# Improving the Drivability of Electric Vehicles using Advanced Control and Estimation

David Hodgson

School of Electrical and Electronic Engineering

Newcastle University

A thesis submitted for the degree of

*Engineering Doctorate*

August 2014

# Abstract

The mechanical drivetrain dynamics of electric vehicles can have a detrimental effect on the driving experience and performance of the vehicle. Factors such as large gear backlashes, low damped axles and the relatively low inertia and friction of electric motors can lead to a drivetrain that is prone to oscillation and therefore giving a non-smooth acceleration response. When this is combined with unknown driving conditions, such as low traction surfaces and significant changes in the vehicle's mass, this can cause the vehicle's drivability to be poor, especially when closed loop speed control is required.

Generally for all industrial vehicles and many on-road vehicles the only feedback to the vehicle motor controller is from the encoder on the motor, as it is already required for the flux vector control of induction and permanent magnet AC motors. Although the motor speed can easily be converted into a vehicle speed after taking into account gear ratios and tyre radius, it is not valid to assume that the motor and vehicle speeds are proportionally equal during transient conditions where there can be significant differences between them due to flexibility in the drivetrain.

Electric vehicles offer improved efficiency over internal combustion engine (ICE) vehicles which can then be enhanced further through regenerative braking. Although this can lead to undesirable and dangerous conditions such as loss of traction during braking due to excessive regenerative braking, especially if the vehicle is rear wheel drive. Without any additional feedbacks it is traditionally assumed to be difficult to detect loss of traction or maximise the energy recovered through regenerative braking without risking wheel lock up.

It has been shown that vehicle drivability can be greatly improved if estimates of vehicle speed and mass are obtained. Using a fixed gain Kalman Filter (KF) for state estimation and a Recursive Least Squares (RLS) scheme for parameter estimation, both vehicle speed and mass have been estimated and used to improve the closed loop speed performance. The estimated values are also used for preventing loss of tyre traction with the road surface. Additionally a scheme for reducing oscillations without estimating vehicle speed for use in torque control mode applications is developed.

The Kalman Filter only works effectively when the correct process noise matrix **Q** and measurement noise matrix **R** are used, as well as the model being a reasonably accurate representation of the real system. Optimal tuning has been carried out using Genetic Algorithms (GA), with the estimation accuracy then analysed for robustness to varying vehicle mass.

Gearbox backlash is the most significant issue within the drivetrain, as it is often the initial cause of the oscillations during torque reversals; allowing the motor speed to accelerate much faster than that of the vehicle during the disconnect. A scheme that can reduce the delay when traversing the backlash during torque reversals but also decrease the impact speed to virtually zero has been created.

# Acknowledgements

Thank you to everyone that has helped and supported me during this project.

First of all I would like to thank Barrie Mecrow for encouraging me to do the EngD course and who along with my other supervisors: Howard Slater, Peter Barrass, Shady Gadoue and Damian Giaouris, have helped me greatly and guided me throughout this project. Without their help, advice and technical knowledge I would have achieved a lot less and on a number of occasions it was required for my research work to be steered in the right direction.

Thanks to EPSRC and Sevcon for the funding to carry out this project and to the numerous people at the university and Sevcon who have helped me with various aspects of my project and who have made working at both Sevcon and the university an enjoyable experience.

I would also like to thank Barrie and Howard for their careful proofreading of my first draft, and without their numerous English and grammatical corrections this thesis would not be a pleasant read.

And finally thank you to friends and family for being supportive and patient with me when I haven't had time to see them during my lengthy write up or when I was trying to solve some problem with my project, it is difficult to remember a time when I didn't have the work within this project weighing on my mind.

# Table of Contents

# List of Figures

# List of Tables

# List of Publications

## Conference Paper

D. Hodgson, B.C. Mecrow, S.M. Gadoue, H.J. Slater, P.G. Barrass, D. Giaouris, "Accurate Estimation of Electric Vehicle Speed Using Kalman Filtering in the Presence of Parameter Variation", 6th IET International Conference on Power Electronic, Machines and Drives (PEMD 2012), March 2012, pp. 1-6

## Journal Paper

D. Hodgson, B.C. Mecrow, S.M. Gadoue, H.J. Slater, P.G. Barrass, D. Giaouris, "The Effect of Vehicle Mass Changes on the Accuracy of Kalman Filter Estimation of Electric Vehicle Speed", IET Electrical Systems in Transportation, vol. 3, no. 3, pp. 67-78, Sep. 2013

## Patents

D. Hodgson, H. Slater, P. Shipley, "Motor controller with oscillation damping system", UK Patent GB2490493, Granted 7th Nov 2012

# Nomenclature

### Roman Symbols

| | |
|---|---|
| $a$ | Acceleration ($m\,s^{-2}$) |
| $A_v$ | Surface area of vehicle ($m^2$) |
| $c$ | Damping coefficient (rotational ($Nm\,rad^{-1}\,s^{-1}$) and linear ($N\,m^{-1}\,s^{-1}$)) |
| $C_d$ | Aerodynamic Drag coefficient |
| $C_r$ | Rolling Resistance coefficient |
| $d$ | Distance ($m$) |
| $Fx$ | Force in vehicle x axis ($N$) |
| $Fy$ | Force in vehicle y axis ($N$) |
| $Fz$ | Force in vehicle z axis ($N$) |
| $h$ | Height ($m$) |
| $h_{steady-state}$ | Height at steady state ($m$) |
| $J$ | Inertia ($kg\,m^2$) |
| $b$ | Frictional coefficient (rotational ($Nm\,rad^{-1}$)) |
| $k$ | Stiffness coefficient (rotational ($Nm\,rad^{-1}$) and linear ($N\,m^{-1}$) ) |
| $T_{stic}$ | Stiction torque ($Nm$) |
| $l$ | Length ($m$) |
| $m$ | Mass ($kg$) |
| $n$ | Gear Ratio |
| $P$ | Power ($kW$) |
| $r$ | Tyre Radius ($m$) |

| | |
|---|---|
| $T$ | Torque $(Nm)$ |
| $v$ | Velocity $(m\,s^{-1})$ |

## Greek Symbols

| | |
|---|---|
| $\alpha$ | Backlash size $(rad)$ |
| $\mu$ | Friction coefficient |
| $\omega$ | Rotational speed $(rad\,s^{-1})$ |
| $\rho$ | Air density |
| $\theta$ | Rotational angle $(rad)$ |

## Subscripts

| | |
|---|---|
| $_{a}$ | suspension acceleration force from changing vehicle C.O.G |
| $_{accel}$ | acceleration (remaining force or torque to accelerate the vehicle) |
| $_{bl}$ | backlash |
| $_{cog}$ | center of gravity |
| $_{ds}$ | shaft displacement |
| $_{f}$ | front half of vehicle |
| $_{fl}$ | front left corner of vehicle |
| $_{fr}$ | front right corner of vehicle |
| $_{g}$ | gearbox |
| $_{gi}$ | gearbox input |
| $_{go}$ | gearbox output |
| $_{m}$ | motor |
| $_{pk}$ | peak |
| $_{r}$ | rear half of vehicle |
| $_{rl}$ | rear left corner of vehicle |
| $_{road}$ | road |
| $_{rr}$ | rear right corner of vehicle |
| $_{s}$ | shaft/axle |

| | |
|---|---|
| $_{sn}$ | suspension |
| $_{sp}$ | sprung |
| $_{ty}$ | tyre |
| $_{usp}$ | unsprung |
| $_{v}$ | vehicle |
| $_{w}$ | wheel |

## Units

| | |
|---|---|
| $kg$ | Kilograms |
| $kHz$ | Kilohertz |
| $kW$ | Kilowatts |
| $m$ | Meters |
| $MHz$ | Megahertz |
| $m\,s^{-1}$ | Meters per second |
| $m\,s^{-2}$ | Meters per second squared |
| $N$ | Newtons |
| $Nm$ | Newton meters |
| $rad\,s^{-1}$ | Radians per second |
| $rad\,s^{-2}$ | Radians per second squared |
| $rpm$ | Rotations per minute |

