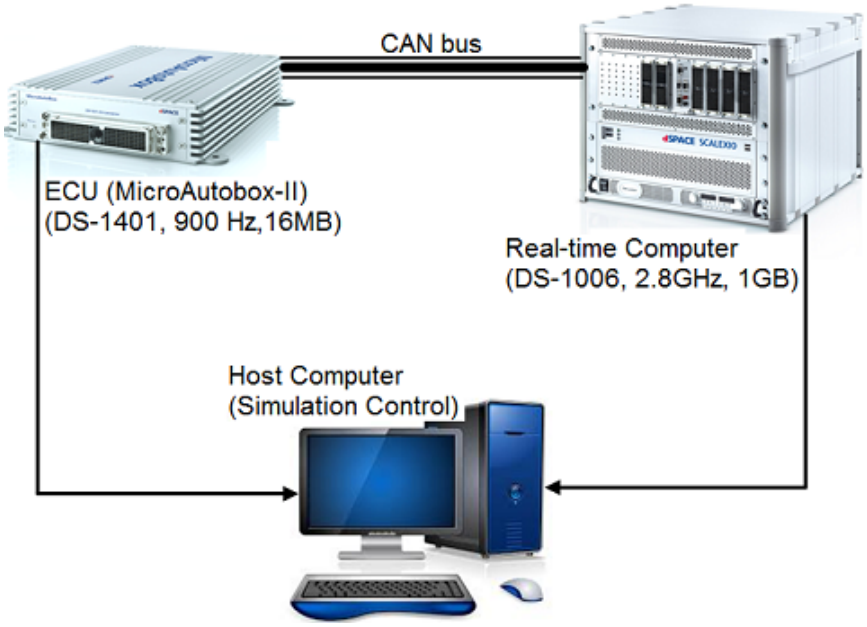

Figure 3.7: Schematic diagram for HIL experiments

the Matlab/Simulink code for the controller was compiled using the rti1401.tlc compiler. The C-code for the controller was then uploaded on the DS1401 processor in the ECU, while the C-code for the plant model was loaded on the DS1006 processor in the RTC.

Component	**Part**	**Specification**
Real-Time Computer (Plant Model)	Processor Memory HIL I/O Board	DS-1006 Quad-Core AMD, 2.8 GHz 1 GB local, 4x128 MB global memory DS-2202
ECU:Micro-AutoBox II (Control-Oriented Model)	Processor Memory HIL I/O interface	DS-1401 PowerPC 750GL 900 MHz 16MB main, 16MB non-volatile memory DS-1511
Host computer	Processor Memory	Core i7, 3.4 GHz 16GB

Table 3.1: Specification for RTC, ECU and Host computer used for HIL experiments

The aim of undertaking HIL experiments is to find the optimal prediction and control horizons (N_p and N_c) that will demonstrate real-time capability of the controller. The real-time performance of the controller is quantified by the turn-around time, which is the time taken for executing the controller code in the ECU. The generally accepted limit for turn-around time is less than 10 milliseconds (ms) for real-time applications [63].

Chapter 4

Anti-Jerk Traction Control

4.1 Introduction and Literature Review

Electric Vehicles (EVs) with central drive powertrain configuration are particularly prone to driveline oscillations. These driveline oscillations are mainly caused by flexible halfshafts and the slip dynamics of tires. However, the central drivetrain is a popular and preferred solution by many manufacturers today primarily due to packaging constraints, and exists in a number of EVs such as the Toyota Rav4EV, Tesla Model S, and Nissan Leaf. The oscillations occur as transients during sudden changes in driver torque demand such as pressing/releasing of accelerator pedal known as tip-in and tip-out, perturbations from the road, backlash in the powertrain or application of friction brakes. These oscillations cause discomfort to the driver and therefore dedicated torque control strategies are required to control these oscillations.

In this chapter, we design an anti-jerk traction or Acceleration Slip Regulation (ASR) controller to maintain vehicle traction during high wheel-slip or sudden changes in speed demanded by the driver.

Typically, traction control designs [64] can be divided into 2 categories:

- Controllers based on wheel-slip feedback control. A number of different approaches including PID, LQR, H_∞, and sliding mode [65–67] have been used to design wheel-slip based traction control
- Controllers independent of wheel-slip such as model-following controllers [68, 69] or maximum transmissible torque estimation (MTTE) based controllers [70–72].

However, anti-jerk control has not been included in both these strategies.

There are also independent strategies for design of anti-jerk controllers ([18, 47, 48]). These strategies mainly deal with damping the oscillations due to halfshaft flexibility. In [73], an anti-jerk controller is designed for damping the resonant low-frequency of the drivetrain in the inner loop and eliminating high-frequency oscillations due to torque reaction in the wheels in the outer loop.

However, there are few design strategies for unified traction and anti-jerk controllers. Recently, a unified approach for design of traction controller with anti-jerk function has been presented in [74]. The authors presented a unified design of anti-jerk traction controllers by adding a driveline oscillation controller developed in [73] in series with gain-scheduled PI, sliding mode, MTTE and H_∞ traction controllers. However, the MPC approach has not been used so far in the design of a unified traction and anti-jerk controller.

In this chapter, we design a real-time MPC-based anti-jerk traction controller for EVs with central drive powertrains. Key features of the MPC controller presented here are: (a) ability to handle multiple control requirements of traction and drivability, (b) handle constraints such as maximum motor torque and wheel slip, and (c) ability to usefully process future road information available through technologies such as Global Positioning Systems (GPS), Intelligent Transportation System (ITS), real-time traffic data and vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) communication systems in connected vehicles.

In particular, the look-ahead (LA) MPC-based traction controller developed in this paper utilizes future road and traffic information to calculate the motor torque demand. The LA-MPC based controller will find potential use in connected vehicles in vehicle-following or speed-following modes. We have designed our controller to meet the dual objectives of traction as well as driveline oscillation/jerk control while maintain drivability and lowering the energy consumption of the vehicle.

4.2 Anti-jerk traction control design

The aim of the anti-jerk traction controller is to regulate the motor torque input so as to achieve the desired slip-tracking performance, with no wheel speed oscillations, minimizing jerk and energy consumption with minimum change in final speed achieved by the vehicle. The following different control designs have been compared:

- Model predictive anti-jerk controller developed as a part of our current research on

connected EVs

- Gain scheduled integral-action controller (as described in [74])
- Maximum transmissible torque estimation (as described in [74, 75])
- Integral sliding model controller (as described in [67])

4.2.1 MPC based control design

The MPC control effort is aimed at regulating the input motor torque (T_m) to serve the following multiple objectives:

- Track a reference slip ratio ($\lambda_{t_{ref}}$) corresponding to an accelerator pedal position input.
- Control the halfshaft torsion by including ($\frac{\theta_m}{gr} - \theta_w$) in the cost function.
- Minimize the energy consumption and jerk and wheel/torque oscillations by including ($\Delta u = \Delta T_m$) in the cost function.

The plant and control-oriented models have been discussed in Section 2.2 and 3.2.2 respectively have been used to design the MPC based traction contoller. The overall control scheme of the MPC traction control is shown in Fig. 5.1.

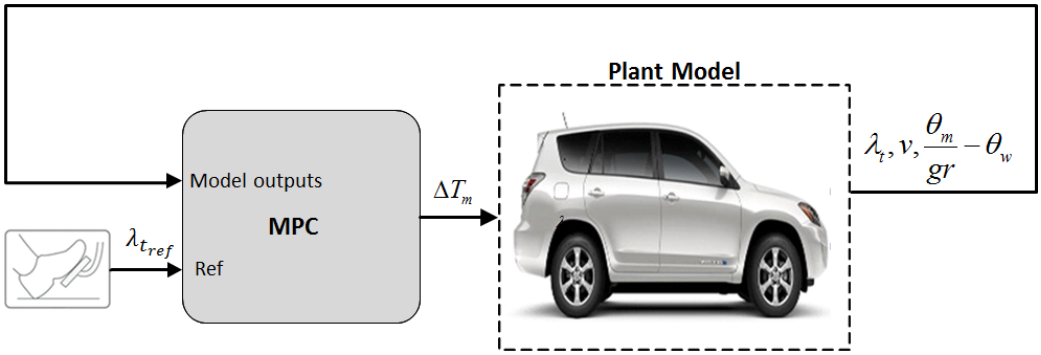

Figure 4.1: Control scheme of MPC controller

The quadratic objective function applied to the optimal control problem is:

$$\underset{u}{Min}\ J = \int_0^{N_p} \left[w_1(\lambda_t - \lambda_{t_{ref}})^2 + (w_2 \Delta u^2) + w_3 \left(\frac{\theta_m}{gr} - \theta_w \right)^2 \right] \tag{4.1}$$

where w_1, w_2 and w_3 are weighting factors for slip tracking, rate of change of torque, and halfshaft torsion respectively in the objective function.

The optimal control problem is subjected to the following constraints:

$$\begin{cases} v > 0 \\ \omega_w > 0 \\ \lambda_{t_{\min}} \leq \lambda_t \leq \lambda_{t_{\max}} \\ T_{\min} \leq u \leq T_{\max} \end{cases} \tag{4.2}$$

The control design for the linear MPC based traction control system has been explained in section 3.3.1. The above problem was solved *mpcqp* solver in Matlab with a sampling period (Δt) of 10 *ms* for discretization.

4.2.2 Gain-scheduled integral action

To implement the slip response characteristics accurately, a gain-scheduled integral action controller, where the integral gain changes with vehicle speed has been implemented. Since the slip response characteristics change with the vehicle operating speed as seen in Fig. 4.3, we implement a gain-scheduled integral action controller. The overall control scheme of the integral action controller is shown in Fig 4.2.

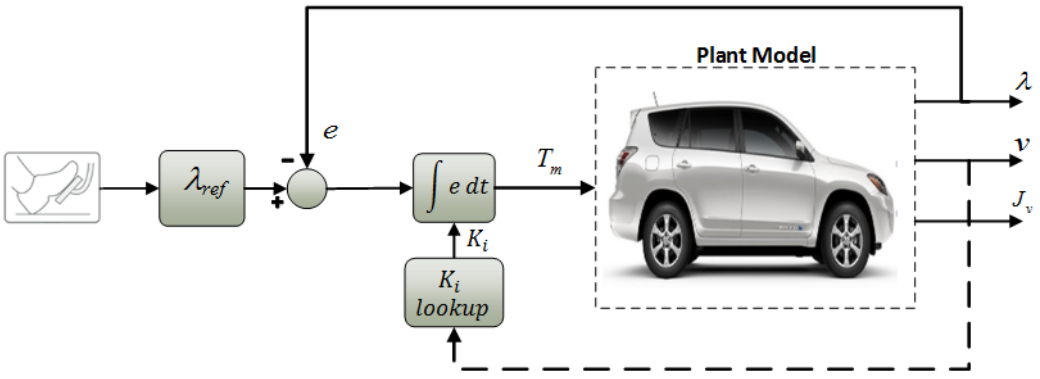

Figure 4.2: Control scheme of Integral action controller

The integral action controller was tuned for anti-jerk by imposing a gain margin (GM) of 15 dB, a phase margin (PM) of 89°, and a tracking bandwidth of 30-50 Hz at -3dB. The integral-gain controller computes the required motor torque input ($u = T_m$), which is

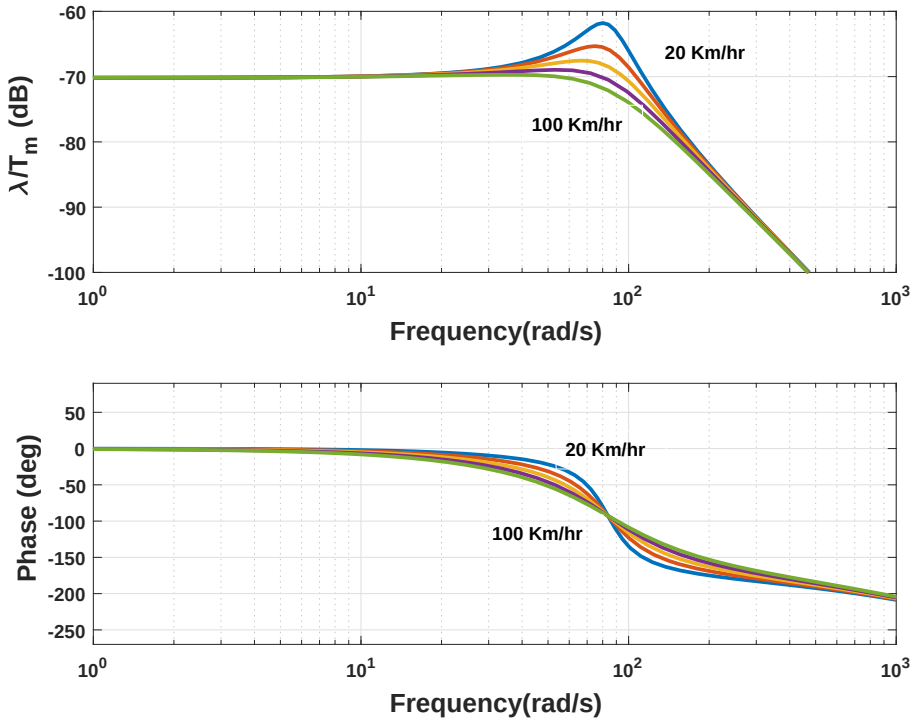

Figure 4.3: Frequency response over a range of operating speeds from 20 Km/hr to 100 Km/hr

expressed over time t as:

$$u(t) = \mathrm{K}_i \int_0^t e\, dt \tag{4.3}$$

where: $e = \lambda_{t_{ref}} - \lambda_t$. The integral-gain constant K_i is tuned to achieve a GM of 15dB at different vehicle speeds, ranging from 20 - 100 Km/hr, as given in Table 4.1.

Vehicle speed (Km/hr)	**Integral-gain** (K_i) (Nm/s)
20	7790
40	10865
60	14580
80	18055
100	21296

Table 4.1: K_i for vehicle speeds ranging from 20-100 Km/hr

4.2.3 Maximum transmissible torque estimation controller

The maximum transmissible torque estimation (MTTE) controller as developed by [75] computes the maximum transmissible torque ($T_{w,max}$) so as to keep the slip within limits. The major advantage of using this controller is that it does not require knowledge of chassis velocity or tire–road condition. It only makes use of torque reference and the wheel spin to estimate the $T_{w,max}$ required to achieve anti-slip control, and accordingly this $T_{w,max}$ is applied as a dynamic limiter to the reference torque. Fig. 4.4 shows the control scheme of the MTTE controller. The wheel torque estimate ($\hat{T_w}$) corresponding to the force between

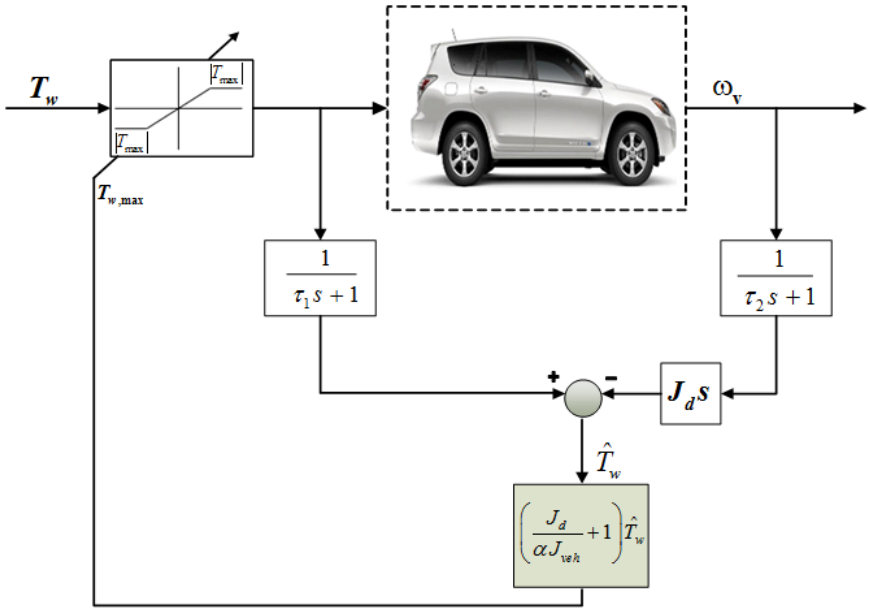

Figure 4.4: Control scheme of MTTE controller

the tire and road surface, neglecting the driveline resistance due to rolling and aerodynamic resistance forces, is given by:

$$\hat{T_w} = (T_m - J_d \dot{\omega}_m) g r \tag{4.4}$$

where the inertia of the onboard drivetrain J_d is given by equation (3.2). An appropriate difference between chassis and wheel speeds due to the tire-road friction is estimated using a relaxation factor α. The relaxation factor is represented as the ratio of angular accelerations of the chassis and the wheel as:

$$\alpha = \frac{\dot{\omega}_v}{\dot{\omega}_w} = \frac{2\hat{T}_w / J_{veh}}{(T_{w,\max} - \hat{T_w})/J_d} \tag{4.5}$$

where $T_{w,\max}$ represents the drivetrain torque in limit conditions and J_{veh} is the equivalent vehicle inertia given by $J_{veh} = Mr_{eff}^2 + 2J_w$. Rearranging equation (4.5), the maximum transmissible torque can be written as:

$$T_{w,\max} = \left(\frac{2J_d}{\alpha J_{veh}} + 1\right)\hat{T}_w \tag{4.6}$$

As $\hat{T}_w$ is calculated from equation (4.4), $T_{w,\max}$ can be calculated from equation (4.6) by considering $\alpha = 0.95$ [75]. Initially, we select α close to 1 which corresponds to a small slip ratio for a dry road condition. As the vehicle enters a slippery road, $T_{w,max}$ is reduced adaptively due to decrease in $\hat{T}_w$.