## Other Symbols

| | |
|---|---|
| $sgn()$ | Sign function |

## Acronyms

| | |
|---|---|
| AB | Quadrature encoder |
| AWP | Aerial Work Platforms |
| CAN | Controller Area Network |
| DSP | Digital Signal Processor |
| EKF | Extended Kalman Filter |

| | |
|---|---|
| EMF | Electromotive Force |
| EngD | Engineering Doctorate |
| EPSRC | Engineering and Physical Sciences Research Council |
| EV | Electric Vehicle |
| FL | Fuzzy Logic |
| FLT | Fork Lift Truck |
| GA | Genetic Algorithm |
| GPS | Global Positioning System |
| ICE | Internal Combustion Engine |
| IC | Internal Combustion |
| KF | Kalman Filter |
| MPC | Model Predictive Control |
| PhD | Doctor of Philosophy |
| PMAC | Permanent Magnet Alternating Current |
| ppr | Pulse per revolution |
| PU | per-unit |
| RLS | Recursive Least Squares |
| SI | International System of Units |
| UKF | Unscented Kalman Filter |
| UVW | Hall effect encoder |

# Chapter 1

# Introduction

## 1.1 Engineering Doctorate

The Engineering Doctorate (EngD) is a program that primarily contains PhD level research carried out with an industrial sponsor. It also consists of a number of taught modules in the relevant field; in this case power electronics, drives and machines, and additionally a number of business modules. The scheme is promoted by the Engineering and Physical Sciences Research Council (EPSRC), who along with the sponsoring company, provide the funding for the research. Due to the industrial involvement, the research will be of commercial interest to the sponsoring company with the majority of the work carried out at the company, rather than at a university.

## 1.2 Electric Vehicles

A great number of the first powered on-road vehicles in the mid to late 19th century were electric, with steam power being the only real alternative. But in the early 20th century with the advancement of the internal combustion engine (ICE) and the availability of a suitable fuel, their popularity dropped as the energy storage of the battery could not match that of petrol, which allowed for a much greater driving range. Very few electric on-road vehicles were sold since then until recently in the last 25 years or so where there has been a large increase in on-road vehicles using a pure electric or hybrid drivetrain. This has mostly been motivated by environmental concerns of ICE vehicles, along with the increasing cost of oil. Other factors contributing to the increased interest in on-road electric vehicles (EV) are improved battery capacity, although the cost is still prohibitive; and more efficient motor technologies combined with better power electronics and processors for advanced control.

Although for the majority of the last century on-road EVs have only been used for a few niche applications such as milk floats, a number of industrial and recreational vehicles have still used an all-electric drivetrain through this time. Vehicles designed for indoor use such as forklift trucks (FLT) use electric propulsion along with utility vehicles and golf buggies. These vehicles generally use deep cycle lead acid batteries, which although have a low energy density, they are comparatively low cost when compared to Lithium-Ion batteries used in most on-road EV. The increased weight of the lead acid battery is not always an issue as a number of industrial vehicles such as FLT need a significant counterbalance for the vehicle to remain stable when the lift is raised.

## 1.3 Sevcon

Sevcon was the industrial partner for the research carried out in this thesis; they design and manufacture motor controllers for electric vehicles. The motor controllers are not just responsible for motor control, they also control everything associated with the vehicle such as: vehicle speed control/limit, digital and analogue input/output, CAN communications and battery monitoring and protection (voltage and current limit and state of charge). The focus of this work is at the vehicle control level between the pedal input and the motor torque demand.

The applications that the controllers are used on vary greatly. Some examples are: FLTs, aerial work platforms (AWP), airport tow tractors, mining haulers with a loaded mass of over 150 tonnes, golf buggies and utility vehicles, cars, motorbikes and scooters. Each application will have a unique drivetrain that the controller has to be able to deal with; where generally the vehicle parameters are unknown and the same control scheme is used for each. The mechanical aspects of the vehicle drivetrain have a sizeable impact on the vehicles response with all applications demanding a smooth driving response with minimal set up and configuration time.

## 1.4 Objectives

The word drivability is given the dictionary definition of *“the way a vehicle responds to whatever driving conditions it is exposed to”* [1]. It is commonly used to describe a vehicle’s driving response; its main use is to describe the smoothness of a vehicle’s acceleration. As the dictionary definition describes it can also be applied to how the vehicle responds to different driving conditions and how easy that makes the vehicle to use for the driver.

The aims of this research can be summarised as follows:

- Improved smoothness of vehicle drive, whilst still maintaining a fast response time - when operating in both torque and speed control mode the vehicle should be able to accelerate rapidly or change torque direction when entering braking without drivetrain oscillations making the vehicle drive feel non-smooth.

- Easier speed control gain tuning, whilst also improving the control accuracy - currently the speed control scheme can be difficult to tune and usually the most optimal gains chosen will have to be a compromise, still giving a response time below what is desired with some amount of overshoot and oscillations.

- Robustness to vehicle load changes - the ability for the vehicle driving feel to not be effected by changes in vehicle mass.

- Compensation of gearbox backlash - remove the sudden 'jerk' when the gear cogs impact.

- Preventing dangerous tyre slip that can lead to loss of control - detect and reduce the possibility of the tyres loosing traction due to large accelerating or braking torques being applied.

- Improved vehicle hill hold through estimating mass - determine significant load mass changes and adapt to them.

### 1.4.1 Vehicle Dynamics Consideration

EVs all require smooth progressive acceleration, no oscillations in the drivetrain and precise speed control for slow speed maneuvers. This response is expected, regardless of load changes and poor mechanical drivetrain components for example large gear backlash and low mechanical damping. During steady state, constant torque and load conditions, the motor and vehicle speeds are equal, after taking into account the gear ratio and tyre radius. Whenever the motor torque changes, the drivetrain mechanical dynamics allow the motor speed to initially change much more rapidly than the vehicle speed, which then develops into a damped oscillation of motor speed if the torque change is significant or the axle has low damping. This affects both the speed feedback and the smoothness of the vehicle's response. Speed feedback is often only available from the motor encoder, as most industrial vehicles do not have wheel speed sensors to give vehicle speed. The model of the drivetrain can be compared to that often found in some industrial systems such as rolling mills; where there is a low inertia motor, gearbox with backlash, flexible shaft and a large inertia load.

Industrial electric vehicles are not normally fitted with additional sensors such as speed sensors on the wheels as it will add an additional cost to the vehicle and some operate in harsh environments where reliability could be an issue.

There are a number of other issues experienced by the vehicle controller caused by unknown driving conditions such as: icy or wet roads, variable inclines, unpredictable driver commands (the demanded acceleration rate or operating speed is not known in advance) and large mass changes, especially for industrial vehicles.

### 1.4.2 Cost Consideration

As this research work is intended for a commercial product, the processor used for the implementation of the control only does fixed point maths and has limited processing power, especially as a great deal of the processing resources is used providing other functionality such as motor control.

The processor used is a Texas Instruments TMS320F2811 32bit fixed point processor operating at $100MHz$. The current functionality includes but is not limited to: current control every $125\mu s$; motor control and hardware temperature estimation every $250\mu s$; speed control every $5ms$; battery and controller voltage, current and temperature protection every $5ms$; CAN bus - for other controllers and displays; vehicle control - torque and speed references to other motor controllers (if the vehicle master controller); logging odometer and other data and digital and analogue inputs and output - directions and throttle inputs, contactor/relay PWM, brake lights. A large amount of the research has involved optimising advanced non-linear estimation methods to implement them in a resource limited environment, whilst still giving the same results.

## 1.5 Contributions to Knowledge

The following list are believed to be areas of the work not published by other authors:

- Sensorless traction control and anti-lock regenerative braking scheme without wheel speed sensors and with vehicle mass changes. The control scheme detects when the tyres are going to lose traction with the road and reduces the torque being applied, no additional sensors are required and it is robust to mass changes.

- An open loop backlash scheme that recovers energy from the axle twist and uses it to reduce backlash traversal time. Ideally the gear backlash should be traversed as quickly as possible and by using the energy stored in the axle twist, the time to mesh the gears to the other side of the backlash can be reduced without a large collision.

- Vehicle state estimator model comparison 5 state, 6 state and 4 state. A number of different vehicle models are compared to determine the advantages of each.
- Correcting a vehicle speed estimator, switched steady state Kalman Filter (KF), with a mass estimator using Recursive Least Squares (RLS), when the vehicle is subject to significant mass changes. Vehicle speed estimators are very dependant on the correct mass being used in the model, but it is common for the vehicle mass to change by a significant amount which can reduce its accuracy.
- Different sample rates within a Kalman Filter estimator; the prediction and correction run at different rates. The Kalman Filter measured states are only obtained at a lower rate than what the estimator needs to operate at.
- Comparison of mass estimation techniques which obtain vehicle mass quickly. A number of methods (using RLS to correct KF, using a different vehicle model and using different KF noise matrices) are compared to determine the most effective way of dealing with vehicle mass changes.
- Speed PI control improvements - resetting speed demand. After traversing the backlash the speed PI loop demand is reset back to the actual vehicle speed to avoid a large overshoot in acceleration when normally the speed loop would try to catch up with the demand.
- Using Genetic Algorithms to tune drivetrain parameters by rapidly removing motor torque at multiple torque levels. Accurate parameters are required for simulation and the estimator. Theses are determined by removing the applied motor torque rapidly and measuring the response.
- Using Genetic Algorithms to tune Kalman Filter noise matrices to compromise between multiple vehicle driving conditions. A compromise is achieved when a number of different conditions are tested simultaneously in simulation to find the optimum gains (noise matrices $\mathbf{Q}$ and $\mathbf{R}$) for the Kalman Filter.
- Extracting noise from an estimated and measured signal to compare noise reduction of an estimator. The noise component is separated from the signal to determine its covariance as this is required for the Kalman Filter $\mathbf{R}$ matrix.

## 1.6 Thesis Overview

Chapter 2 looks in more detail at the issues encountered by the vehicle controller, first listed above in section 1.4.

Chapter 3 will review and compare published work in the field; to highlight the differences between previously published work and the results of this research.

Chapter 4 includes the derivation of the vehicle model used for simulation and to develop estimation and control schemes.

Chapter 5 contains the results of the tests for determining the parameters of the test vehicle and a simplification of the model for real time estimation and control.

Chapter 6 contains the theory and results for the estimation of vehicle speed, mass and gradients, along with other vehicle states.

Chapter 7 contains the theory and results for a control scheme to give improved vehicle response, preventing tyre slip, and an open loop backlash traversal scheme.

Chapter 8 summarises the results of the previous chapters and looks at any further work that could be done to improve on the work carried out.

The appendix (chapter A) contains the work that would spoil the flow of the text, as it is not required for reading the thesis.

# Chapter 2

# Vehicle Dynamics Issues

## 2.1 Introduction

As was first discussed in the introduction sections 1.4 and 1.4.1, there are a number of mechanical characteristics of vehicles that lead to an undesirable driving response. This section will give a brief overview of how the present control schemes work along with some measured vehicle traces from the test vehicle highlighting the issues encountered.

## 2.2 Present Control Schemes

The present control scheme for speed control mode is shown in Fig. 2.1 and torque mode with speed limit in Fig. 2.2, both these diagrams are simplified. The only speed feedback is from the motor encoder, which in the next section of this chapter is shown to not be equal to the speed of the vehicle. There is a desire to not fit any additional sensors to the vehicle itself, such as wheel speed sensors. In some cases there is great difficulty in tuning the speed control loops, and they have to be significantly detuned to not react to the motor speed oscillations and drivetrain dynamics.

In speed control mode the accelerator pedal is used to give a speed demand level with different acceleration rates applied depending upon the rates specified for each condition, such as the accelerator being released or brake pedal being pressed, as shown in Fig. 2.1. The PI speed loop using the motor speed feedback then calculates a torque demand level input to the motor control.

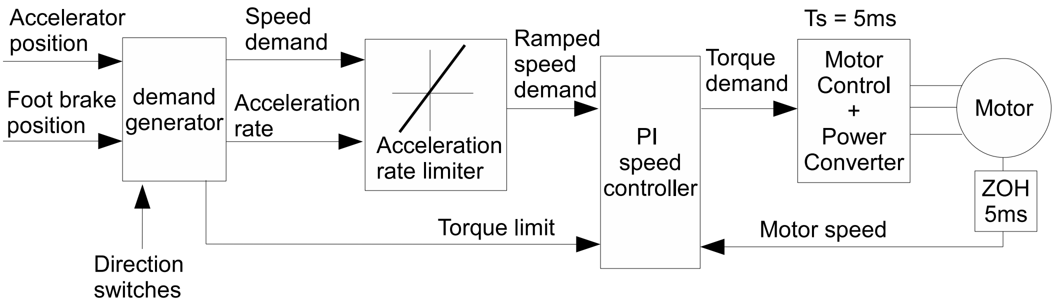

Figure 2.1: Typical vehicle speed control scheme

In torque control mode the pedal demand is used to give a motor torque level, scaled and rate limited depending upon the pedals being pressed and direction selected, so typically releasing the accelerator will cause torque opposing the motor speed to be demanded, which is then increased if the footbrake it pressed, as shown in Fig. 2.2. Speed control is still used in torque mode for limiting the speed; this speed limit can also be just ramped up to the maximum for the direction selected, or set to a level depending upon the position of the accelerator pedal, to give a controllable speed when descending gradients.

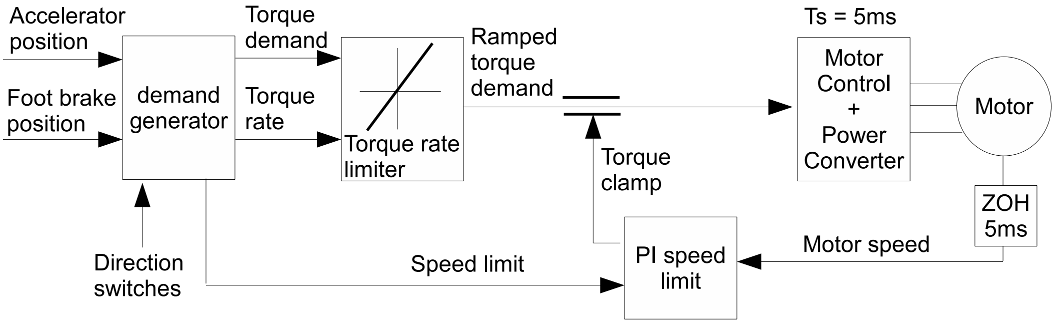

Figure 2.2: Typical vehicle torque control scheme with speed limit

## 2.3 Vehicle Dynamics Issues - Experimental Results

For the experimental results within this thesis a commercially available golf buggy is used which is fully described in section 5.1 and shown in Fig. 2.3. The vehicle is rear wheel drive with a single permanent magnet synchronous motor. All vehicle tests were done indoors in a $35m \times 5m$ driving area (shown in Fig. 2.4 and detailed in section 5.1.6) which limits the top speed and duration of tests, although the chosen vehicle has a limited top speed due to a high gear ratio.

Figure 2.3: Photo of the test vehicle

Figure 2.4: Photo of the driving area where the experimental work was carried out

An overview of the vehicle is shown in Fig. 2.5. Mechanical connections between blocks are indicated with solid lines and the dotted line indicate electrical connections. There is no torque measurement on the vehicle, but the motor has been characterised with a calibrated torque transducer on a dynamometer (section 5.1.3) to confirm that the actual motor torque is equal to the torque demand, therefore torque demand is used for the motor torque for all experimental results. The motor speed is measured from the encoder on the motor used for vector control (section 5.1.3). The vehicle has wheel speed sensors fitted to two wheels (section 5.1.4). In the following trace wheel speed comes from the rear right wheel and vehicle speed from front left wheel (non-driven) as this should always have good traction due to having very little torque applied to it. In order to measure vehicle acceleration an accelerometer is connected to the motor controller processor (section 5.1.5).