4.2.4 Integral sliding mode controller

The Integral Sliding Mode (ISM) controller maintains a desired slip by defining a sliding surface. In the proposed implementation scheme seen in Fig. 4.5, a ISM controller is designed to limit the maximum motor torque to $T_{m,law}$ which is calculated depending on the maximum wheel slip permissible on the road surface. The sliding surface $S(\lambda)$ is defined as:

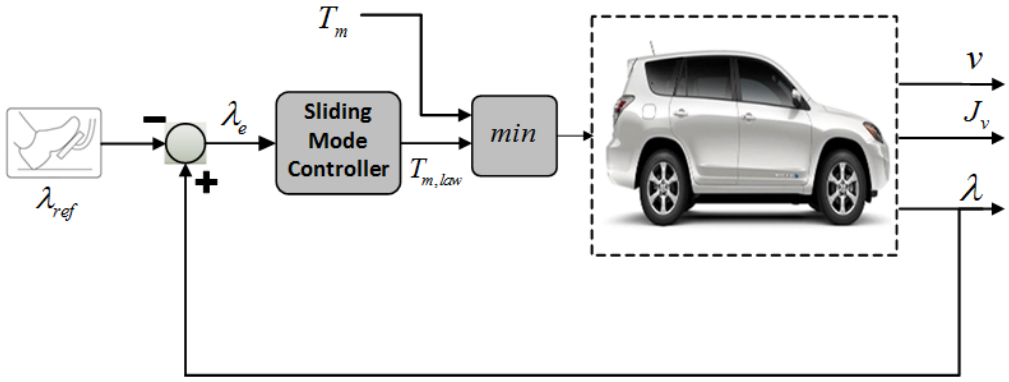

Figure 4.5: Control scheme of ISM controller

$$S(\lambda) = \lambda_e + K_{in}\int_0^t \lambda_e(t)dt \tag{4.7}$$

where $\lambda_e = \lambda - \lambda_{ref}$ and the integral gain $K_{in} > 0$. The integral part action with gain ensures a smooth control action through first order filtering of the discontinuous part of the control action [76]. As the system trajectory is required to stay on the sliding surface, the derivative

of the sliding surface $S(\dot{\lambda}) = 0$. Differentiating equation (4.7) gives:

$$S(\dot{\lambda}) = \frac{-\dot{v} + (1-\lambda)\dot{W}h_{sp}}{\dot{W}h_{sp}} \tag{4.8}$$

where Wh_{sp} is the wheel speed. Combining the wheel dynamics equation (3.3) with (4.8), the following is obtained:-

$$S(\dot{\lambda}) = f + bT_m \tag{4.9}$$

where:

$$\begin{aligned} f &= -\tfrac{g\mu}{Wh_{sp}gr}\left[1 + (1-\lambda)\tfrac{r_{eff}^2 M}{J_w}\right] + J_d\dot{\omega}_m \\ b &= \tfrac{(1-\lambda)\,r_{eff}}{J_w Wh_{sp}} \end{aligned} \tag{4.10}$$

Using (4.7) and (4.9), the sliding model control law can be derived by adding a switching control input ($T_{m_{eqv}}$) to the equivalent control input ($T_{m_{sw}}$) as:

$$\mathrm{T}_{m_{law}} = \mathrm{T}_{m_{eqv}} + \mathrm{T}_{m_{sw}} \tag{4.11}$$

where:

$$\begin{aligned} \mathrm{T}_{m_{eqv}} &= \tfrac{1}{b}\left[-f - K_{in}\lambda_e\right] \\ T_{m_{sw}} &= \tfrac{1}{b}\left[-K\,sat\left(\tfrac{\lambda}{\Phi}\right)\right] \\ sat\left(\tfrac{\lambda}{\Phi}\right) &= \begin{cases} -1, & \lambda < -\Phi \\ \lambda/\Phi & -\Phi \le \lambda \le \Phi \\ 1 & s > \Phi \end{cases} \end{aligned} \tag{4.12}$$

The design parameter Φ defines the boundary layer thickness around the sliding surface and sliding gain K is a tunable parameter. For the system under consideration, the chosen parameters K_{in} and Φ are chosen as 100 and 0.06 respectively.

To improve robustness of the ISM controller to changes in road conditions and vehicle mass as cited in [77], the value of the sliding gain is defined as $K = F + \eta$ where:

$$F = \frac{g}{|v_w|}\left\{|\mu_{\max} - \mu| + (1-\lambda)\frac{r_{eff}^2}{J_w}\left|\frac{M_{\max}}{4}\mu_{\max} - \frac{M}{4}\mu\right|\right\} \tag{4.13}$$

and η is tuned to 15 Nm. The parameter μ_{max} has been selected as 1.18 and M_{max} as 2200 kg.

4.3 Performance assessment

The performance of the four controllers has been compared based on assessment criteria defined in section 3.4.1 in three different scenarios:

- *Scenario 1.* Apply a step throttle input of 0-100% at $t = 2s$ for a vehicle initially at rest. The performance of frozen-time MPC (FT-MPC) with $N_p = N_c = 1$ and look-ahead MPC (LA-MPC) with $N_p = 100$ ($i.e.$ $1s$), $N_c = 5$ is compared to the integral action, ISM and MTTE controllers.
- *Scenario 2.* Apply a throttle pedal input in three steps of 2 seconds each with λ_{ref} increasing in steps of 0.02. The performance of FT-MPC and LA-MPC is compared with the other controllers.
- *Scenario 3.* Comparison of the controllers over a reference throttle input that approximates a US06 drive cycle over a time period of 500 s.

Scenario 1

In Scenario 1, a comparison has been drawn between controllers on application of a step input (0-100% throttle position) at $t = 2s$. The controllers were tuned to achieve similar slip tracking performance and final speed. A comparison of IAJ and maximum jerk generated by each of four controllers for achieving similar tracking performance is given in Table 4.2.

A comparison of results for a FT-MPC controller with $N_p = N_c = 1$ and other controllers can be seen in Fig. 4.6. The comparison of results for a LA-MPC controller with $N_p = 100, N_c = 5$ with other controllers can be seen in Fig. 4.7.

Analysis of Scenario 1 and 2 reveals the following:

- In Scenario 1, for similar slip tracking performance, there is no significant difference between the IAJ and maximum jerk achieved by both Integral-action and the FT-MPC controller. However, the anti-jerk performance of the LA-MPC controller as seen in Scenario 2, far exceeds any other controller. This is because the MPC is able to usefully process the future information available through technologies such as GPS, ITS, V2V or V2X to compute the torque demand in advance.
- While the maximum jerk produced by the ISM controller is lower as compared to some other controllers, the slip tracking performance is not at par with the other controllers,

	Controller	RMS-STE	Final speed	IAJ	Max Jerk
		($\times 10^{-3}$)	(Km/hr)	(m/s^3)	(m/s^3)
Scenario 1	Integral-action	10	56.8	4.3	19.6
	ISM	12	56.5	5.8	10.1
	MTTE	11	56.0	9.2	26.8
	FT-MPC	10	56.8	4.3	**19.6**
	LA-MPC	10	56.3	3.2	**4.5**

Table 4.2: Scenario 1: Comparison of FT-MPC controller ($N_p = N_c = 1$) and LA-MPC controller ($N_p = 100, N_c = 5$) with other controllers

especially in the range of 2-3 s. This is due to the use of input torque rate controller in series with the ISM controller. The IAJ is also higher as compared to the MPC and Integral-action controller.

- The MTTE controller produces the highest jerk and IAJ. Since the input wheel torque is calculated based on a filtered wheel speed output, the effectiveness of the jerk will depend on the filter used. However, use of a higher frequency filter results in lower final speeds.

Scenario 2

To check the system response with a different pedal input, λ_{ref} was increased in three steps of 2 seconds each in Scenarios 2. The tuning parameters were held the same as used in Scenario 1. As in case of Scenario 1, a comparison has been drawn between the controllers by comparing the IAJ and max jerk while achieving the same slip tracking performance and final speeds. Table 4.3 shows the performance analysis of the four controllers.

A comparison of results for a frozen-time FT-MPC controller and LA-MPC controller with other controllers can be seen in Fig. 4.8 and Fig. 4.9 respectively. Analysis of Scenario 2 reveals following:

- Similar slip tracking performance (RMS-STE), jerk and final speed was achieved by both Integral-action and the FT-MPC controller as seen in Fig. 4.8 which is also consistent with results in Fig. 4.6.

- The anti-jerk performance of LA-MPC controller exceeds any other controller as seen in Fig. 4.9 which is consistent with results in Fig. 4.7.

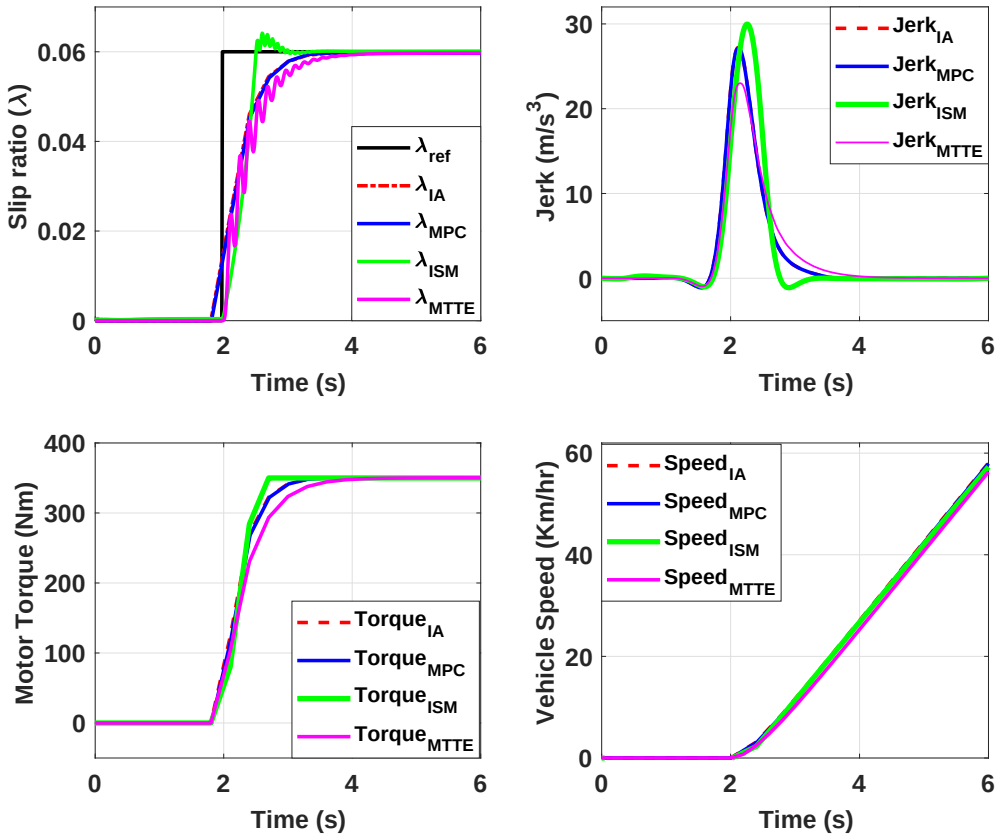

Figure 4.6: Scenario 1: Comparison of FT-MPC controller with other controllers (with λ_{ref} increasing from 0-0.06) to achieve the same slip tracking performance and final speed

- The slip tracking performance of the ISM and MTTE controllers is not at par with the Integral-action and MPC controllers in the intermediate steps (t=1-3s and t=3-5s). However, the final slip tracking performance when λ_{ref}=0.06 is as good as the Integral-action and MPC controllers.
- The jerk produced by the ISM and MTTE controllers in the intermediate steps (t=1-3s and 3-5s) is also much higher in comparison to the integral-action and MPC controllers.

Scenario 3

The performance of the Integral action and the LA-MPC controllers have been compared over a US06 drive cycle (500 s). The ISM and MTTE controllers have not been compared as

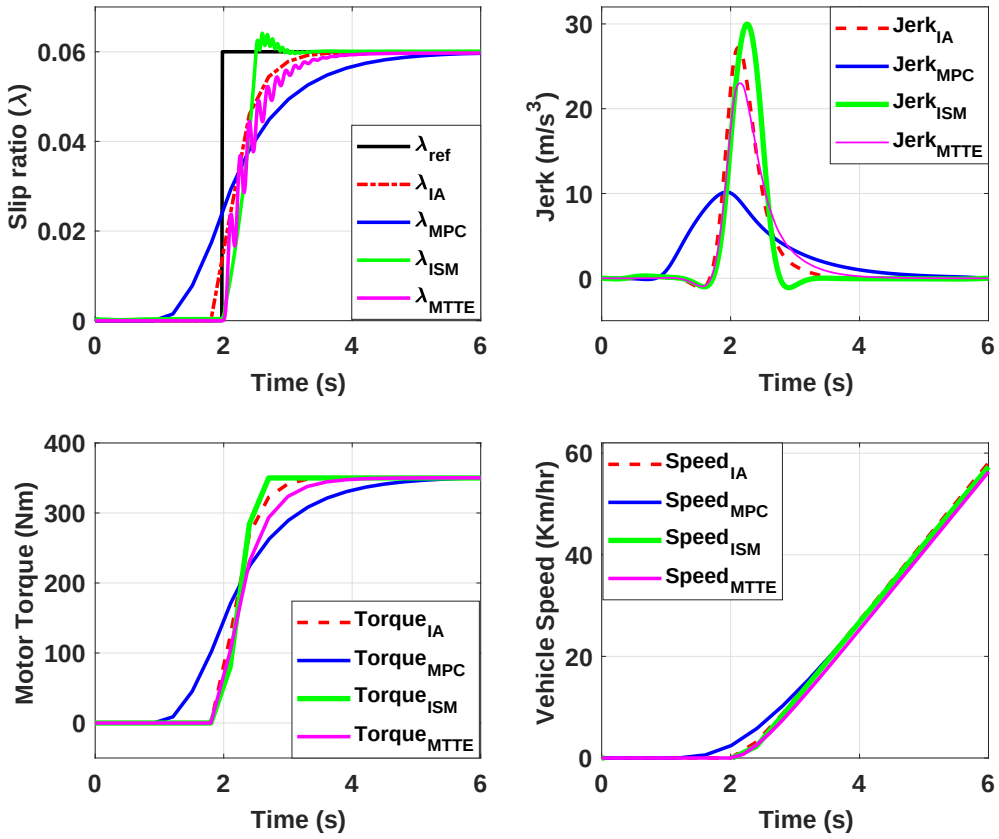

Figure 4.7: Scenario 1: Comparison of LA-MPC controller with other controllers (with λ_{ref} increasing from 0-0.06) to achieve the same final speed

	Controller	RMS-STE ($\times 10^{-3}$)	Final speed (Km/hr)	IAJ (m/s^3)	Max Jerk (m/s^3)
Scenario 2	Integral-action	5.3	68.2	4.3	7.9
	ISM	4.8	72.1	7.8	15.2
	MTTE	4.9	70.6	9.2	11.0
	FT-MPC	5.2	68.5	4.4	**7.9**
	LA-MPC	5.3	68.0	4.2	**1.3**

Table 4.3: Scenario 2: Comparison of FT-MPC controller ($N_p = N_c = 1$) and LA-MPC controller ($N_p = 100, N_c = 5$) with other controllers

their use is limited to acceleration-slip-regulation (ASR) on the positive slip region only, and are activated only when required [73, 74]. Negative slip or braking action as existing in

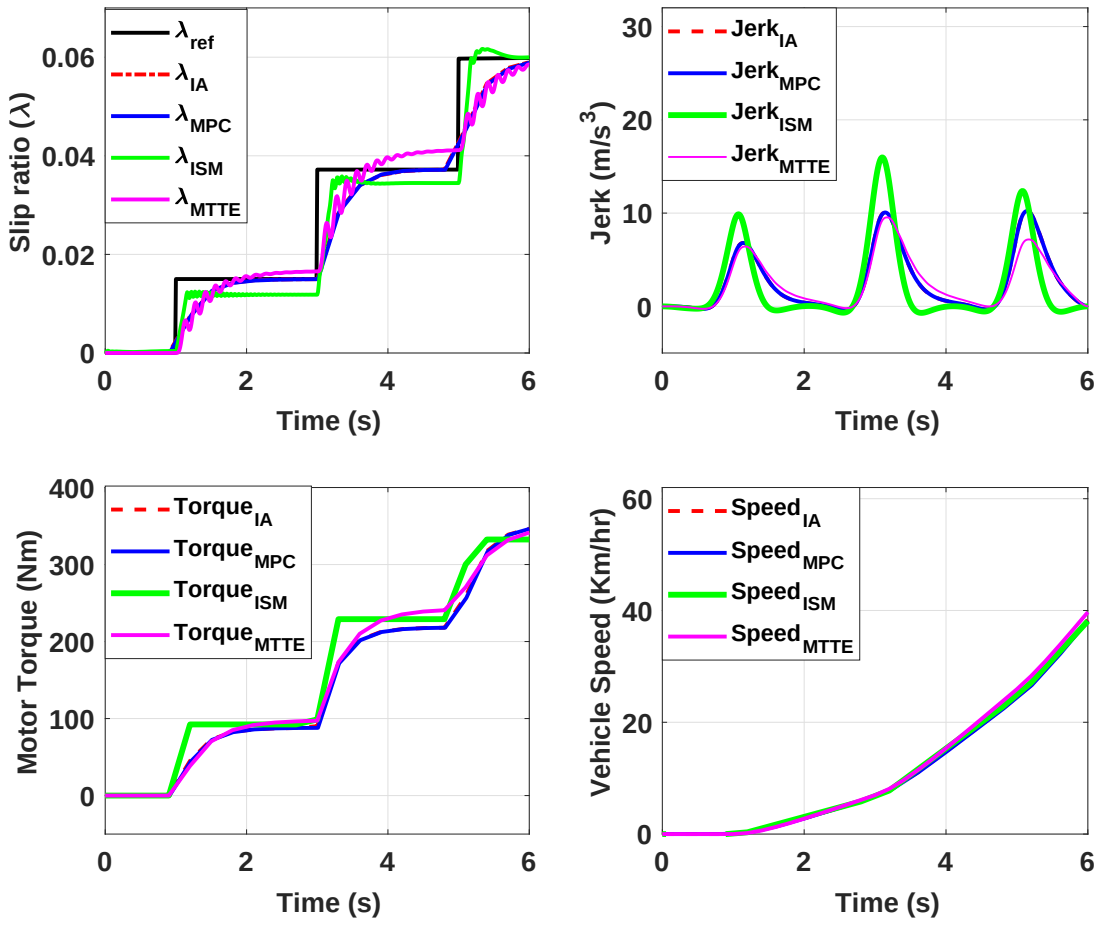

Figure 4.8: Scenario 2: Comparison of FT-MPC controller with other controllers (with λ_{ref} increasing in three steps of 0.02 each over 6 s)

a drive cycle situation cannot be applied to these controllers.