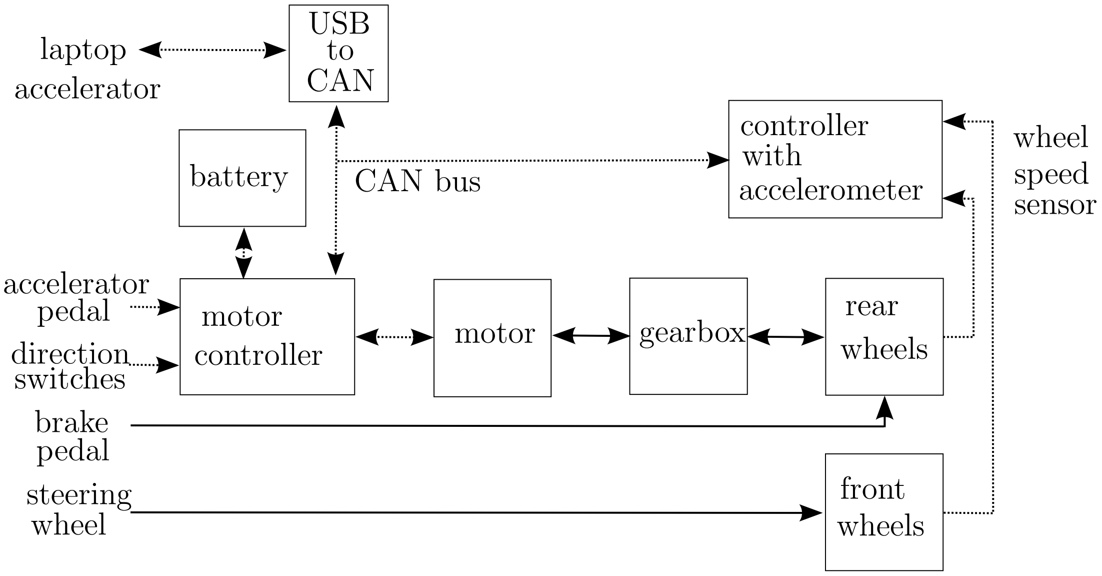

Figure 2.5: Block diagram of the test vehicle

### 2.3.1 Torque Reversals

The main issue with the vehicle dynamic response is the smoothness of the vehicle's response; oscillations in motor speed and therefore vehicle acceleration are often triggered during torque reversals where the backlash in the gearbox has to be traversed, as shown in Fig. 2.6 and 2.7. In these traces the oscillations continue for a significant and noticeable amount of time leading to non smooth vehicle deceleration, with a large jerk upon entering braking. It is possible to reduce the oscillation's magnitude and duration by reducing the slew rate on the torque demand, both removing the driving torque more slowly and applying the braking torque more slowly. But on many vehicles the rate of change has to be decreased to the point where the vehicle feels as though braking is not going to be applied as there is an extended delay before removing torque and entering braking, causing the driver to apply the mechanical brakes to act against the driving motor torque. The control should be improved so that the torque can be changed rapidly without any oscillations.

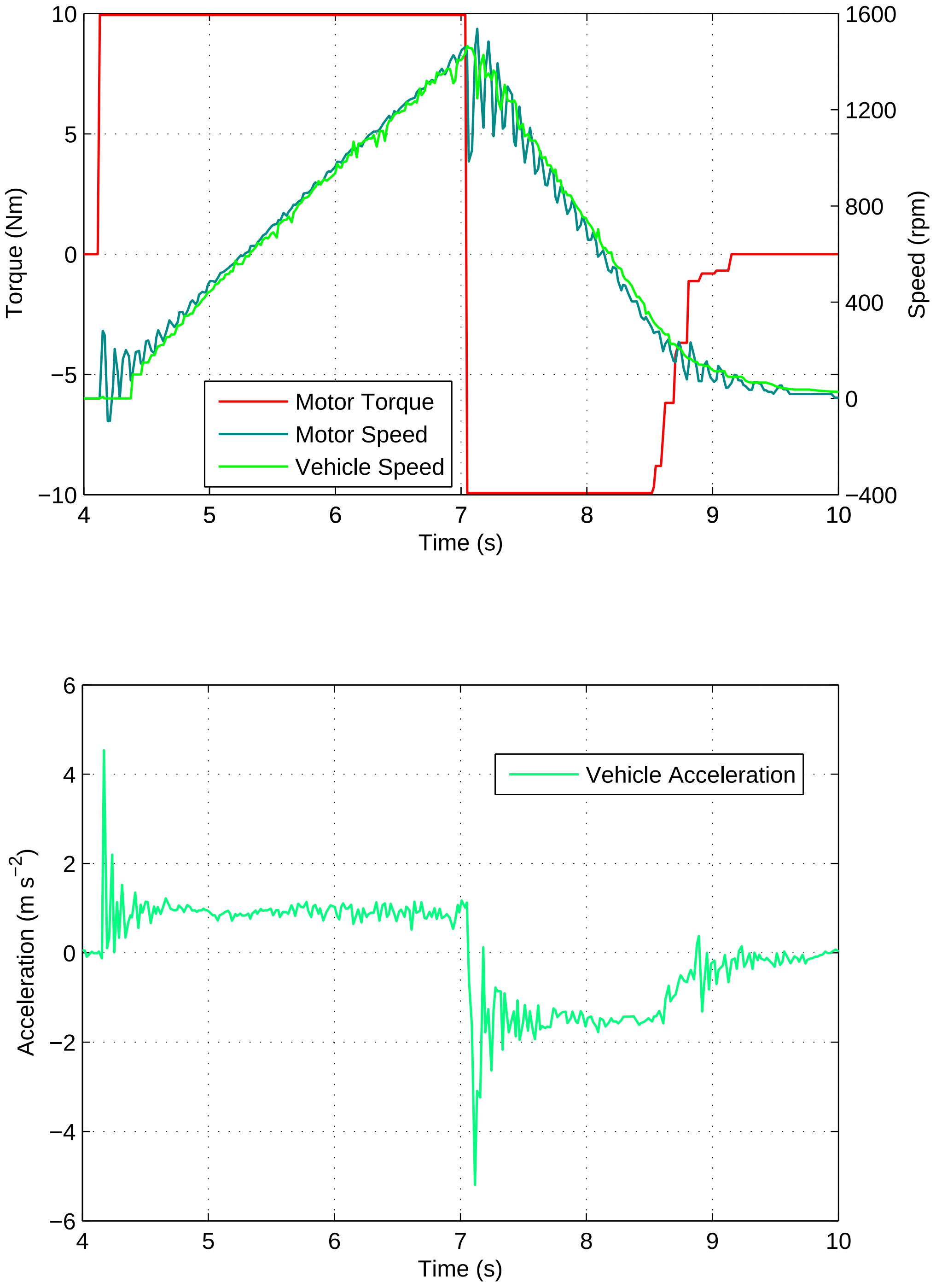

Figure 2.6: Experimental results of torque reversal - forward direction

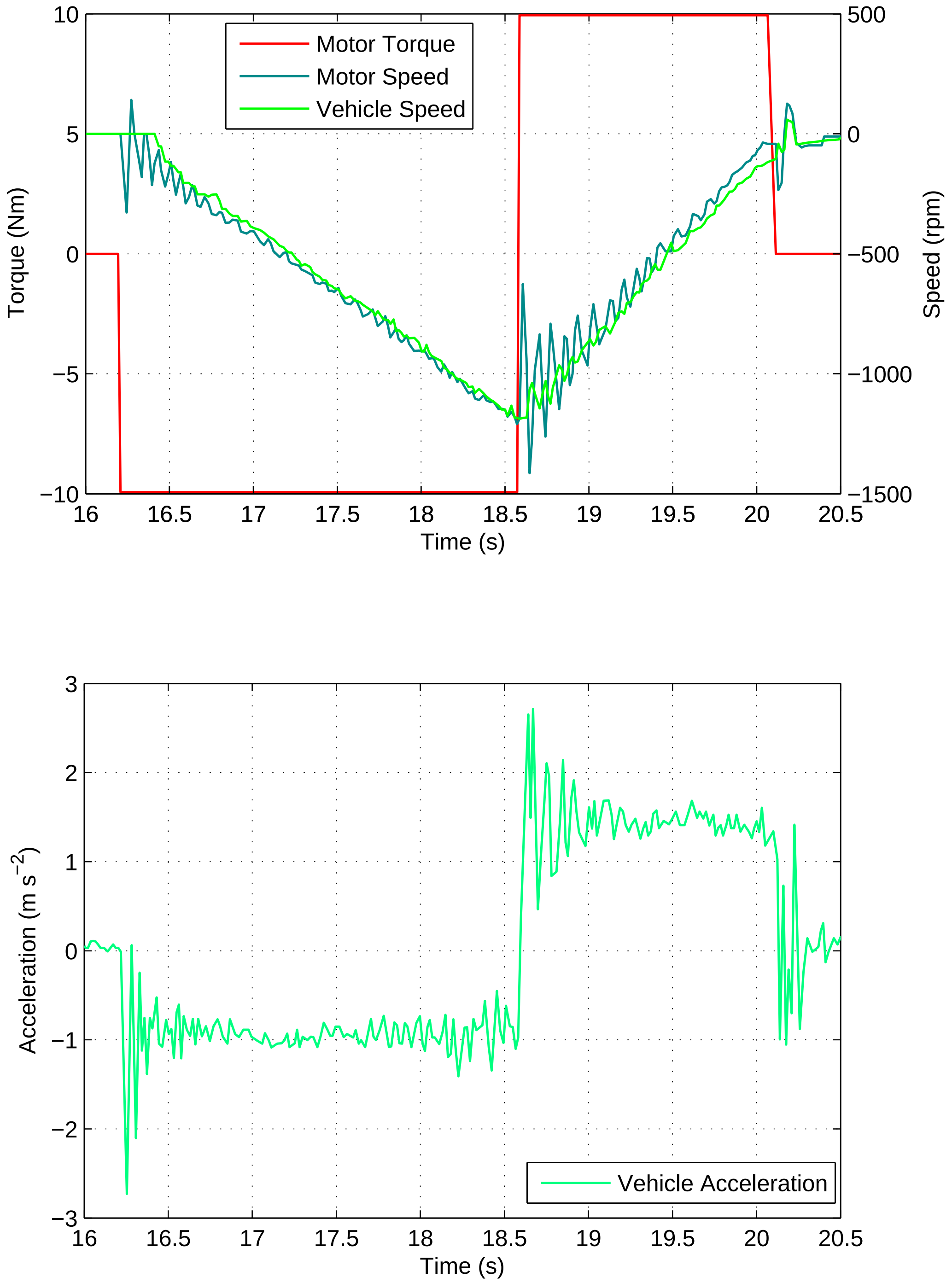

Figure 2.7: Experimental results of torque reversal - reverse direction

### 2.3.2 Loss of tyre traction

If large torques are applied, especially braking torques when driving forward as the vehicle is rear wheel drive, it is easy to lock the wheels, leading to loss of traction, shown in Fig. 2.8 where vehicle speed is measured from the front left wheel (the non driven wheel). In this trace the braking torque is increased greatly to emphasise the issue, but similar results would be seen with relatively low levels of braking torque and a reduced traction driving surface, such as ice. The braking torque causes the wheels to lock immediately, signalling to the controller that the vehicle has come to a stop and so the vehicle coasts to a stop for the remainder of the test. In the reverse direction, as the vehicle is rear wheel drive, (driving wheels now at front of vehicle), much higher braking torques are required to lock the wheels as shown in Fig. 2.9, highlighting that different levels of braking torque are available in different directions.

A common solution to this for on-road vehicles is to set the braking torque to very low values, such as 5% of the maximum torque, but this reduces the regenerative braking capability of the vehicle, leading to decreased energy recovery and therefore shorter vehicle driving range.

Losing traction is much more serious in speed control mode, as the controller then has the tendency to keep the wheels locked until the vehicle has come to a stop. This is because the demand has to track above the measured speed in case the actual speed is decelerating faster than the demand due to use of the mechanical foot brake or road gradient. If the measured speed drops to zero speed during a wheel lock event, the speed demand is also reduced to near zero.

The following figures, 2.8 to 2.11, vehicle speed is given from the front left wheel sensor (FL) and wheel speed is given from one of the driving wheels, the rear right wheel speed sensor (RR). The torque is seen to decrease in these traces at higher speeds due to field weakening.

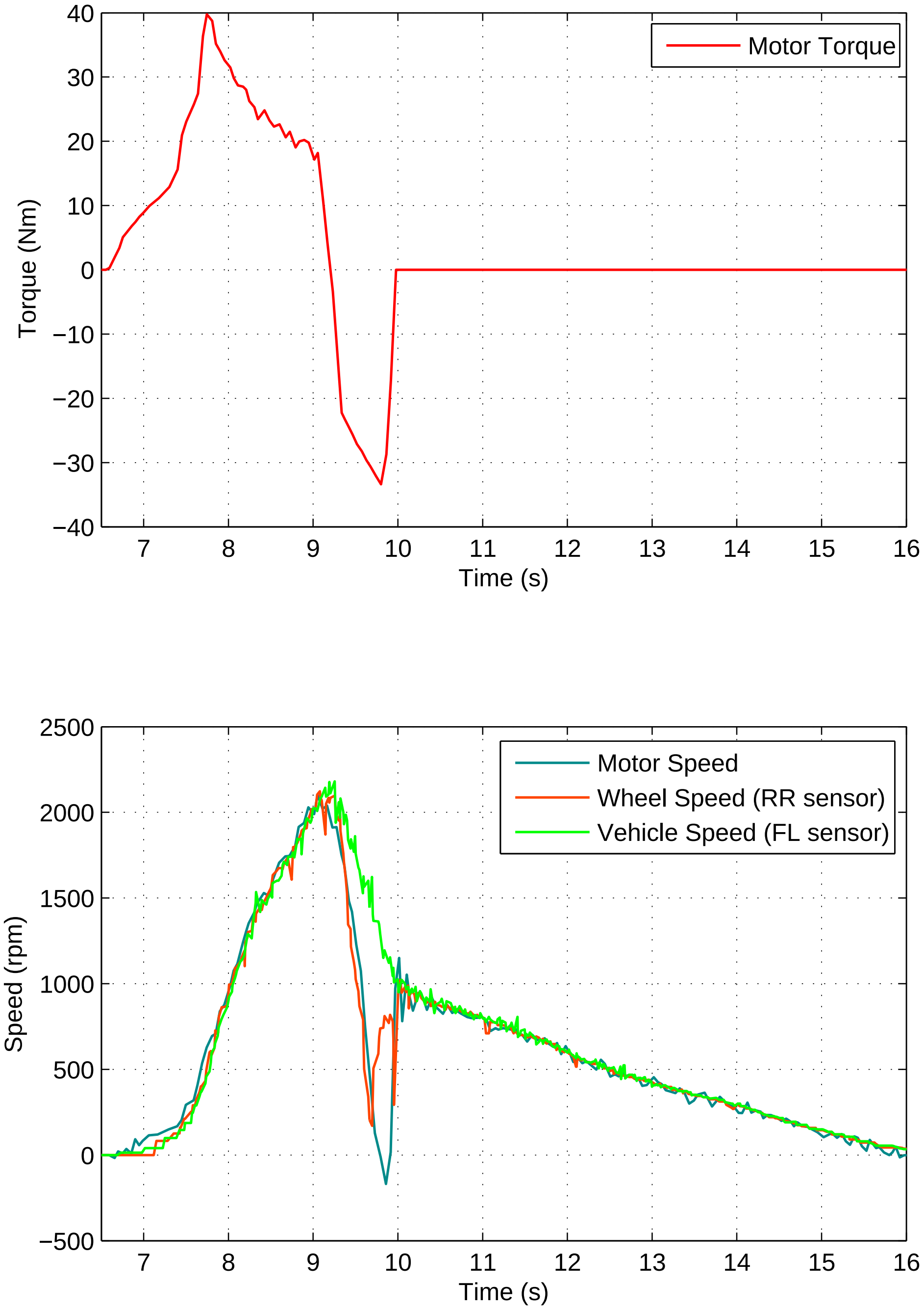

Figure 2.8: Experimental results showing traction loss whilst braking in torque mode - forward direction