The performance of the controllers was compared at three RMS-STE set-points that correspond to a GM of 15, 20 and 30 dB for the Integral-gain controller. The LA-MPC controller was also tuned to achieve the same RMS-STE and approximately the same energy consumption as the integral action controller. The resulting IAJ and maximum jerk achieved by the two controllers was compared to assess their anti-jerk performance.

Results presented in Table 4.4 show that, when tuned to achieve the same slip tracking performance, the MPC controller has a better anti-jerk performance as compared to the Integral action controller. The energy efficiency of the MPC controller, however, reduces as compared to the Integral-action controller as the GM increases from 15 to 30 dB. Fig. 4.10

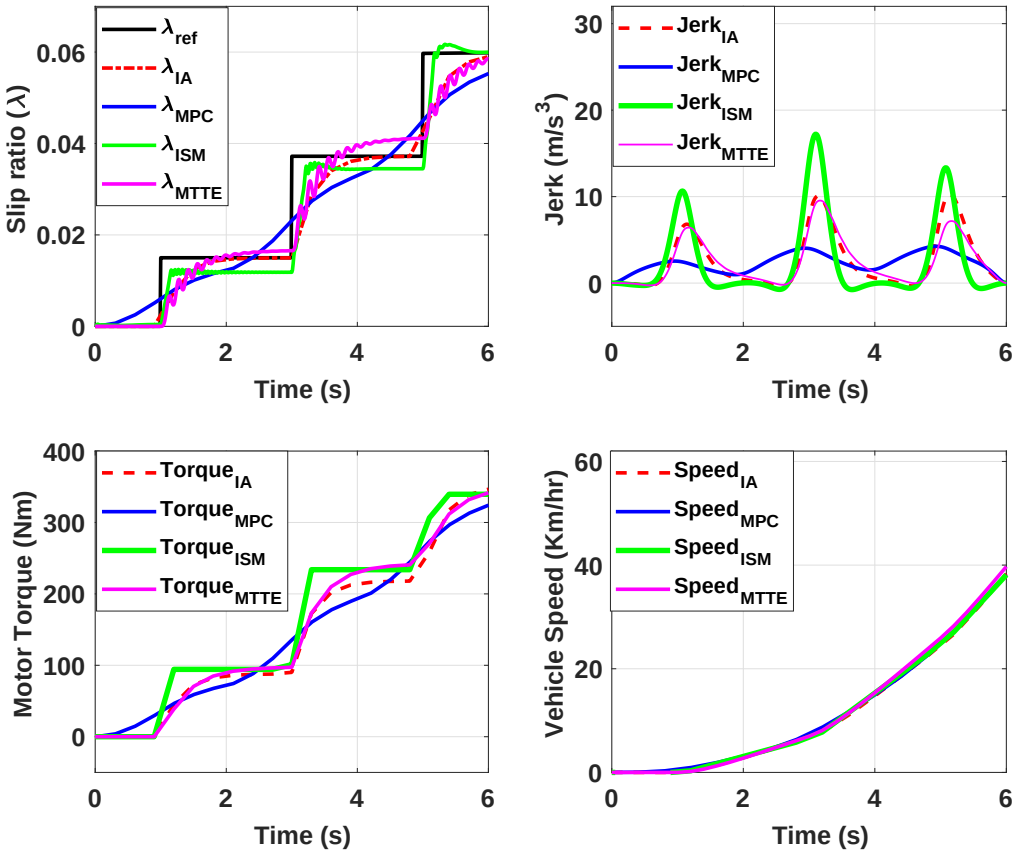

Figure 4.9: Scenario 2: Comparison of LA-MPC controller with other controllers (with λ_{ref} increasing in three steps of 0.02 each over 6 s)

	Controller	RMS-STE ($\times 10^{-3}$)	Energy (Whr)	IAJ (m/s^3)	Max Jerk (m/s^3)
Scenario 3	IA (GM:15dB)	1.5	956	133	3.02
	LA-MPC	1.5	**945**	**86**	**2.15**
	IA (GM:20dB)	2.6	872	116	2.37
	LA-MPC	2.6	**875**	**66**	**1.61**
	IA (GM:30dB)	2.9	819	110	2.13
	LA-MPC	2.9	**856**	**62**	**1.49**

Table 4.4: Integral-action (IA) controller tuned to achieve gain margins of 15, 20 and 30 dB. IAJ and max jerk are compared for three conditions while tuning the LA-MPC controller to achieve the same RMS-STE slip and energy consumption as the IA controller

shows the comparison of energy consumption and Integral jerk between the Integral-action and MPC controllers when tuned to achieve a RMSE slip of 2.6×10^{-3}.

Analysis of the three scenarios shows that both the Integral action and FT-MPC controller have similar performance and therefore can be used as anti-jerk traction controllers in conventional vehicles. The LA-MPC has a superior anti-jerk performance and lower energy consumption (upto a GM of 15 dB) as compared to the other controllers and therefore could find use in connected or intelligent vehicles that have access to future information through V2V or V2I communication.

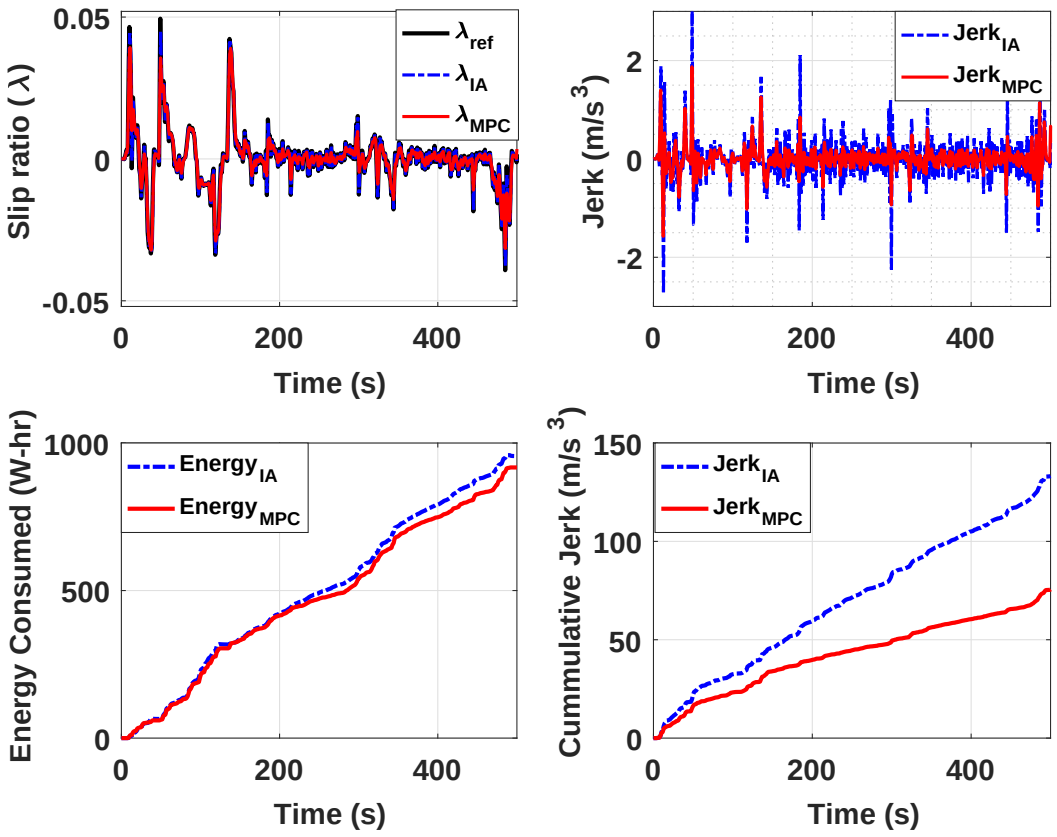

Figure 4.10: Scenario3: Comparison of Integral-action and MPC controller when tuned to a RMSE slip of 0.0015)

4.4 Hardware-in-the-loop experiments

Hardware-in-the-loop (HIL) experiments are used to demonstrate real-time performance of the MPC on an Electronic Control Unit (ECU). A general description of the hardware setup for the HIL experiments has been discussed in section 3.4.2. The aim of the experiments is to compute the optimal prediction horizon (N_p) and control horizon (N_c) that will demonstrate the real-time capability of the controller. The generally accepted upper limit for turn-around time is 10 ms for real-time applications [63].

The choice of prediction and control horizons is generally a trade off between performance and computational time. Increase in prediction horizon (N_p) corresponds to an increase in the knowledge of the future trip while the control horizon (N_c) is proportional to the ECU control effort. The HIL experiments were performed by tuning the controller with different N_p and N_c, such that similar end results are achieved with each setting. Fig. **??**(a) shows the controller turn-around time when increasing N_p from $0-300$ (with $N_c = 10$). It can be seen in the figure that turn-around time increases linearly with increase in N_p up to 1 ms. However, there is a higher variance in turn-around time when $N_p < 50$ (or 0.5 s). Fig. **??**(b) shows that as N_c is increased from $2-18$ (with $N_p = 100$), the turn-around time increases exponentially up to 3.8 ms.

A prediction horizon of $N_p = 100$ and control horizon of $N_c = 5$ has been selected as tuning parameters for the LA-MPC to achieve a maximum turn-around time of 0.2 ms, thus ensuring real-time performance.

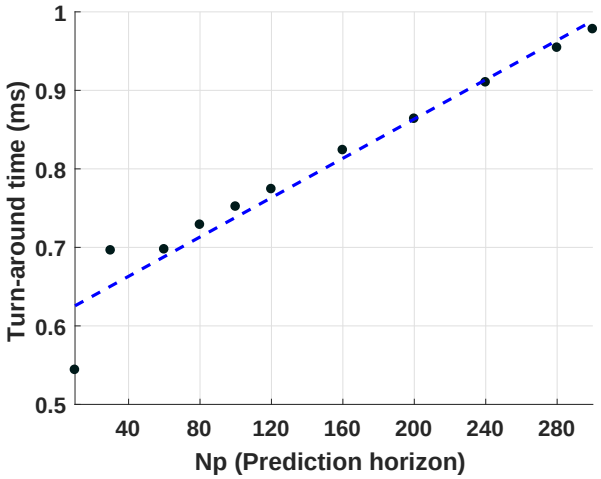

Figure 4.11: ECU maximum turn-around time for different lengths of prediction horizon, $N_c = 10$

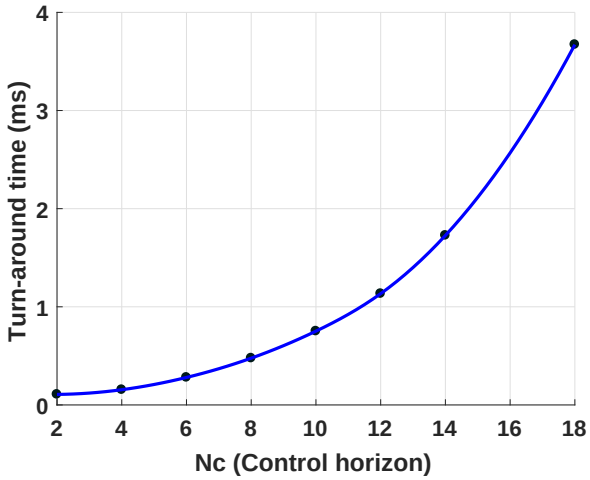

Figure 4.12: ECU maximum turn-around time for different lengths of control horizon, $N_p = 100$

4.5 Summary

A predictive LA-MPC anti-jerk traction controller has been designed for a central drivetrain architecture in EVs to satisfy the multiple objectives of slip tracking performance and enhanced drivability at the same time. A comparison has been drawn between the LA-MPC, Integral-action controller, a robust ISM controller and MTTE controller, all tuned for anti-jerk performance.

Chapter 5

Anti-Jerk Cruise Control

5.1 Introduction and Literature Review

Cruise Control (CC) system is an Advanced Driver Assistance Systems (ADAS), which regulates the speed of the vehicle in response to a desired speed profile. Unlike the anti-jerk traction controllers which regulate the motor torque in response to a driver pedal input, anti-jerk cruise controllers are designed to regulate torque in response to reference speed input. Since most anti-jerk traction controllers are on-off controllers which are only activated when slip exceeds the desired limits, the same design cannot be used for a cruise controller since sudden activation/deactivation could deteriorate the performance of the cruise controller.

Cruise control design for Electric Vehicles (EVs) has gained particular attention in recent years. According to a survey report by National Highway Traffic Safety Administration (NHTSA) in 2015 [23], high jerk in cruise controllers are preventing users from using cruise-control; NHTSA since formed an important criteria for evaluating the performance of CC systems.

A Cruise controller typically comprises of an upper-level controller and a lower-level controller. The upper-level controller determines the vehicle kinematics such as regulating the vehicle speed in speed following mode or a desired distance from a preceding vehicle in distance following mode. The lower level controller [78] is a longitudinal dynamics controller which regulates the torque demand to achieve the desired acceleration or deceleration demanded by the upper level controller. Further, the lower-level cruise controller is also responsible to regulate the acceleration during conditions when the rate of torque increase is high, or when the wheel slip is high due to road-friction conditions.

Over the years, several advances have been made to the design of the upper-level cruise controller with regards to comfort [24], fuel-economy [79, 80] and traffic flow smoothing [81, 82]. The design of upper-level cruise controllers has also been extended to Adaptive cruise control (ACC) [83, 84] to cater for maintaining a fixed distance with respect to the host vehicle, a ACC-Stop & Go [85, 86] to cater for maintaining inter-vehicular distance even in high traffic and low-speed situations. Cooperative ACC [87, 88] could help platoon vehicles to share sensor data thus enabling faster and safer travel by maintaining a shorter inter-vehicular distance.

The lower-level cruise control performs the function of a traction or longitudinal dynamics controller when the vehicle is in cruise control. It is generally assumed while developing the cruise controllers that the lower-level controller is designed well [89]. We could find little literature on anti-jerk design of lower-level cruise controllers. Therefore, there is a need to design an anti-jerk cruise controller which could regulate the motor torque to achieve low jerk. It is pertinent to mention that low-jerk can only be achieved by compromising the velocity tracking performance. In this chapter, we present a cruise controller that achieves the best velocity tracking performance while maintaining jerk in the desired range. Key goals of the anti-jerk Model Predictive cruise control (MPCC) are:

- Meet the multiple objectives of an upper-level and lower-level cruise controller simultaneously.
- Achieve anti-jerk control by including (a) halfshaft torsion control (b) wheel-slip constraints to ensure a smooth ride even on rapidly-changing road conditions.

The net result is that the controller should be able to track the desired velocity profile with minimum jerk while increasing road safety.

The design of the anti-jerk MPCC is discussed in Section 5.2. The performance assessment/tuning of the controller is presented in Section 5.3 and HIL experiments demonstrating the real-time implementation of the controller are described in Section 5.4.

5.2 Anti-jerk cruise control design

The aim of the anti-jerk MPCC is to regulate the motor torque input so as to achieve the desired velocity-tracking performance with minimum jerk by using future road information available from sources such as GPS, ITS, VTV and VTI. The overall control scheme of the

MPCC is shown in Fig. 5.1. The MPC control effort is aimed at regulating the input motor torque to serve the following multiple objectives:

- Track a reference velocity profile (v_{ref}) with minimum jerk by looking ahead into the future.
- Minimize the energy consumption by including input torque ($u = T_m$) in case of non-linear MPC and ($u = \Delta T_m$) for the linear MPC in the cost function.
- Control the halfshaft torsion by including the halfshaft torsion angle ($\frac{\theta_m}{gr} - \theta_w$) in the cost function.

The quadratic objective function applied to the linear and non-linear anti-jerk cruise control problem is:

$$\underset{u}{Min}\, J = \int_0^{N_p} \left[w_1(v - v_{ref})^2 + w_2(u^2) + w_3\left(\frac{\theta_m}{gr} - \theta_w\right)^2 \right] \tag{5.1}$$

where w_1, w_2 and w_3 are weighting factors for velocity tracking, input torque and halfshaft torsion respectively in the objective function. The optimal control problem is subjected to the following constraints:

$$\begin{cases} v > 0 \\ \omega_w > 0 \\ \lambda_{t_{\min}} \leq \lambda \leq \lambda_{t_{\max}} \\ T_{\min} \leq u \leq T_{\max} \end{cases} \tag{5.2}$$

5.2.1 Design of Linear MPCC

The linear MPC is designed such that it is adaptive to changes in road-friction conditions. The Recursive Least Square (RLS) estimator has been designed to estimate the longitudinal stiffness of the tire (C_x) and update the MPCC controller in real time. The overall control scheme of the MPCC is shown in Fig. 5.1.

A linear-time-varying (LTV) form of the model explained in section 3.3.1 is formulated to solve the linear MPCC as:

$$x_{k+1} = A(t)x_k + B(t)u_k + E(t)\bar{u}_k \tag{5.3a}$$

$$x_{u_{k+1}} = x_{u_k} + \Delta u_k \tag{5.3b}$$

$$y_k = C(t)x_k \tag{5.3c}$$

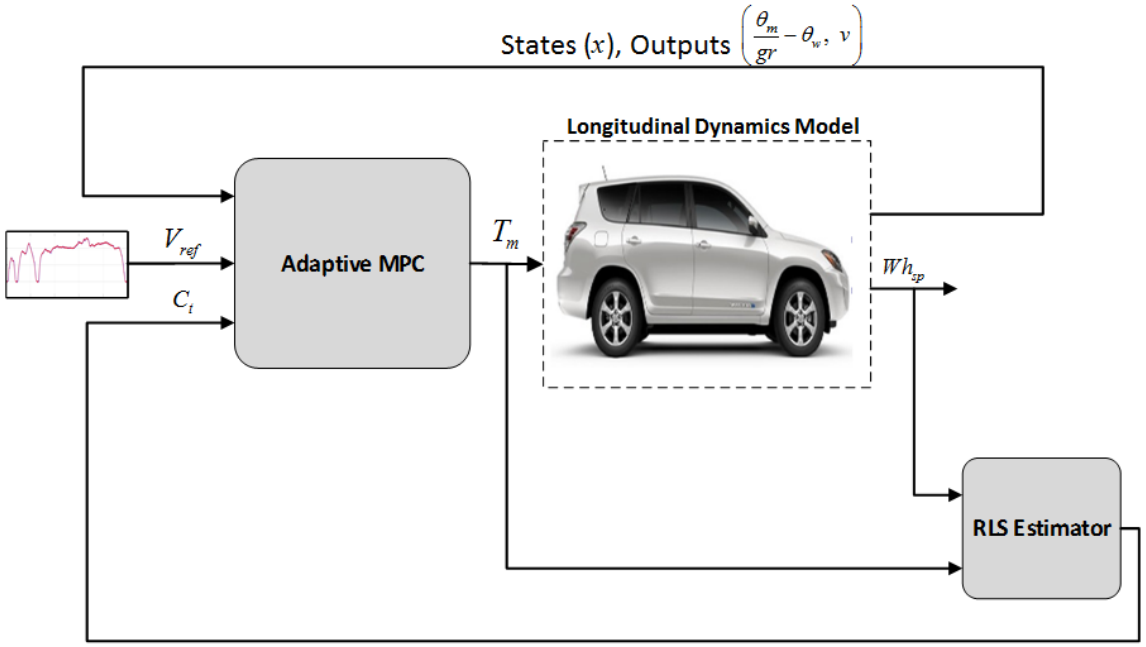

Figure 5.1: Control scheme of anti-jerk Model Predictive Adaptive Cruise Controller

with states: $x = \begin{bmatrix} \omega_m & \omega_w & v & \theta_m & \theta_w & \lambda_t \end{bmatrix}^T$, model input: $u = \Delta T_m$, and disturbance input $u_d = \begin{bmatrix} -F_r & -F_d & -v_{op} \end{bmatrix}^T$. The model outputs are $y = \begin{bmatrix} \frac{\theta_m}{gr} - \theta_w & v & Wh_{sp} \end{bmatrix}$.

The linear MPC problem is formulated as a quadratic programming (QP) problem as discussed in section 3.3.1 using the cost function in (5.1) and constraints in (5.2). The problem is solved using `'mpcqp'` solver in Matlab. The sampling period (Δt) for discretization has been chosen as 10 ms.

Since the MPC controller is designed to adapt to changes in road conditions, the maximum slip/torque constraints and slip stiffness applicable under various road friction conditions also need to be determined. In the following sections, the methodology used to calculate the torque/slip constraints, and design of the RLS estimator for the slip stiffness in real time is discussed.

Maximum motor torque and slip

To find the maximum motor torque and slip constraints to be used by the adaptive MPC, we establish a relationship between motor torque and wheel slip as follows:
The relationship between motor torque T_m and wheel torque T_w for steady state speeds based

on equation (3.1) is:

$$\mathrm{T}_m = \frac{2\mathrm{T}_w}{gr} \tag{5.4}$$

For steady state conditions, the relation between wheel torque and longitudinal tire force represented by equation (3.3), while neglecting rolling resistance is given by:

$$T_w = r_{eff}\frac{F_x}{2} \tag{5.5}$$

From equation (5.4) and (5.5), where the longitudinal force F_x is represented by the Pacejka tire model (3.11), the relationship between motor torque T_m and wheel slip λ_t is given as:

$$\mathrm{T}_m \simeq \frac{2D\sin\left[C\tan^{-1}\left\{B\lambda_t - E\left(B\lambda_t - \tan^{-1}B\lambda_t\right)\right\}\right]F_{zf}r_{eff}}{gr} \tag{5.6}$$

Fig. 5.2 shows the plot of (T_m versus λ_t) obtained by fitting the Pacejka tire model. While the tire model for the dry-road was determined from experimental measurements as discussed in Section 2.4.1, the tire model for wet-road and snowy-road were approximated by scaling down results obtained on dry-road. The operating limits for maximum motor torque and longitudinal slip based on the operating curve the three road conditions are: $T_{m_{\max}} = [355, 190, 110]$ and $\lambda_{t_{max}} = [0.062, 0.057, 0.045]$.

The constraints for the three road conditions are specified just below the peak point of the torque/slip curves for the three road conditions as: $T_{m_{min/max}} = [\pm 350, \pm 180, \pm 100]\ Nm$ and $\lambda_{t_{min/max}} = [\pm 0.06, \pm 0.052, \pm 0.04]$.

Slip stiffness estimation and constraints specification

Since the longitudinal dynamics plant model is derived using the Pacejka tire model, whereas the MPC controller is designed based on a linear tire model, we derive the equivalent slip stiffness of tires in dry-road, wet and snowy road conditions for the linear tire model from the Pacejka tire model. The transient longitudinal stiffness (C_t) as represented in equation (3.14) is:

$$C_t = (BCD)F_{zf}Rl \tag{5.7}$$

For a linear tire model, the relation between motor torque (T_m) and wheel slip (λ) is calculated from equation (5.4), (5.5), and (5.7) as:

$$\mathrm{T}_m \simeq \frac{2C_t\lambda_t r_{eff}}{gr} \tag{5.8}$$

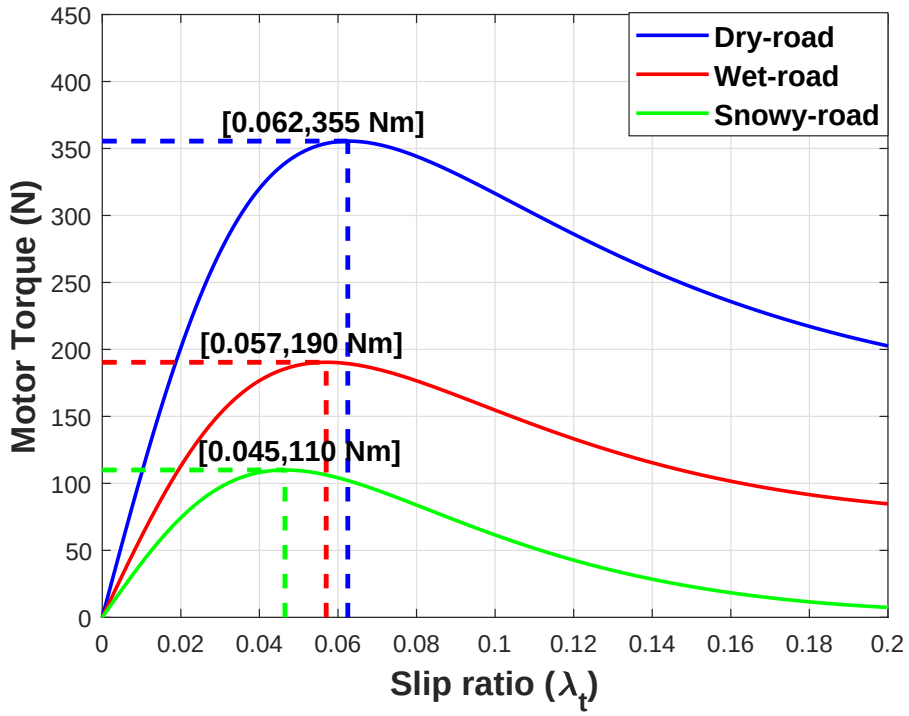

Figure 5.2: Plot showing maximum limit for torque and slip from the Pacejka tire model

The slope of the Pacejka tire curves and the equivalent linear tire model curves are plotted in Fig. 5.3. The equivalent longitudinal stiffness C_t calculated using (5.7) for dry-road condition, wet-road condition and snowy-road condition is $C_t = [163750, 99000, 90000]\ N$ per unit slip ratio.

The Adaptive MPC is designed such that it switches the torque and wheel slip constraints based on road condition, depending on C_t estimated by the RLS estimator. Since C_t could vary based on various factors such as types of tires, tire wear, pressure and temperature [90], the range for C_t defined for implementing the torque/slip constraints is shown in Table 5.1.

Recursive Least-Square (RLS) estimator

A slip-slope based tire-road estimation method [91, 92] has been used to estimate the longitudinal stiffness of the tire. Considering a linear tire model (with constant C_t and F_{zf}), the relationship between normalized traction force (μ) and slip (λ_t) is:

$$\frac{d\mu}{dt} = C_t \frac{d\lambda_t}{dt} \tag{5.9}$$

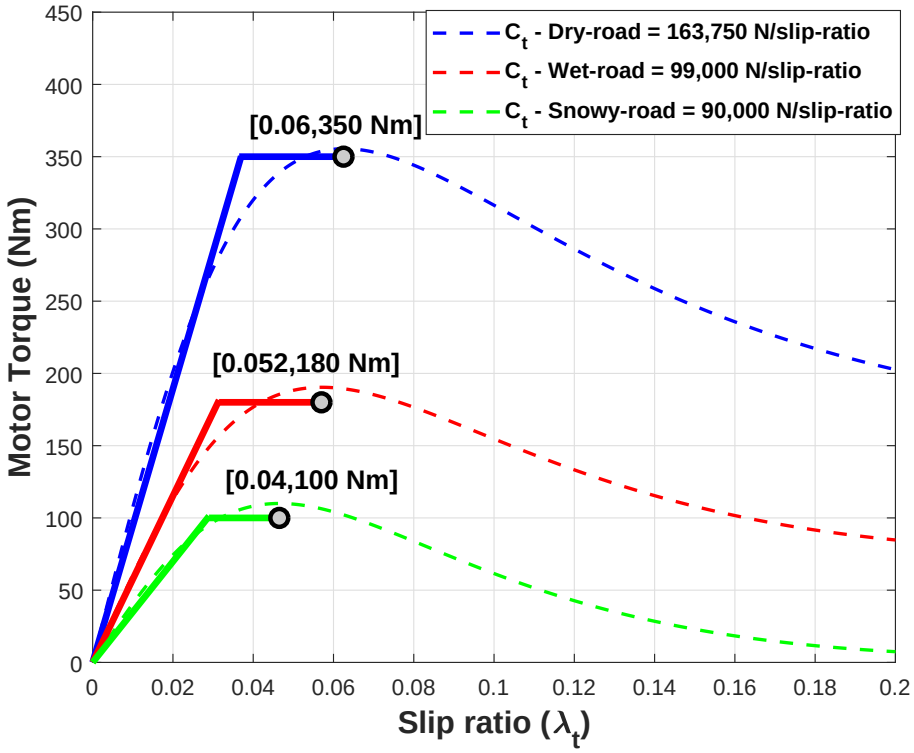

Figure 5.3: Motor torque operating lines corresponding to dry, wet and snowy roads

	Road friction condition	**Longitudinal Stiffness (C_t)**	**Motor Torque (T_m)**	**Slip ratio (λ_t)**
Units	–	**N/slip ratio**	**Nm**	–
Adaptive Constraints	Dry-road	$C_t \geq 130,000$	±350	±0.06
	Wet-road	$95,000N < C_t < 130,000N$	±180	±0.052
	Snowy-road	$C_t \leq 95,000$	±100	±0.04

Table 5.1: Adaptive constraints for various road friction conditions

A recursive least squares algorithm with forgetting factor [93] has been used to iteratively update the unknown parameter C_x from the regressor model:

$$\frac{Wh_{sp}(t)}{T_m(t)} = \frac{\phi(t)}{u(t)} = \frac{b_3 s^3 + b_2 s^2 + b_1 s + b_0}{s^5 + a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \tag{5.10}$$

where the coefficients $(a_4, a_3, a_2, a_1, a_0)$ and (b_3, b_2, b_1, b_0) of the transfer function are

determined from the model described in section 3.2.2. The problem can be formulated as follows:

$$y(t) = \theta(t)\phi(t) \tag{5.11}$$

where $\theta(t)$ is the estimated parameter $C_t(t)$, $y(t)$ is the measured Wh_{sp} signal and $\phi(t)$ is determined from the regression model given in equation (5.10). The procedure involves calculating the regression vector $\phi(t)$, calculating the identification error:

$$\epsilon(t) = y(t) - \phi^T\theta(t-1) \tag{5.12}$$

which is the difference between the actual plant output and that of the predicted model. Then, calculate the update gain vector given by:

$$K(t) = \frac{P(t-1)\phi(t)}{\sigma_f + \phi^T(t)P(t-1)\phi(t)} \tag{5.13}$$

and covariance matrix given by:

$$P(t) = \frac{1}{\sigma_f}\left[P(t-1) - \frac{P(t-1)\phi(t)\phi^T(t)P(t-1)}{\sigma_f + \phi^T(t)P(t-1)\phi(t)}\right] \tag{5.14}$$

The parameter estimate for $\theta(t)$ is then updated as:

$$\theta(t) = \theta(t-1) + K(t)\epsilon(t) \tag{5.15}$$

where the forgetting factor σ_f has been chosen as 0.96 to reduce the influence of old data, and track changes quickly.

5.2.2 Design of Non-linear MPCC

The NMPC control problem was formulated using the control-oriented model in Section 3.2.2 is written as:

$$\dot{x} = f(x, u, u_d) \tag{5.16}$$

with states: $x = \begin{bmatrix} \omega_m & \omega_w & v & \theta_m & \theta_w & \lambda_t \end{bmatrix}^T$, model input: $u = T_m$, and disturbance input $u_d = \begin{bmatrix} -F_r & -F_d \end{bmatrix}^T$. The model outputs are $y = \begin{bmatrix} \frac{\theta_m}{gr} - \theta_w & v \end{bmatrix}$. A schematic diagram explaining the implementation of the NMPC problem is shown in Fig. 5.4.

The NMPC problem has been solved using the cost function and constraints defined

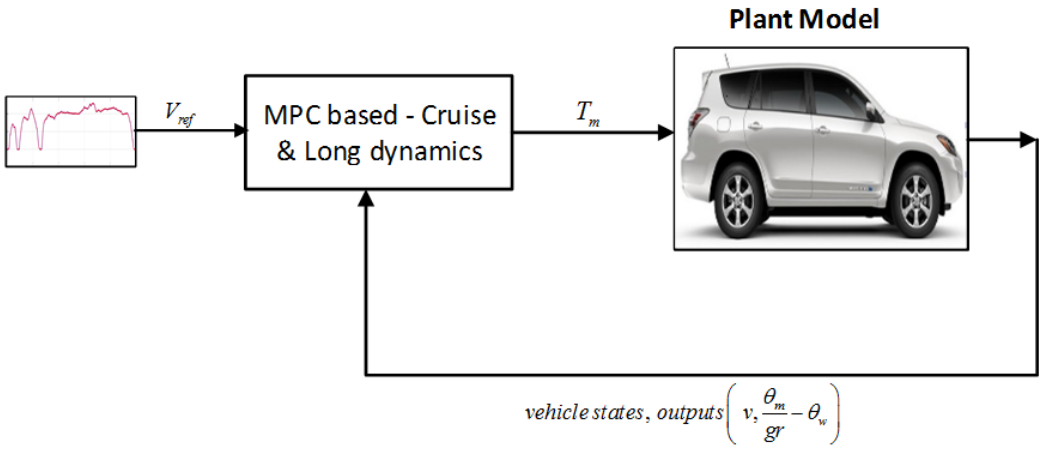

Figure 5.4: Implementation scheme on non-linear model predictive cruise control

in Section 5.2. The problem was solved using (a) SQP solver `'fmincon'` function in Matlab/Simulink [61] and (b) A self-coded Single Shooting NMPC solver developed by Maitland et al. [62] as explained in section 3.3.2. Since `'fmincon'` function does not support C-code generation, HIL implementation was undertaken using the Single Shooting NMPC solver.

5.3 Performance assessment

The objective of the anti-jerk MPCC is to objectively demonstrate velocity-tracking ability with minimum jerk. The assessment criterion defined in Section 3.4 has been used to evaluate the performance of the linear and non-linear anti-jerk MPCC.

5.3.1 Linear MPCC

The performance of the anti-jerk MPCC has been assessed by comparing the performance of the linear MPCC over the US06 and UDDS drive cycles. The following section explains the tuning of the controllers and compares the performance of adaptive and non-adaptive controllers for varying road conditions.

Tuning of Linear MPCC

We would like to tune the controller so as to maintain velocity tracking error as low as possible while keeping maximum jerk below 2-3 m/s^3. Tuning the weighting factor w_1 helps

achieve a good tracking performance, weighting factor w_2 helps decrease jerk by controlling the halfshaft torsion during sudden changes in acceleration, and weighting factor w_3 helps make the input torque profile smoother while achieving the control action using minimum energy.

The choice of prediction and control horizon is generally a trade off between performance and computational time. Increase in prediction horizon (N_p) corresponds to an increase in the knowledge of the future trip, while increase in control horizon (N_c) is proportional to the ECU control effort. A larger prediction and control window results in a more optimal control action but makes real-time implementation a challenge. In this case, we have tuned the controller for the highest accuracy and then verified that it can be implemented in real time through HIL testing.