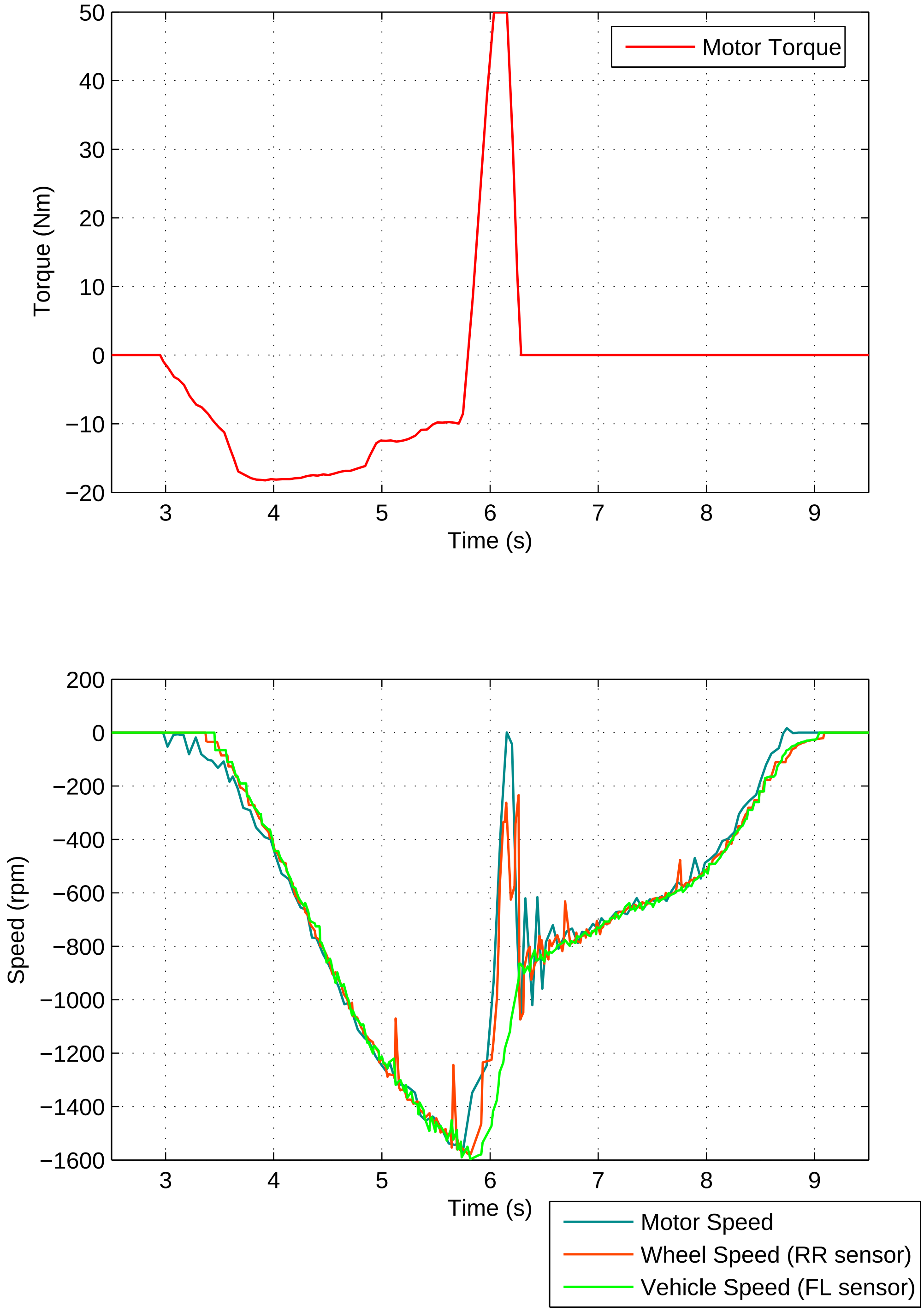

Figure 2.9: Experimental results showing traction loss whilst braking in torque mode - reverse direction

The same issue is true for acceleration (but in the opposite direction) and much higher torques can be used when accelerating forward shown in Fig. 2.10 than in reverse in Fig. 2.10.

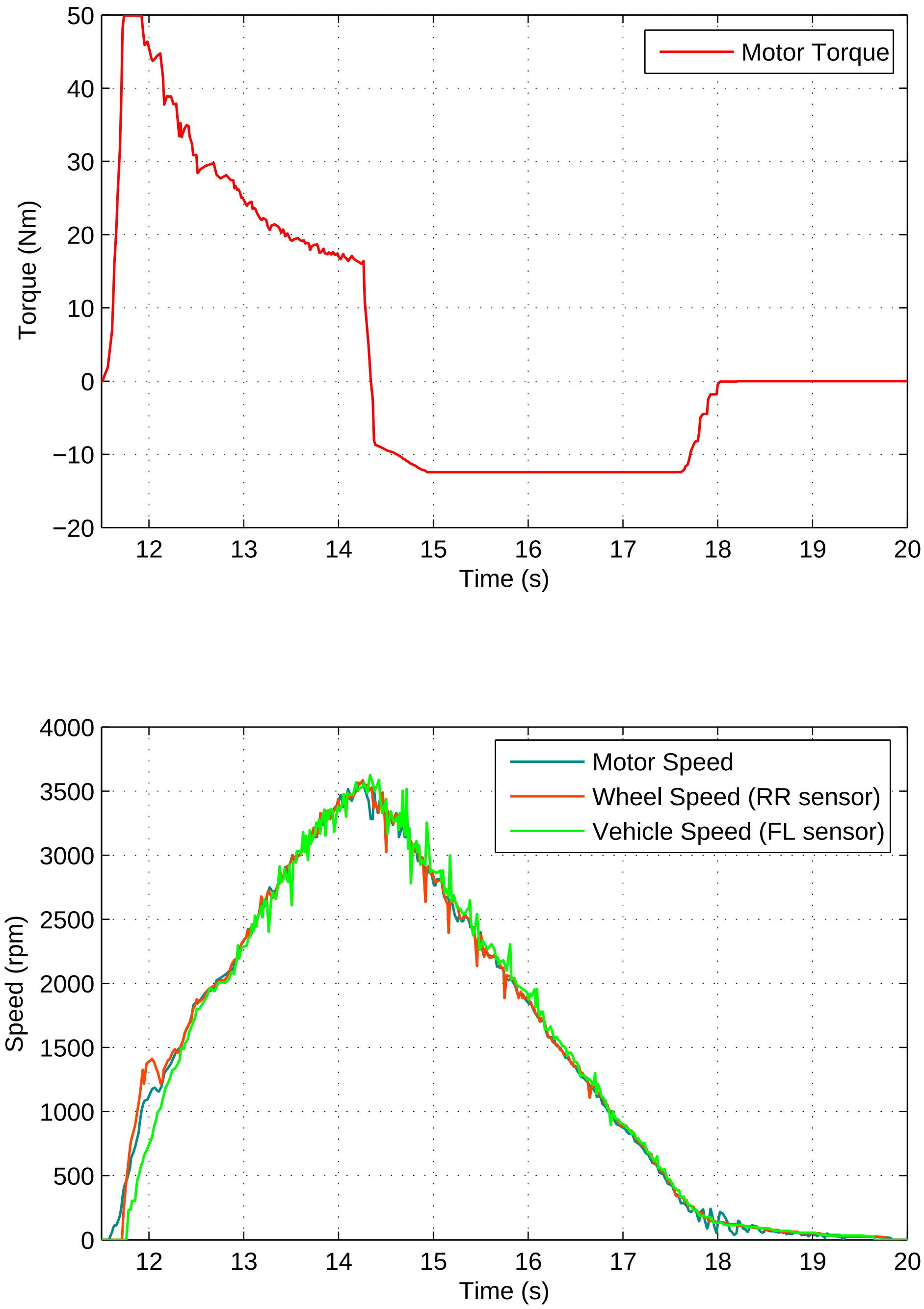

Figure 2.10: Experimental results showing traction loss whilst accelerating in torque mode - forward direction

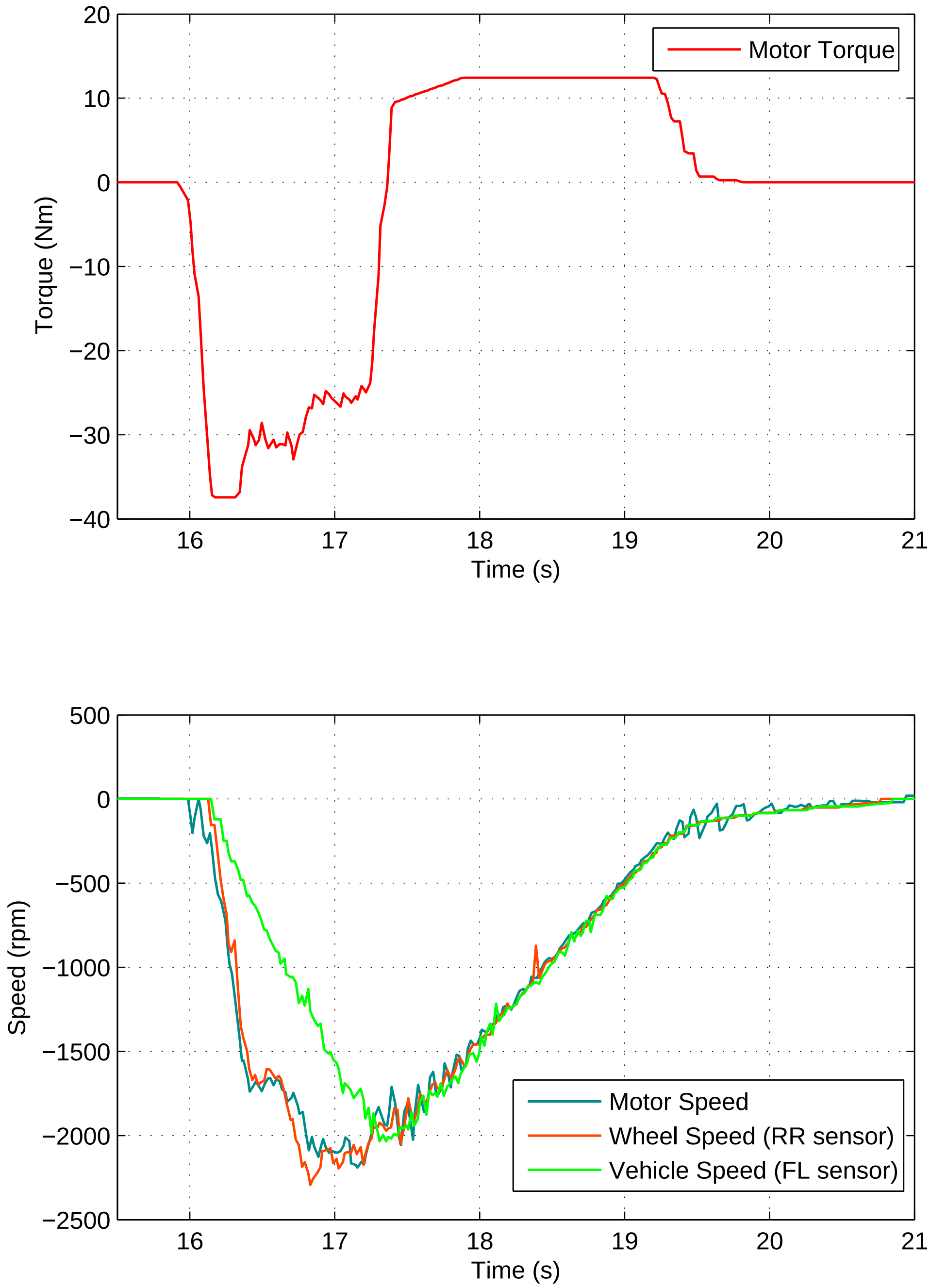

Figure 2.11: Experimental results showing traction loss whilst accelerating in torque mode - reverse direction

A difference can be seen in the above trace between the measured motor speed and measured rear right wheel sensor, this is because the both rear driven wheels will not necessarily apply equal torques, so it is possible for one wheel to lock but the other remain in traction.

### 2.3.3 Speed control

Closed loop speed control is required for some industrial applications and generally all torque control applications require some form of speed limit control. Tuning of PI gains can lead to a lengthy vehicle commissioning procedure for each controller application and finding suitable gains that cause zero overshoot or oscillations often lead to a very sluggish response, as shown in Fig. 2.12 with large speed errors during acceleration. Increasing the integral gains improve the response time, but always leads to overshoot as seen in Figs. 2.13, 2.14 and 2.15, which have increasing integral gains. Any increase in proportional gain leads to high frequency oscillation or chattering due to the drivetrain dynamics.

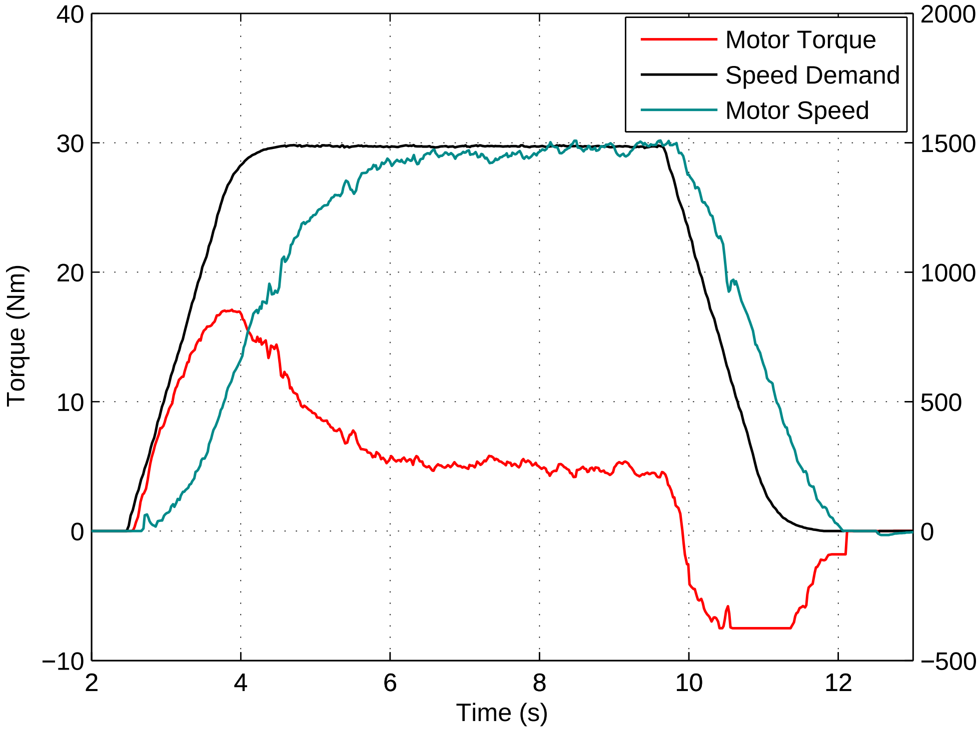

Figure 2.12: Experimental results of speed control tuning ($K_p = 80$, $K_i = 1$)