A four-step process to tune the controller for a US06 drive cycle is illustrated below:

- **Step 1**: The weighting factors w_1 and w_2 are chosen with the aim of achieving a good tracking performance. This is achieved by setting the weighting factors $w_1 = 150$, $w_2 = 150$, $w_3 = 0$. The prediction horizon is selected as $N_p = 10$ (0.1 s) and control horizon as $N_c = 3$ (0.03 s).

- **Step 2**: The tuning parameters are selected as in Step 1, while the prediction horizon is increased in steps from 10 to 100.

	Prediction horizon	**RMS VTE**	**Max Jerk**	**IAJ**
Units	**-**	m/s	m/s^3	m/s^3
Tuning for Anti-jerk (N_p)	10	1.95	10.4	185.53
	30	1.4	6.4	148.95
	50	0.58	4.8	142.23
	60	0.38	4.4	137.82
	70	**0.27**	**3.5**	**133.42**
	80	0.18	2.95	127.8
	100	0.07	2.82	113.25

Table 5.2: Anti-jerk performance of linear MPCC with N_p ranging from 10 to 100 steps

Table 5.2 shows that as prediction horizon increases, the RMS-VTE decreases indicating a good velocity tracking performance. The maximum jerk and IAJ also decrease with

increase in prediction horizon. However, based on HIL experiments discussed in Section 5.4.1, we select $N_p = 70$ as the tuning parameter so that the turnaround time is less than 7 ms. This is less than accepted limit of 10 ms for real time performance discussed in section 3.4.2.

- **Step 3**: The tuning parameters are selected as $w_1 = 150$, $w_2 = 150$, $w_3 = 0$, $N_p = 70$ and N_c is increased in steps from 3 to 10.

	Control horizon	**RMS VTE**	**Max Jerk**	**IAJ**
Units	**-**	m/s	m/s^3	m/s^3
Tuning for Anti-jerk (N_c)	3	0.27	3.54	133.42
	5	0.27	3.53	133.41
	7	0.27	3.52	133.09
	10	0.27	3.51	132.85

Table 5.3: Anti-jerk performance of linear MPCC with N_c ranging from 3 to 10 steps.

Fig. 5.3 shows that as N_c increases, there is only a marginal improvement in the velocity tracking and jerk performance. However, the turn-around time for implementation in HIL also increases. To keep the turn-around time below 7 ms as discussed in Section 5.4.1, $N_c = 3$ has been selected as the tuning parameter.

- **Step 4**: The tuning parameters are selected as $w_1 = 150$, $w_2 = 150$, $N_p = 70$, $N_c = 3$ and weighting factor w_3 is increased in steps from 0 to 240,000. Table 5.4 shows that as w_3 is increased, the RMS-VTE increases, while the jerk and IAJ decrease. Keeping a good balance between velocity tracking performance ($\leq$ 0.5–0.6 m/s) and drivability ($\leq 2\ m/s^3$), we have selected $w_3 = 180,000$ as the final tuning parameter.

The performance of linear MPCC has been tested on a US06 and UDDS drive cycle. Fig. 5.5 shows the US06 drive cycle can be tracked with a maximum error of $\pm 1.8\ Km/hr$, a maximum jerk of 1.96 m/s^3 while maintaining the desired motor torque within specified limits ie- Motor torque within $\pm 350\ Nm$ and slip ratio within $\pm$ 0.06. The performance of the linear MPCC has also been validated over 500 s of a UDDS drive cycle. Fig. 5.6 shows that the UDDS drive cycle can be tracked with a maximum error of $\pm 1.7\ Km/hr$ and a maximum jerk of 1.15 m/s^3.

	Weighting (w_3)	**RMS VTE**	**Max Jerk**	**IAJ**
Units	**-**	m/s	m/s^3	m/s^3
Tuning for Anti-jerk (w_3)	0	0.27	3.54	133.42
	45000	0.28	2.56	110.61
	100,000	0.37	2.26	93.70
	140,000	0.41	2.12	88.89
	180,000	**0.51**	**1.96**	**81.09**
	240,000	0.72	1.72	69.44

Table 5.4: Anti-jerk performance of linear MPCC with w_3 ranging from 0 to $100,000$

Adaptive performance of linear MPCC

The performance of the adaptive and non-adaptive controllers, both tuned with the same input and output weights, has been compared when road changes from dry-asphalt road to snowy condition after about 40 s of a US06 drive-cycle. In the adaptive MPC, the road condition is diagnosed with the help of the parameter estimator. As seen in Fig. 5.7, the longitudinal stiffness C_t changes from $163,750$ N on dry-asphalt road to $82,000$ N on snowy road between 40 and 47 s. It can be seen in Fig. 5.8 that the velocity tracking performance of the adaptive MPC is marginally lower as compared to non-adaptive MPC due to constraints on the maximum torque and slip (shown in dotted lines). However, anti-jerk performance of the adaptive MPC is much better, with less slip and torque oscillations after 40 s. The adaptive MPC avoids slipping in low-friction conditions by constraining the motor torque and wheel slip.

5.3.2 Non linear MPCC

The following section explains the methodology used for tuning and performance assessment of the NMPCC controller over US06 and UDDS drive cycles. The importance of including slip-based constraints in the NMPCC has been also demonstrated.

Tuning of NMPCC

The four-step process to tune the NMPCC for a US06 drive cycle is illustrated below:

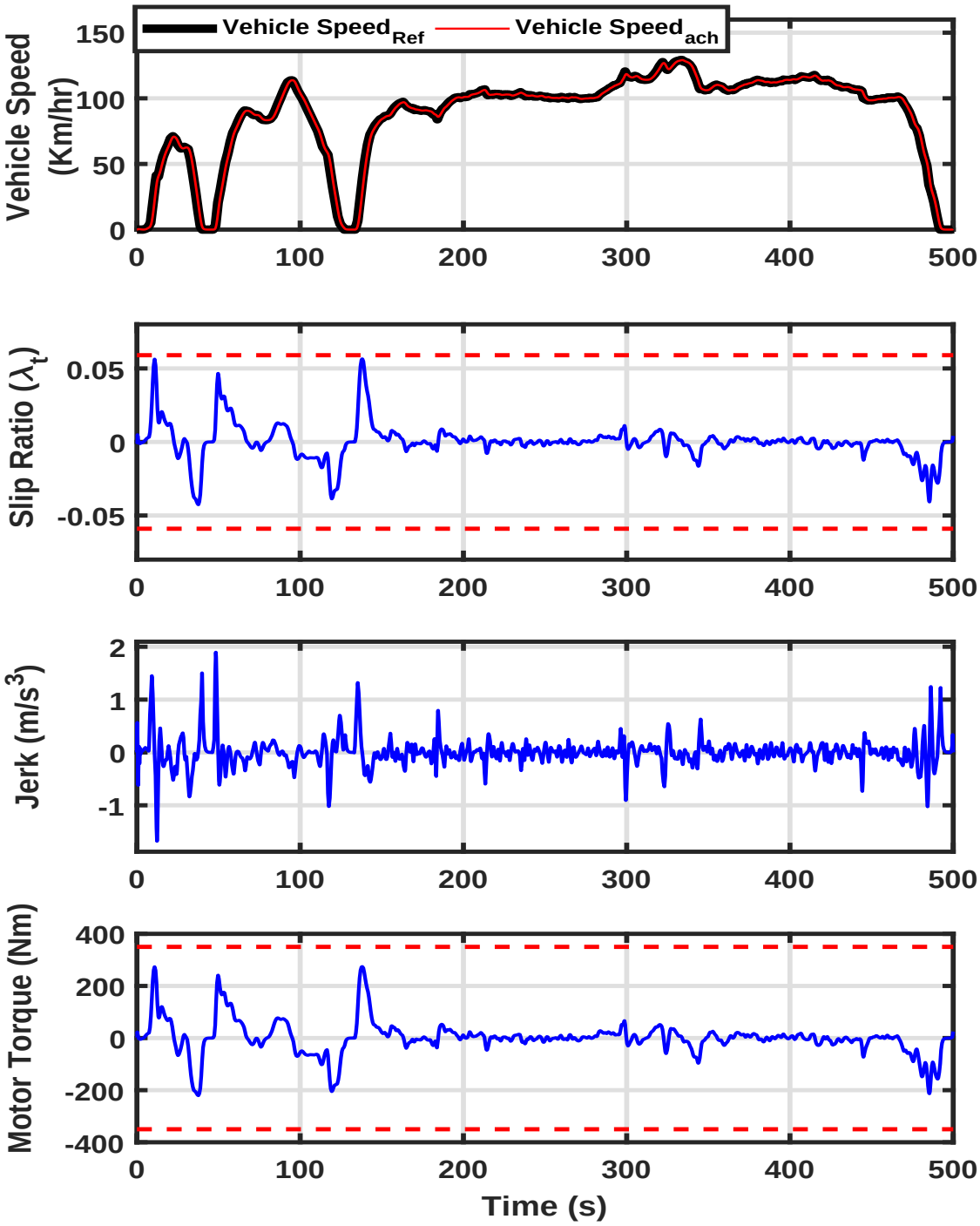

Figure 5.5: Plot showing anti-jerk performance of linear MPCC following a US06 drive cycle.

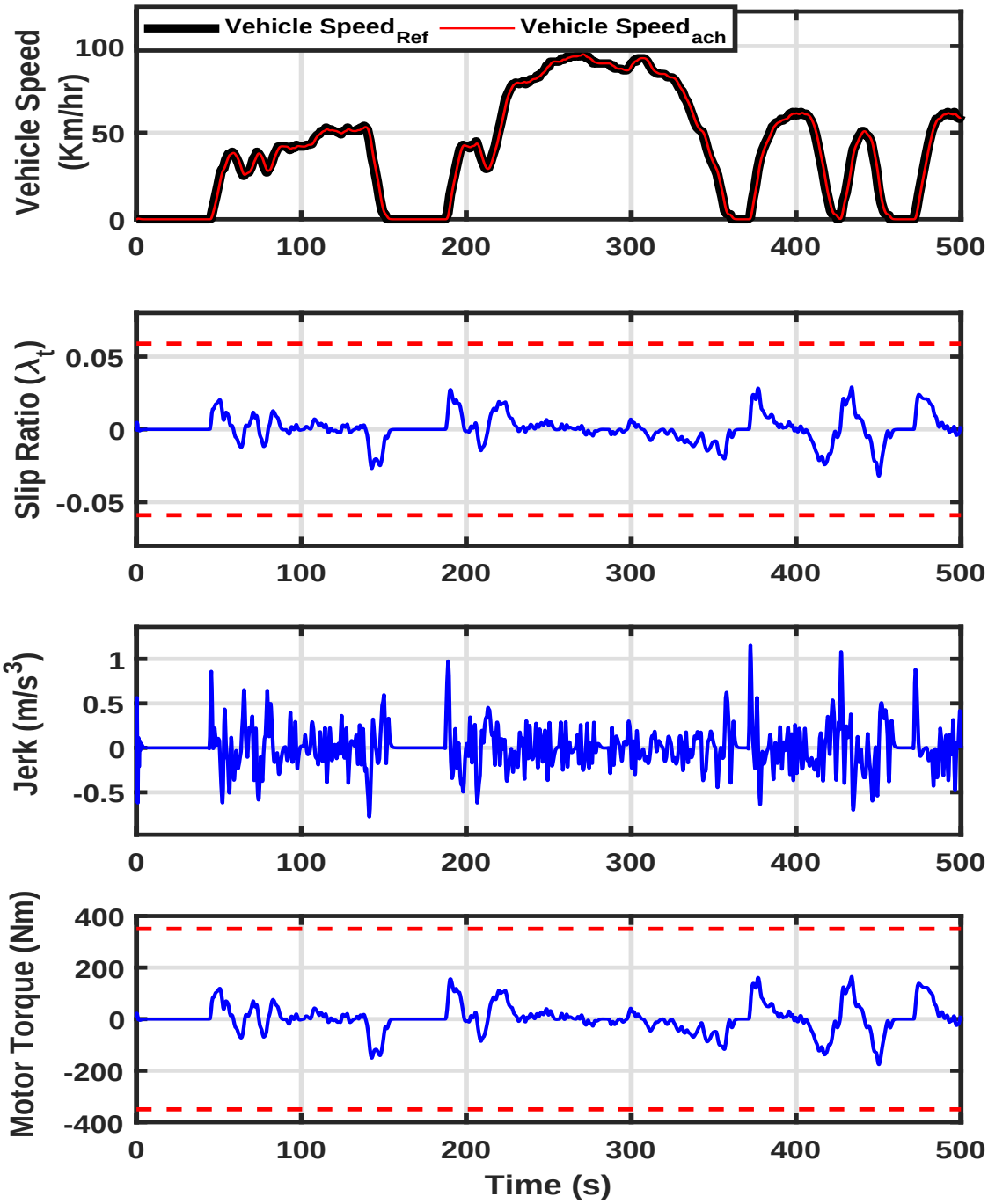

Figure 5.6: Plot showing anti-jerk performance of linear MPCC following a UDDS drive cycle.

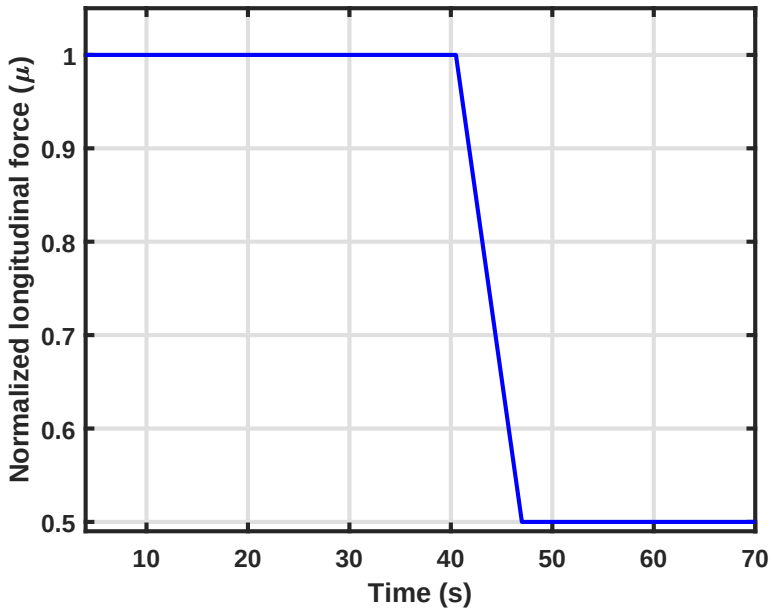

Figure 5.7: Plot showing road friction changing to snowy-road between 40-47 s.

- **Step 1**: The weighting factors w_1 and w_2 are chosen with the aim of achieving a good tracking performance. This is achieved by setting the weighting factors $w_1 = 400$, $w_2 = 0.04$, $w_3 = 0$. The prediction horizon is selected as ($N_p = 10$) and control horizon as ($N_c = 3$).

- **Step 2**: The tuning parameters are selected as in Step 1, while the prediction horizon is increased in steps from 10 to 70. Table 5.5 shows that as prediction horizon increases, the RMS-VTE decreases, indicating a good velocity tracking performance. However, the maximum jerk and IAJ increase as we increase the prediction horizon. As discussed in Section 5.4.2, a maximum CPU turn-around time of 0.18 ms is recorded during HIL experiments when increasing prediction horizon. We select $N_p = 27$ as the tuning parameter to keep the maximum jerk below 3 m/s^3.

- **Step 3**: Next, we tune for the optimal control horizon by setting tuning parameters as $w_1 = 400$,$w_2 = 0.04$, $w_3 = 0$ and $N_p = 27$ while increasing N_c from 3 to 10. Table 5.6 shows that as N_c increases, the velocity tracking as well as jerk performance improves upto $N_c = 7$. As discussed in Section 5.4.2, a maximum CPU turn-around time of 0.1 ms is recorded during HIL experiments when increasing control horizon upto $Nc = 10$. We have select $N_c = 7$ as the tuning parameter, which results in maximum RMS-VTE of 1.49 m/s and maximum jerk 2.13 m/s^3.

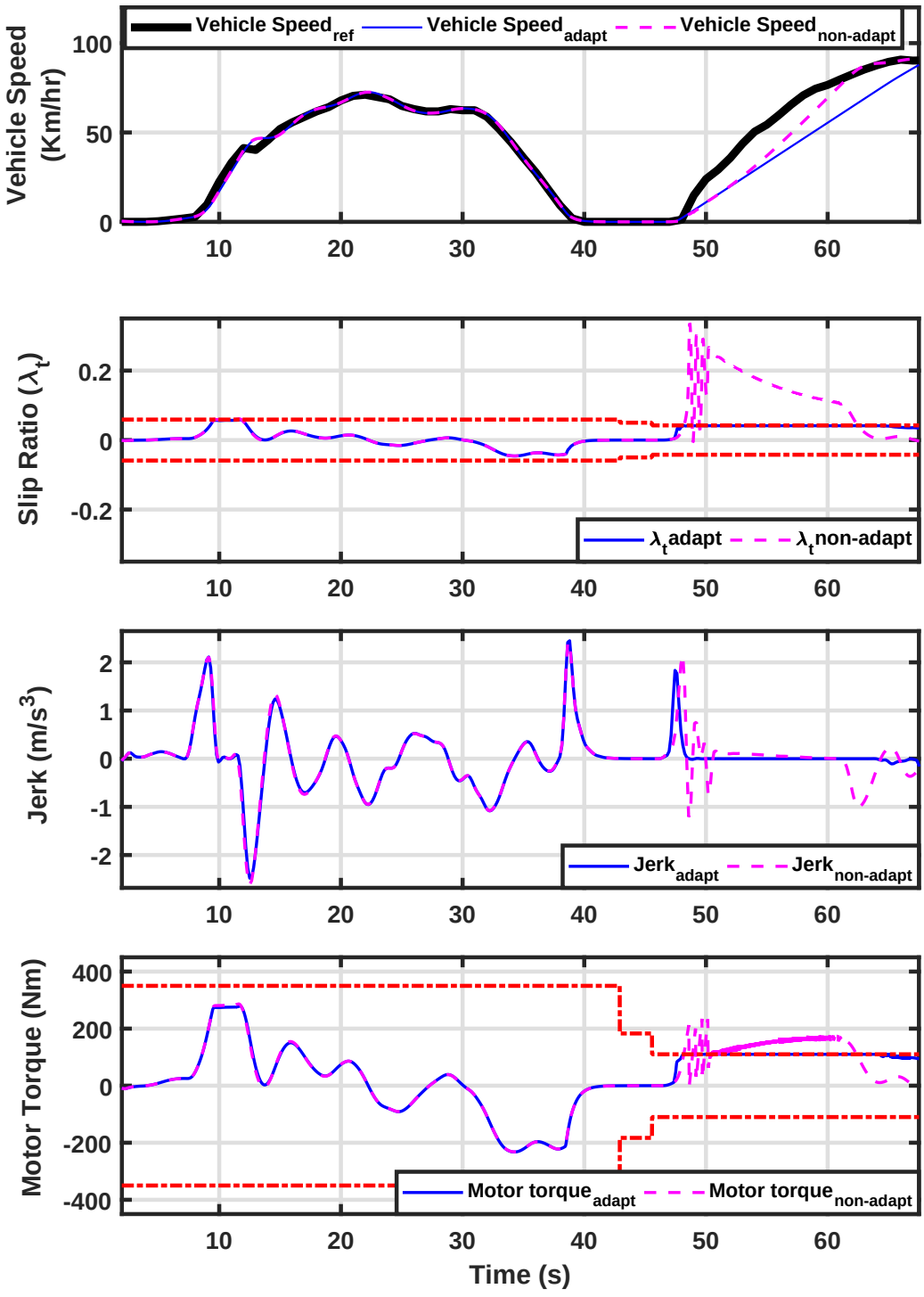

Figure 5.8: Plot showing comparison of adaptive and non-adaptive linear MPCC.

	Prediction horizon	RMS VTE	Max Jerk	IAJ
Units	-	m/s	m/s^3	m/s^3
Tuning for Anti-jerk (N_p)	10	5.77	1.06	6.53
	20	2.37	1.71	13.71
	25	1.74	2.52	15.74
	27	**1.63**	**2.45**	**15.72**
	30	1.36	3.3	17.26
	50	0.84	5.07	20.84
	70	0.80	5.60	21.50

Table 5.5: Anti-jerk performance of NMPCC with N_p ranging from 10 to 100 steps.

	Control horizon	RMS VTE	Max Jerk	IAJ
Units	-	m/s	m/s^3	m/s^3
Tuning for Anti-jerk (N_c)	3	1.63	2.45	15.72
	5	1.55	2.33	13.71
	7	**1.49**	**2.13**	**11.38**
	10	1.56	2.36	13.92

Table 5.6: Anti-jerk performance of NMPCC with N_c ranging from 3 to 10 steps.

- **Step 4**: Further tuning is progressed with $w_1 = 400$ and $w_2 = 0.04$, $N_p = 27$ and $N_c = 7$. The weighting factor w_3 is progressively increased from 0 to 75,000 and results are listed in Table 5.7. It shows that as w_3 is increased, the RMS-VTE, jerk and IAJ decreases up to $w_3 = 45000$ and then increase. Therefore, $w_3 = 45000$ has been selected as the final tuning parameter.

	Weighting (w_3)	RMS VTE	Max Jerk	IAJ
Units	-	m/s	m/s^3	m/s^3
Tuning for Anti-jerk (w_3)	0	1.49	2.13	11.38
	25000	1.44	2.01	11.12
	45000	**1.37**	**1.97**	**10.97**
	75000	1.62	2.12	12.02

Table 5.7: Increasing w_3 of NMPCC from 0 to 75000.

The performance of NMPCC has been tested for 500 s of the US06 and UDDS drive cycles. A plot of the tuned NMPCC to achieve a maximum jerk of 2.01 m/s^3 and a maximum error of ± 2 Km/hr for a US06 drive cycle is shown in Fig. 5.9. Fig. 5.10 shows that that the UDDS drive cycle can be tracked with a maximum error of ± 1.9 Km/hr and a maximum jerk of 0.7 m/s^3.

5.3.3 Performance of NMPCC with and without slip-based constraints

To illustrate the significance of slip-based constraints, the velocity-tracking and jerk performance of the MPCC controller during the first 50 s of a US06 drive-cycle is compared with and without slip-based constraints. It may be noted that all other constraints in (5.2) are still satisfied. The weighting factors for both conditions are selected as $w_1 = 400$, $w_2 = 0.0005$ and $w_3 = 0$ to draw an even comparison between them, while achieving a good velocity tracking performance. A summary of the simulation results are compiled in Table 5.8. The

	RMS VTE	**Max Jerk**	**IAJ**
Units	m/s	m/s^3	m/s^3
Without slip constraints	0.267	8.78	41.36
With slip constraints	0.178	6.69	32.88

Table 5.8: Performance of NMPCC with and without slip-based constraints

vehicle speeds, jerk, slip-ratio and motor torque achieved in the two conditions is compared in Fig. 5.11. It can be seen that if the slip-based constraints are not included in the control design, even though motor torque is constrained between $\pm$ 350 Nm, a high jerk (maximum 8.78 m/s^3) is observed due to the vehicle operating at a high wheel slip ratio of $0.12 - 0.15$ in the negatively-sloped region of the Pacejka tire model (Fig 5.2). However, by including the slip-based constraints in the control design, not only is the jerk reduced, but the velocity tracking performance is also improved. This is because the vehicle is forced to operate in the positively-sloped region of the tire model.

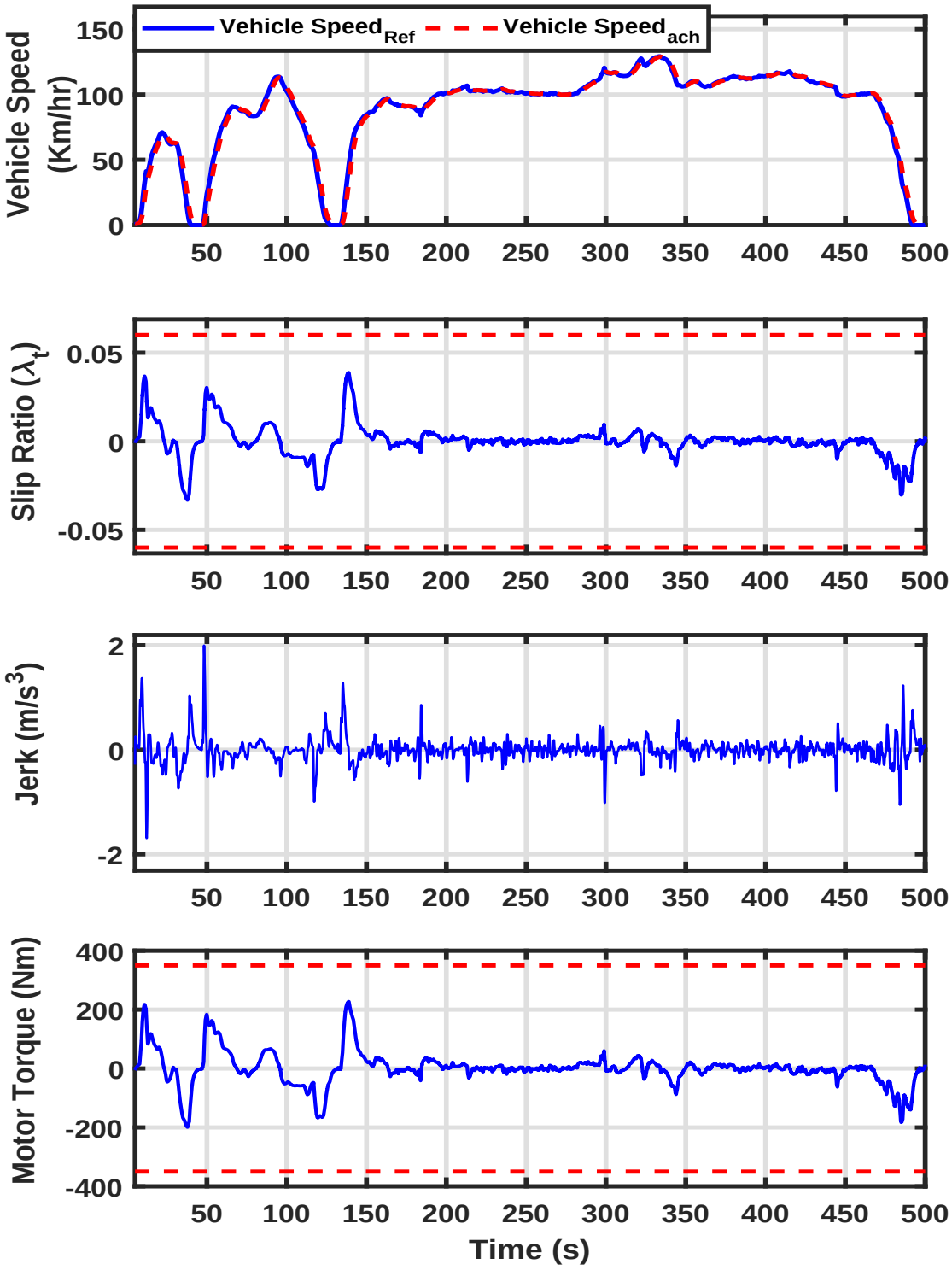

Figure 5.9: Plot showing anti-jerk performance of NMPCC following a US06 drive cycle.

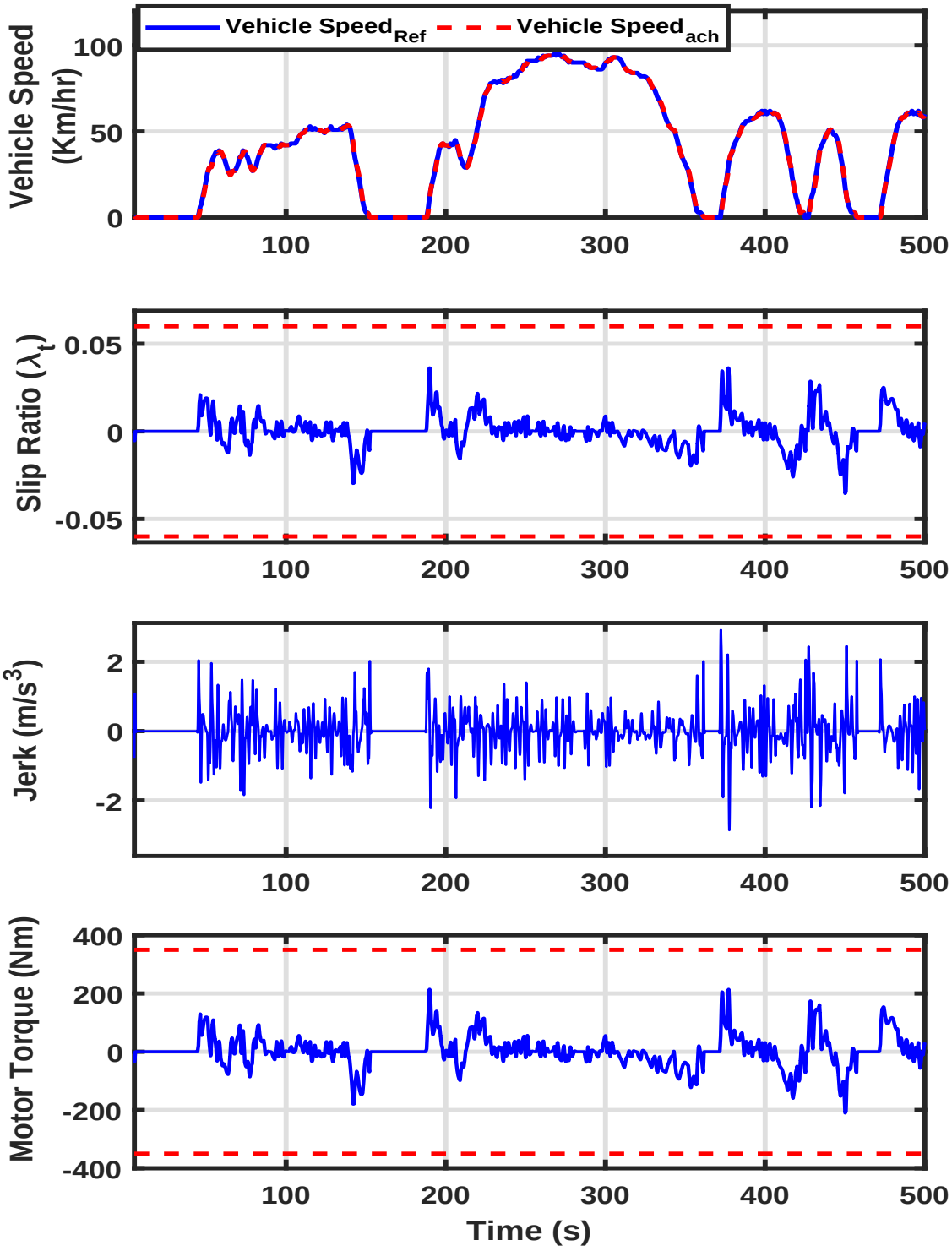

Figure 5.10: Plot showing anti-jerk performance of NMPCC following a UDDS drive cycle.

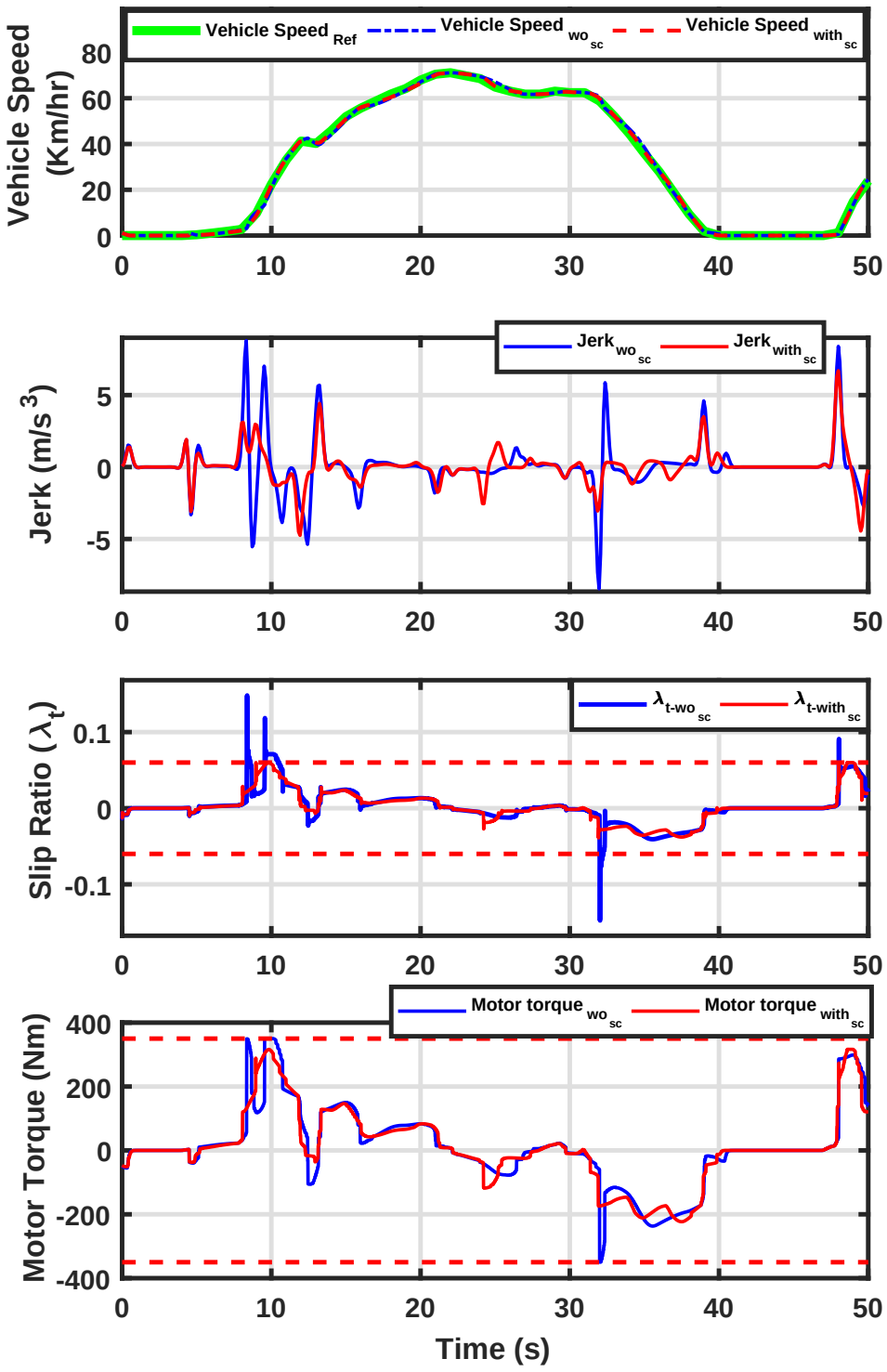

Figure 5.11: Plot showing velocity tracking performance with and without slip constraints for the first 50s of US06 drive cycle

5.4 Hardware-in-the-loop experiments

Hardware-in-the-loop (HIL) experiments are used to demonstrate real-time capability of the MPC controllers. A general description of the HIL setup and the procedure used for undertaking the HIL experiments is discussed in Section 3.4.2. The aim of the HIL experiments is to determine the optimal prediction and control horizons (N_p and N_c) that result in real-time capability of the controllers. The real-time performance of the controller is quantified by the turn-around time, which is the time taken for executing the controller code in the ECU. The generally accepted upper limit for turn-around time is 10 ms for real-time applications [63].

5.4.1 HIL experiments: Linear MPCC

The LMPCC, formulated as a quadratic programming (QP) problem (discussed in Section 5.3.1), was implemented on the HIL setup. The experiments were performed by tuning the controller with different N_p and N_c, such that similar end results are achieved with each setting.

It can be seen from Table 5.2 that as N_p increases from 10 to 80, the velocity tracking performance improves, while the jerk performance deteriorates. Real-time performance is not possible with $N_p > 80$. The simulation was run on a HIL setup with different N_p ranging from $10 - 80$ (with $N_c = 3$). The controller turn-around time of 0.39-8.46 ms has been recorded at various prediction horizons as seen in Fig. 5.12. However, real-time performance was not possible with $N_c > 3$ (turnaround time > 10 ms). A maximum turn-around time of 6.97 ms has been selected with $N_p = 70$ and $N_c = 3$ for demonstrating the real-time implementation with the linear MPCC.

5.4.2 HIL experiments: NMPCC

The NMPCC formulated using the Single Shooting NMPC solver (discussed in Section 5.3.1) was implemented on the HIL setup. It can seen in Table 5.5 that the velocity tracking improves as N_p increases from 10 to 70. However, jerk performance deteriorates with $N_p > 27$. The controller turn-around time of 0.039-0.17 ms has been recorded at various prediction horizons from 10 to 70 as seen in Fig. 5.13. Setting $N_p = 27$, a turn-around time ranging from 0.06-0.1 ms has been recorded while increasing N_c from 3 to 10. Fig. 5.14 shows the increase in turnaround times with increase in N_c. However, the velocity tracking

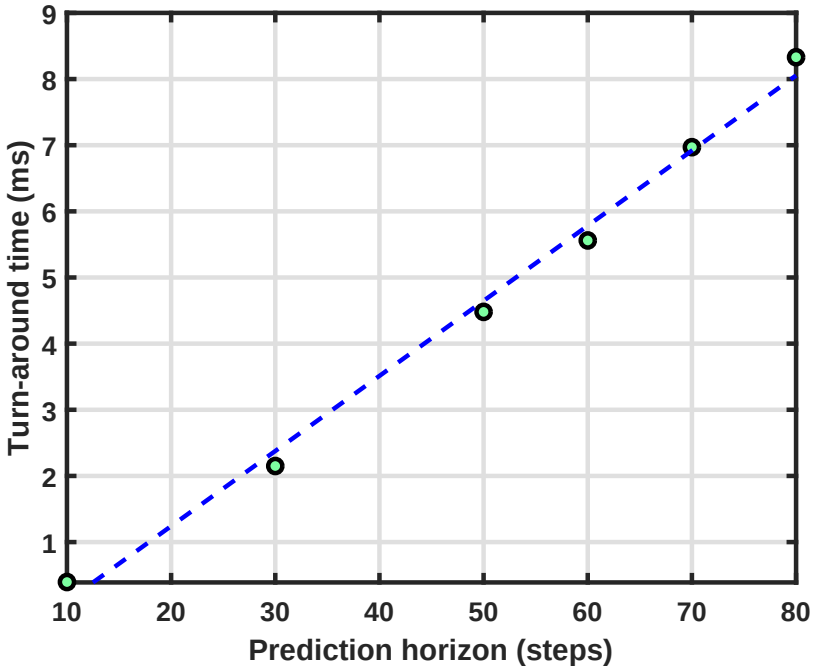

Figure 5.12: ECU maximum turn-around time for for linear MPCC with different lengths of prediction horizon

performance deteriorates after $N_c = 7$ as seen in Table 5.6. Therefore, $N_p = 27$ and $N_c = 3$ has been selected to achieve a maximum turn-around time of 0.082 ms with the NMPCC controller.

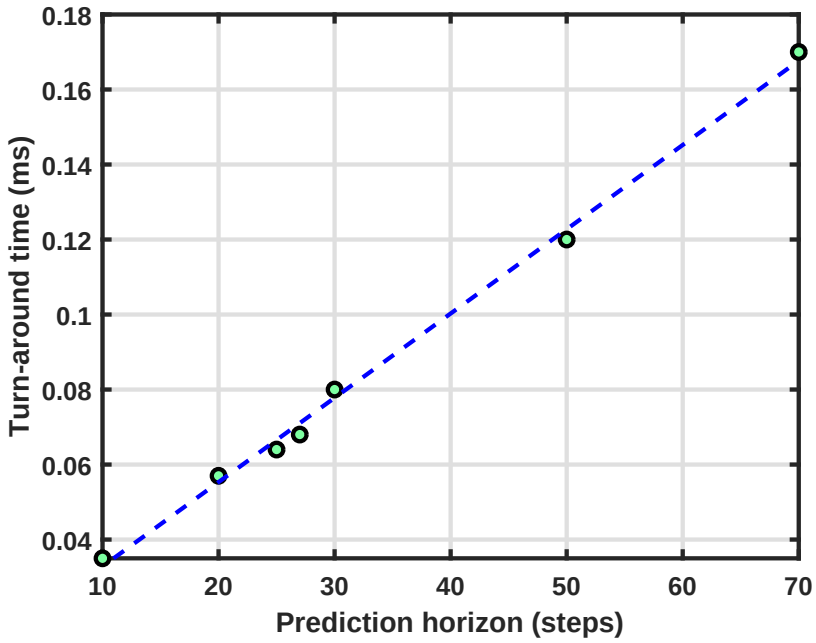

Figure 5.13: ECU maximum turn-around time for NMPCC with different lengths of prediction horizon

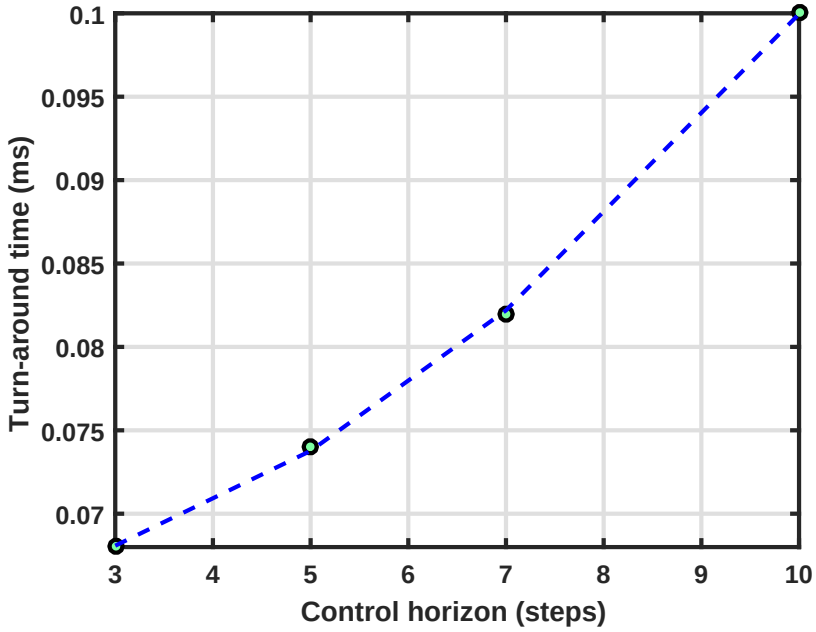

Figure 5.14: ECU maximum turn-around time for NMPCC with different lengths of control horizon, $N_p = 100$

5.5 Comparison of Linear MPCC and NMPCC

The overall performance of the linear MPCC and NMPCC for the US06 drive cycle and UDDS drive cycles is summarized in Table 5.9.

	Linear MPCC					**NMPCC**				
	RMS -VTE	**Jerk**	**Np**	**Nc**	**Ta time**	**RMS -VTE**	**Jerk**	**Np**	**Nc**	**Ta time**
Units	**Km/hr**	**m/s^3**	**steps**		**ms**	**Km/hr**	**m/s^3**	**steps**		**ms**
US06	± 1.83	1.96	70	3	6.97	± 2	2.01	27	7	0.082
UDDS	± 1.7	1.15				± 1.9	0.7			

Table 5.9: Comparison of overall performance: linear MPCC and NMPCC

It can be seen that a similar velocity tracking and jerk performance can be achieved by both linear MPCC and NMPCC. However, the computation or turn-around time (**Ta**) of the NMPCC problem is much faster than the linear MPCC. The difference in **Ta** is due to use of different solvers for the NMPCC and linear MPCC problems. While the NMPCC problem is solved using the Single-shooting NMPC solver [62], the linear MPCC problem is solved using the Matlab based 'mpcqp' solver. Since the main goal in this research is only to demonstrate the real-time capability of the developed controllers, we have not attempted to compare the computational efficiency the two solvers. Future work may involve comparing different MPC solvers for the anti-jerk MPCC problem.

5.6 Summary

Novel linear and non-linear MPC-based anti-jerk cruise controllers have been designed for central drivetrain architecture in EVs to meet multiple objectives of velocity tracking performance and enhanced drivability, while adapting to changes in road conditions. The significance of incorporating wheel-slip constraints in addition to halfshaft torsion in the control design to enhance anti-jerk performance has been demonstrated. A stiff set of system equations has been solved in the powertrain model, without increasing the discretization time step. The real-time capability of our design has been demonstrated on a HIL setup using two different solvers, NMPC solver for the NMPCC and Matlab based 'mpcqp' solver for the linear MPCC.

Chapter 6

Conclusions

In this thesis, anti-jerk longitudinal dynamic controllers have been designed for an electric vehicle with central drive powertrain using the model predictive control approach.

Road tests have been conducted on our test vehicle, the Toyota Rav4EV, to identify the parameters for the longitudinal dynamics model. Experimental data measured using an integrated measurement system designed to simultaneously collect data from the vehicle measurement system (VMS), global-positioning system (GPS), and inertial measurement unit (IMU) has been used for parameter identification. A two-stage parameter identification procedure has been followed to identify the parameters using different models. First, component level vehicle sub-system models have been considered to identify the individual model parameters. The parameters identified using this approach were thereafter optimized together using a full vehicle MapleSim model to obtain the final set of parameters. Adopting such a technique eliminates the need for laboratory-based parameter identification procedures.

A real-time implementable longitudinal dynamics plant model of an electric vehicle with flexible halfshafts was developed in the MapleSim software. The symbolic computation capability of MapleSim allows significant reduction in the number of equations and generates a highly optimized simulation code for real-time implementation. An equation based control-oriented model of the electric vehicle, of lower fidelity than the longitudinal dynamics plant model, has been developed. The plant and control-oriented models have been used to evaluate the control performance through model-in-the-loop (MIL) and hardware-in-the-loop (HIL) simulations.

The anti-jerk capability of the longitudinal dynamics controllers has been demonstrated in both traction as well as cruise-control applications. The MPC-based anti-jerk traction controller simultaneously meets the requirements of slip-tracking and enhanced drivability.

A comparison drawn between the proposed MPC-based anti-jerk traction controller and other controllers in the literature, such as the integral-action controller, integral-sliding-mode controller and maximum-torque-transmissible-estimation controller, demonstrates its superior performance.

The anti-jerk cruise controller has been designed to meet multiple objectives of velocity tracking performance and enhanced drivability, while adapting to changes in road conditions. The significance of incorporating wheel-slip constraints in addition to halfshaft torsion in the control design so as to enhance anti-jerk performance has been demonstrated. A stiff set of system equations has been solved in the powertrain model. Simulation results on a US06 drive cycle show that the reference velocity profile can be tracked with a maximum error of $\pm 2\ Km/hr$ and a maximum jerk of $2.01\ m/s^3$.

The designed controllers have been tested through HIL simulations using DSpace hardware. The plant model was loaded on the real-time computer (RTC), while the anti-jerk controllers were loaded on the electronic control unit (ECU). A Control Area Network (CAN) bus connected the ECU to the RTC. The real-time capability of the traction and cruise controllers has been demonstrated on a HIL setup using two different solvers, the Single Shooting NMPC solver based on symbolic computation with Maple, and a Matlab-based 'mpcqp' solver.

6.1 Summary of Contributions

The research contributions can be summarized as follows:

1. Developed a longitudinal dynamics model of an electric vehicle with a central drive powertrain which
   - is validated with experimental results, and
   - can be implemented in real-time.
2. Developed a novel control-oriented model of the electric powertrain by
   - including slip-based constraints in addition to halfshaft torque in the powertrain model. Adding slip-based constraints helps to control jerk during sudden changes in road-friction conditions while keeping the torque demand within limits.
   - solving a stiff system of ODEs as a result of including slip-based dynamics.

3. Demonstrated a model-based parameter identification technique based on road-tests, which

    - is simpler and faster than standardized laboratory tests. An innovative approach is used to connect a system of different sensors (IMU, GPS, VMS and vehicle CAN) together in a single bus.
    - eliminates the need for standardized testing, which generally involves the use of a number of different expensive indoor facilities.

4. Developed an MPC-based anti-jerk traction controller that

    - adds the anti-jerk function to traction control,
    - incorporates wheel-slip constraints to enable operation when wheel-slip is high, and
    - is implementable in real-time.

5. Developed a novel design for an MPC-based cruise controller, which

    - combines the upper and lower level cruise controllers in a single unit,
    - can be implemented in deteriorated road-friction conditions, and
    - is implementable in real-time.

6.2 Recommendations for Future Research

Although the performance of the MPC-based anti-jerk longitudinal dynamics controller has been demonstrated through MIL and HIL experiments, there is further scope for improvement in modeling, model validation and controller design as illustrated below:

6.2.1 Modeling and Validation

Motor model. A physics-based model of the induction motor based on field-oriented control could be included to represent the electric motor dynamics more accurately. A detailed parameter identification procedure based on no-load tests, locked rotor tests and back-emf tests will be required to estimate the motor parameters.

Parameter validation. The model parameters identified from road-tests may be validated with ASTM and other standardized tests procedures to prove the efficacy of the road-test procedures.

6.2.2 Development of controller

Tuning of Controllers. Tuning of the developed controllers is an iterative process of model-based design as discussed in Chapter 1. Although the controllers have been tuned through MIL and HIL testing, the developed anti-jerk traction and cruise controllers could be calibrated better using algorithms such as particle swan optimization [94] for meta-optimization to accurately estimate the parameters of the MPC. Further tuning could also be undertaken during component-in-the-loop (CIL) testing. The power-train test facility in the Green and Intelligent Automotive (GAIA) Laboratory commissioned recently at University of Waterloo has the necessary infrastructure to test the controllers.

Fast MPC solvers. In this research, we have used NMPC solver and Matlab based 'mpcqp' solvers to solve the MPC problem in real-time. However, a comparative analysis in the performance of the solvers has not been done. Also, a number of fast MPC approaches (explicit MPC, general minimum residual) have been developed to enable real-time deployment of controllers and reduction in ECU computation time. A comparative analysis of different MPC solvers will help find the best solver for the problem.

Future trip-information. As discussed in Chapter 4, future-trip information through technologies such as GPS, ITS, real-time traffic data and V2V or V2I communication can improve the performance of MPC. It may be prudent to undertake CIL testing of traction and cruise controllers whilst proving additional future trip information to these controllers.

References

[1] Torque news magazine, URL http://www.torquenews.com/gallery/tesla-motor, accessed in Nov 2015, 2004.

[2] T. Lee, The V diagram representation of Model-based design for automotive control systems, URL http://www.eeweb.com/, accessed in Nov 2015, 2013.

[3] Lotus Engineering magazine, URL https://lotusproactive.wordpress.com/2012/10/04/full-acceleration/, accessed in Oct 2015, 2012.

[4] M. Guarnieri, Looking back to electric cars, in: History of Electro-technology Conference (HISTELCON), 2012 Third IEEE, IEEE, 1–6, 2012.

[5] www.cio.com, Electric cars: Their past present and future, information on Electric cars - Past, present and future, 2015.

[6] B. C. Chan, The state of the art of electric, hybrid, and fuel cell vehicles, Proceedings of the IEEE 95 (4) (2007) 704–718.

[7] J. Larminie, J. Lowry, Electric vehicle technology explained, John Wiley & Sons, 2004.

[8] B. Nykvist, M. Nilsson, Rapidly falling costs of battery packs for electric vehicles, Nature Climate Change 5 (4) (2015) 329–332.

[9] Global EV Outlook, URL https://www.iea.org/publications/freepublications/publication/GlobalEVOutlook2017.pdf, 2017.

[10] R. Cuenca, A. Vyas, L. Gaines, Evaluation of electric vehicle production and operating costs, Citeseer, 1999.

[11] J. Zhang, A. Amodio, B. A. Guvenc, G. Rizzoni, P. Pisu, Investigation Of Torque Security Problems in Electrified Vehicles, in: Dynamic Systems and Control Conference, 2015.

[12] M. Chandrashekar, G. Savage, S. Birkett, J. McPhee, Graph-Theoretic Modelling: Four Decades of Development, Technical Report: University of Waterloo, Canada, 1995.

[13] P. Fritzson, F. Cellier, D. Broman, 2nd International Workshop on Equation-Based Object-Oriented Languages and Tools .

[14] K. Koprubasi, Modeling and control of a hybrid-electric vehicle for drivability and fuel economy improvements, Ph.D. thesis, The Ohio State University, 2008.

[15] J. Fredriksson, H. Weiefors, B. Egardt, Powertrain control for active damping of driveline oscillations, Vehicle system dynamics 37 (5) (2002) 359–376.

[16] A. Sorniotti, The effect of half-shaft torsion dynamics on the performance of a traction control system for electric vehicles, Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering 26 (9) (2012) 1145–1159.

[17] B. Wang, M. Xu, L. Yang, Study on the economic and environmental benefits of different EV powertrain topologies, Energy Conversion and Management 86 (2014) 916–926.

[18] N. Amann, J. Bocker, F. Prenner, Active damping of drive train oscillations for an electrically driven vehicle, IEEE/ASME Transactions on Mechatronics 9 (4) (2004) 697–700.

[19] F. U. Syed, M. L. Kuang, H. Ying, Active damping wheel-torque control system to reduce driveline oscillations in a power-split hybrid electric vehicle, IEEE Transactions on Vehicular Technology 58 (9) (2009) 4769–4785.

[20] A. P. Cann, A. W. Salmoni, P. Vi, T. R. Eger, An exploratory study of whole-body vibration exposure and dose while operating heavy equipment in the construction industry, Applied occupational and environmental hygiene 18 (12) (2003) 999–1005.

[21] H. O. List, P. Schoeggl, Objective evaluation of vehicle driveability, Tech. Rep. 980204, SAE Technical Paper, 1998.

[22] J. J. Momberg, Driveability evaluation for engine management calibration, Master's thesis, Stellenbosch: University of Stellenbosch, 2007.

[23] J. W Jenness, N. D. Lerner, S. Mazor, J. S. Osberg, C. B. Tefft, Evaluation of Adaptive Cruise Control Interface Requirements on the National Advanced Driving Simulator, 2015, National Highway Traffic safety Administration DOTHS 812172.

[24] J.-J. Martinez, C. Canudas-de Wit, A safe longitudinal control for adaptive cruise control and stop-and-go scenarios, IEEE Transactions on Control Systems Technology 15 (2) (2007) 246–258.

[25] X. Wei, G. Rizzoni, Objective metrics of fuel economy, performance and driveability-A review, Tech. Rep., SAE Technical Paper, 2004.

[26] J. Allen, Simulation and test systems for validation of electric drive and battery management systems, SAE Technical Paper (2012-01-2144).

[27] C. Schmitke, K. Morency, J. McPhee, Using graph theory and symbolic computing to generate efficient models for multi-body vehicle dynamics, Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics 222 (4) (2008) 339–352.

[28] Vehicle Test Facilities - IKA, RWTH, URL `https://www.ika.rwth-aachen.de/images/forschung/pruefstaende/pruefstandskatalog-en.pdf`, [Accessed on 15 Nov 2017], 2017.

[29] S. C. Shapiro, C. P. Dickerson, S. M. Arndt, M. W. Arndt, G. A. Mowry, Error analysis of center-of-gravity measurement techniques, Tech. Rep., SAE Technical Paper, 1995.

[30] A. K. Bhoopalam, K. Kefauver, Using Surface Texture Parameters to Relate Flat Belt Laboratory Traction Data to the Road, Tech. Rep., SAE Technical Paper, 2015.

[31] D. Lo Conte, Tyre parameter identification from road tests on a complete vehicle, Master's thesis, Delft University of Technology, Netherlands, 2010.

[32] E. Wilhelm, R. Bornatico, R. Widmer, L. Rodgers, G. S. Soh, Electric vehicle parameter identification, in: EVS26 International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium, Los Angeles, California, 1–10, 2012.

[33] Least-Squares (Model Fitting) Algorithms, Matlab documentation 2017, URL http://www.mathworks.com/help/optim/ug/least-squares-model-fitting-algorithms.html#broz0i4, 2017.

[34] H. B. Pacejka, E. Bakker, The magic formula tyre model, Vehicle System Dynamics 21 (S1) (1992) 1–18.

[35] G. Rill, Road vehicle dynamics: fundamentals and modeling, Crc Press, 2011.

[36] Y. Sam, J. Osman, M. Ghani, Active suspension control: Performance comparison using proportional integral sliding mode and linear quadratic regulator methods, in: Control Applications, 2003. CCA 2003. Proceedings of 2003 IEEE Conference on, vol. 1, IEEE, 274–278, 2003.

[37] M. A. Emam, A New Empirical Formula for Calculating Vehicles' Frontal Area, SAE Technical Paper (2011-01-0763).

[38] R. A. White, H. H. Korst, The determination of vehicle drag contributions from coast-down tests, SAE Technical Paper (720099).

[39] H. Donghao, Z. Changlu, H. Ying, A Reduced-Order Model for Active Suppression Control of Vehicle Longitudinal Low-Frequency Vibration, Journal of Shock and Vibration 1 (5731347) (2018) 22, URL https://doi.org/10.1155/2018/5731347.

[40] S. Orfanidis, Introduction to signal processing, Pearson Education, 2010.

[41] W. H. Kwon, S. H. Han, Receding horizon control: model predictive control for state models, Springer Science & Business Media, 2006.

[42] X. Lu, H. Chen, H. Zhang, P. Wang, B. Gao, Design of Model Predictive Controller for anti-jerk during tip-in/out process of vehicles, in: Proceedings of the 30th Chinese Control Conference, ISSN 1934-1768, 3395–3400, 2011.

[43] S. Terwen, M. Back, V. Krebs, Predictive powertrain control for heavy duty trucks, in: Modeling and Control of Economic Systems 2001:(SME 2001): a Proceedings Volume from the 10th IFAC Symposium, Klagenfurt, Austria, 6-8 September 2001, Elsevier, 105, 2003.

[44] S. Di Cairano, A. Bemporad, I. Kolmanovsky, D. Hrovat, Model predictive control of nonlinear mechatronic systems: An application to a magnetically actuated mass spring damper, IFAC Proceedings Volumes 39 (5) (2006) 241–246.

[45] D. D. Dunlap, E. G. Collins Jr, C. V. Caldwell, Sampling based model predictive control with

application to autonomous vehicle guidance, in: Florida Conference on Recent Advances in Robotics, 2008.

[46] S. F. Sulaiman, M. Rahmat, K. Osman, et al., Disturbance rejection using Model Predictive control for pneumatic actuator system, in: Signal Processing & Its Applications (CSPA), 2016 IEEE 12th International Colloquium on, IEEE, 285–290, 2016.

[47] A. Lagerberg, B. Egardt, Model predictive control of automotive powertrains with backlash, IFAC Proceedings Volumes 38 (1) (2005) 1–6.

[48] H. Kawamura, K. Ito, T. Karikomi, T. Kume, Highly-responsive acceleration control for the Nissan Leaf electric vehicle, Tech. Rep., SAE Technical Paper, 2011.

[49] K. Nam, Y. Hori, C. Lee, Wheel Slip Control for Improving Traction-Ability and Energy Efficiency of a Personal Electric Vehicle, Energies 8 (7) (2015) 6820–6840.

[50] L. Eriksson, L. Nielsen, Modeling and control of engines and drivelines, John Wiley & Sons, 2014.

[51] C. C. De Wit, P. Tsiotras, Dynamic tire friction models for vehicle traction control, in: Decision and Control, 1999. Proceedings of the 38th IEEE Conference on, vol. 4, IEEE, 3746–3751, 1999.

[52] J. Reimpell, M. Burckhardt, A. Preukschat, A. Zomotor, Fahrwerktechnik: Radschlupf-Regelsysteme: Reifenverhalten, Zeitabläufe, Messung des Drehzustands der Räder, Anti-Blockier-System (ABS)-Theorie, Hydraulikkreisläufe, Antriebs-Schlupf-Regelung (ASR)-Theorie, Hydraulikkreisläufe, elektronische Regeleinheiten, Leistungsgrenzen, ausgeführte Anti-Blockier-Systeme und Antriebs-Schlupf-Regelungen, Vogel, 1993.

[53] U. Kiencke, A. Daiss, Estimation of tyre friction for enhanced ABS-systems, Jsae Review 2 (16) (1995) 221.

[54] H. B. Pacejka, R. S. Sharp, Shear force development by pneumatic tyres in steady state conditions: a review of modelling aspects, Vehicle system dynamics 20 (3-4) (1991) 121–175.

[55] J. Maurice, H. Pacejka, Relaxation length behaviour of tyres, Vehicle System Dynamics 27 (S1) (1997) 339–342.

[56] V. V. Vantsevich, J. P. Gray, Relaxation Length Review and Time Constant Analysis for Agile Tire Dynamics Control, in: ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, DETC 2015-46798, American Society of Mechanical Engineers, 2015.

[57] C. Clover, J. Bernard, Longitudinal tire dynamics, Vehicle System Dynamics 29 (4) (1998) 231–260.

[58] J. M. Maciejowski, Predictive control: with constraints, Pearson education, 2002.

[59] L. Wang, Model predictive control system design and implementation using MATLAB®, Springer Science & Business Media, 2009.

[60] Mathworks, MATLAB solver 'mpcqp', https://www.mathworks.com/help/mpc/ug/qp-solver.html, [Accesed on Feb 23, 2017], URL https://www.mathworks.com/help/mpc/ug/qp-solver.html, 2017.

[61] Matlab, MATLAB solver 'Fmincon' https://www.mathworks.com/help/optim/ug/fmincon.html, [Accesed on Sept 20, 2017], 2017.

[62] A. Maitland, M. Batra, J. McPhee, Nonlinear Model Predictive Control Reduction Using Truncated Single Shooting, in: American Control Conference, IEEE, 2017, submitted.

[63] M. Vukov, W. Van Loock, B. Houska, H. J. Ferreau, J. Swevers, M. Diehl, Experimental validation of nonlinear MPC on an overhead crane using automatic code generation, in: American Control Conference (ACC), 2012, IEEE, 6264–6269, 2012.

[64] V. Ivanov, D. Savitski, B. Shyrokau, A Survey of Traction Control and Antilock Braking Systems of Full Electric Vehicles With Individually Controlled Electric Motors, IEEE Transactions on Vehicular Technology 64 (9) (2015) 3878–3896.

[65] H. Z. Li, L. Li, L. He, M. X. Kang, J. Song, L. Y. Yu, C. Wu, PID plus fuzzy logic method for torque control in traction control system, International Journal of Automotive Technology 13 (3) (2012) 441–450, ISSN 1976-3832, URL https://doi.org/10.1007/s12239-012-0041-4.

[66] H. Guo, R. Yu, W. Qiang, H. Chen, Optimal slip based traction control for electric vehicles using feedback linearization, in: 2014 International Conference on Mechatronics and Control (ICMC), 1159–1164, 2014.

[67] V. Utkin, J. Guldner, J. Shi, Sliding mode control in electro-mechanical systems, vol. 34, CRC press, 2009.

[68] J. Deur, D. Pavković, G. Burgio, D. Hrovat, A model-based traction control strategy non-reliant on wheel slip information, Vehicle system dynamics 49 (8) (2011) 1245–1265.

[69] Y. Hori, Future vehicle driven by electricity and control-research on four-wheel-motored "UOT Electric March II", IEEE Transactions on Industrial Electronics 51 (5) (2004) 954–962.

[70] J.-S. Hu, D. Yin, Y. Hori, F.-R. Hu, Electric vehicle traction control: a new MTTE methodology, IEEE Industry Applications Magazine 18 (2) (2012) 23–31.

[71] Z. Cai, C. Ma, Q. Zhao, Acceleration-to-torque ratio based anti-skid control for electric vehicles, in: Mechatronics and Embedded Systems and Applications (MESA), 2010 IEEE/ASME International Conference on, IEEE, 577–581, 2010.

[72] G. A. Magallan, C. H. De Angelo, G. O. Garcia, Maximization of the traction forces in a 2WD electric vehicle, IEEE Transactions on Vehicular Technology 60 (2) (2011) 369–380.

[73] J. Rodriguez, R. Meneses, J. Orus, Active vibration control for electric vehicle compliant drivetrains, in: Industrial Electronics Society, IECON 2013-39th Annual Conference of the IEEE, IEEE, 2590–2595, 2013.

[74] S. De Pinto, C. Chatzikomis, A. Sorniotti, G. Mantriota, Comparison of traction controllers for electric vehicles with on-board drivetrains, IEEE Transactions on Vehicular Technology 1 (99).

[75] D. Yin, S. Oh, Y. Hori, A novel traction control for EV based on maximum transmissible torque estimation, IEEE Transactions on Industrial Electronics 56 (6) (2009) 2086–2094.

[76] T. Goggia, A. Sorniotti, L. De Novellis, A. Ferrara, P. Gruber, J. Theunissen, D. Steenbeke, B. Knauder, J. Zehetner, Integral sliding mode for the torque-vectoring control of fully electric vehicles: Theoretical design and experimental assessment, IEEE Transactions on Vehicular Technology 64 (5) (2015) 1701–1715.

[77] S. Li, T. Kawabe, Slip suppression of electric vehicles using sliding mode control method, Scientific Research Publishing, 2013.

[78] S. Moon, I. Moon, K. Yi, Design, tuning, and evaluation of a full-range adaptive cruise control system with collision avoidance, Control Engineering Practice 17 (4) (2009) 442–455.

[79] B. Asadi, A. Vahidi, Predictive Cruise Control: Utilizing Upcoming Traffic Signal Information for Improving Fuel Economy and Reducing Trip Time, IEEE Transactions on Control Systems Technology 19 (3) (2011) 707–714, ISSN 1063-6536.

[80] B. Alrifaee, Y. Liu, D. Abel, ECO-cruise control using economic model predictive control, in: Control Applications (CCA), 2015 IEEE Conference on, IEEE, 1933–1938, 2015.

[81] A. Kesting, M. Treiber, M. Schönhof, D. Helbing, Adaptive cruise control design for active congestion avoidance, Transportation Research Part C: Emerging Technologies 16 (6) (2008) 668–683.

[82] A. Kesting, M. Treiber, M. Schönhof, F. Kranke, D. Helbing, Jam-avoiding adaptive cruise control (ACC) and its impact on traffic dynamics, in: Traffic and Granular Flow05, Springer, 633–643, 2007.

[83] R. K. Jurgen, Adaptive cruise control, Tech. Rep., SAE Technical Paper, 2006.

[84] R. Rajamani, Adaptive cruise control, Encyclopedia of Systems and Control (2015) 13–19.

[85] P. Venhovens, K. Naab, B. Adiprasito, Stop and go cruise control, International journal of automotive technology 1 (2) (2000) 61–69.

[86] G. Naus, R. Van Den Bleek, J. Ploeg, B. Scheepers, R. van de Molengraft, M. Steinbuch, Explicit MPC design and performance evaluation of an ACC Stop-&-Go, in: American Control Conference, 2008, IEEE, 224–229, 2008.

[87] B. Van Arem, C. J. Van Driel, R. Visser, The impact of cooperative adaptive cruise control on traffic-flow characteristics, IEEE Transactions on Intelligent Transportation Systems 7 (4) (2006) 429–436.

[88] D. De Bruin, J. Kroon, R. Van Klaveren, M. Nelisse, Design and test of a cooperative adaptive cruise control system, in: Intelligent Vehicles Symposium, 2004 IEEE, IEEE, 392–396, 2004.

[89] L.-h. Luo, H. Liu, P. Li, H. 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) (2010) 191–201.

[90] F. Gustafsson, Slip-based tire-road friction estimation, Automatica 33 (6) (1997) 1087–1099.

[91] D. Fodor, K. Enisz, R. Doman, P. Toth, Tire road friction coefficient estimation methods comparison based on different vehicle dynamics models, in: Vehicle Power and Propulsion Conference (VPPC), 2011 IEEE, IEEE, 1–4, 2011.

[92] C. Lee, K. Hedrick, K. Yi, Real-time slip-based estimation of maximum tire-road friction coefficient, IEEE/ASME Transactions on Mechatronics 9 (2) (2004) 454–458, ISSN 1083-4435.

[93] S. Sastry, M. Bodson, Adaptive control: stability, convergence and robustness, Courier Corporation, 2011.

[94] R. Suzuki, F. Kawai, H. Ito, C. Nakazawa, Y. Fukuyama, E. Aiyoshi, Automatic tuning of model predictive control using particle swarm optimization, in: Swarm Intelligence Symposium, 2007, IEEE, 221–226, 2007.

APPENDICES

A: Parameters for Plant Model

Parameter	Symbol	Value	Units
Weight of car	M	1750	kg
Wheel base	L	2.66	m
Frontal area	A_f	2.79	m^2
Parameters for longitudinal force			
Pacejka Tire Model	B, C, D, E	49,1.37,1.25,0.01	–
Suspension stiffness	K_f, K_r	54,370, 35,540	N/m
Tire vertical stiffness	K_t	253,000	N/m
Suspension damping	C_f, C_r	1980,1795	Ns/m
Pitch Inertia of car	I_y	3052	kgm^2
Position of CG	l_r, h	1.15, 0.62	m
Parameters for Aerodynamic drag and Rolling resistance			
Coeff of drag	C_d	0.382	–
Coeff of roll resistance	f_{rr}	0.0015	–
Inertial parameters			
Drivetrain inertia	J_d	0.423	kgm^2
Wheel inertia	J_w	4.7	kgm^2
Height of center of pressure	h_d	0.689	m
Halfshaft stiffness	k	21,600	Nm/rad
Halfshaft damping	c	200	Nm/rad/s

Table 1: Parameters of longitudinal dynamics model: Rav4EV

B: Parameters for Control-Oriented Model

Parameter	Symbol	Value	Units
Weight of car	M	1750	kg
Wheel base	L	2.66	m
Frontal area	A_f	2.79	m^2
Parameters for longitudinal force			
Pacejka Tire Model	B, C, D, E	49.04,1.018,1.101,0.001	–
Position of CG	l_r, h	1.48, 0.526	m
Normal force on each of front tires	F_{zf}	5500	N
Dynamic radius of tire	r_{eff}	0.357	m
Parameters for Aerodynamic drag and Rolling resistance			
Coeff of drag	C_d	0.4	–
Coeff of roll resistance	f_{rr}	0.002	–
Inertial parameters			
Drivetrain inertia	J_d	0.25	kgm^2
Wheel inertia	J_w	4	kgm^2
Halfshaft stiffness	k	21,600	Nm/rad
Halfshaft damping	c	200	Nm/rad/s

Table 2: Parameters of control-oriented model: Rav4EV