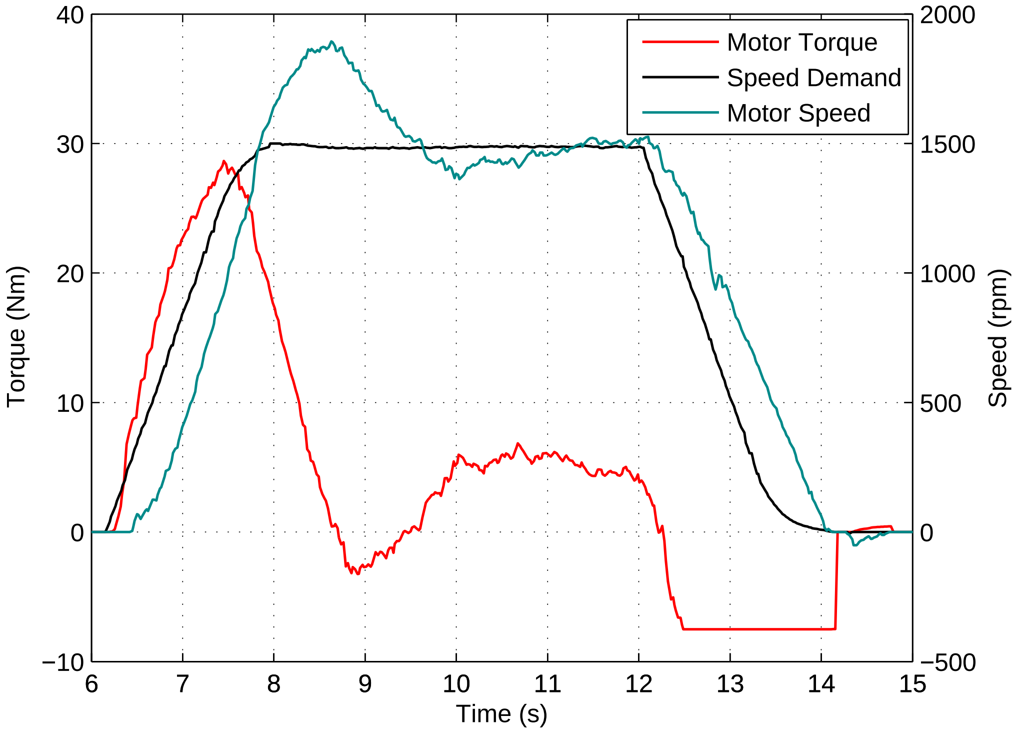

Figure 2.13: Experimental results of speed control tuning ($K_p = 117$, $K_i = 16$)

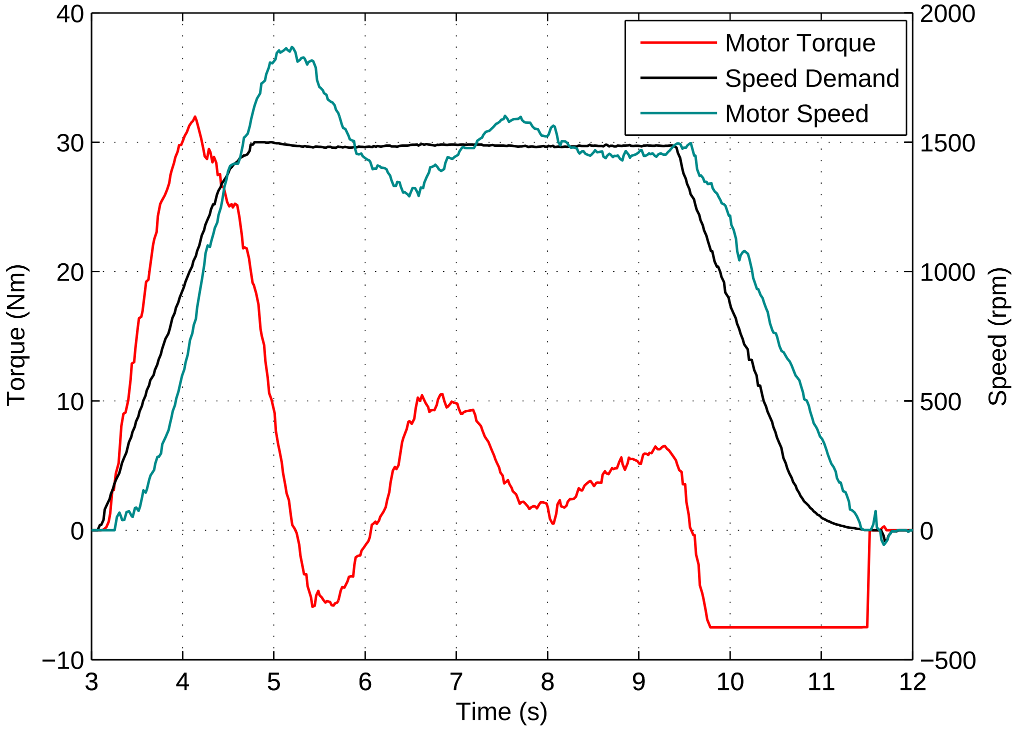

Figure 2.14: Experimental results of speed control tuning ($K_p = 117$, $K_i = 32$)

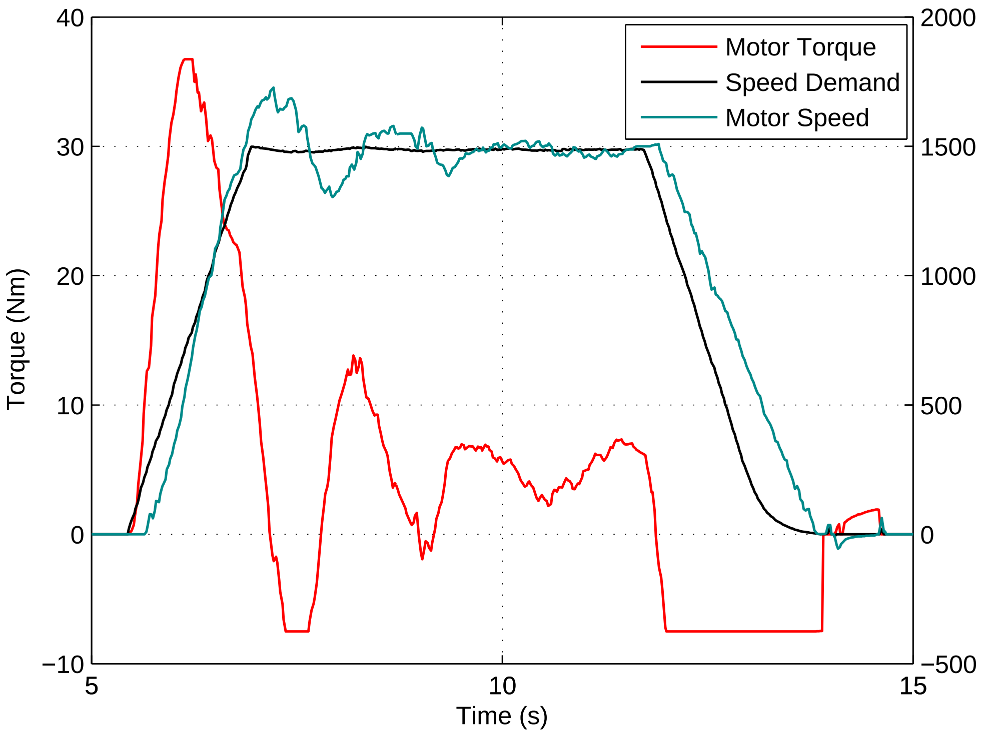

Figure 2.15: Experimental results of speed control tuning ($K_p = 117$, $K_i = 117$)

## 2.4 Speed Control Mode - Simulation Results

In order to simulate the issues described above a vehicle simulation model was developed which is described fully in chapter 4 and then the parameters for the test vehicle are determined in chapter 5. The following section uses this simulation model to compare the speed control performance using different feedbacks (motor or vehicle). A basic overview of the speed control simulation is shown in Fig. 2.16, where the switch is used to select which feedback is to be used.

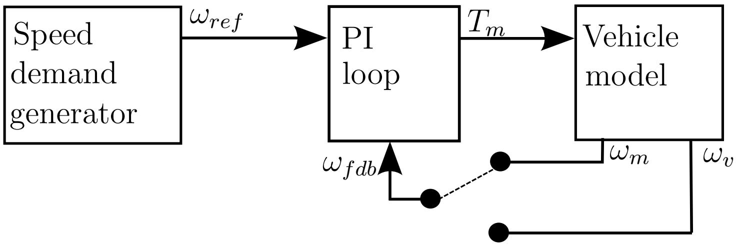

Figure 2.16: Speed control simulation diagram

The vehicle model block shown in Fig. 2.16 is expanded in Fig. 2.17 below. Where

$T_m$ is the motor torque demand, $\omega ref$ is the speed demand reference and $\omega fdb$ is the speed feedback which is either from the motor $\omega m$ or vehicle $\omega v$. In the simulation it is assumed that vehicle speed is measured. The torque is measured in $Nm$ and the speed is in $rpm$, the vehicle speed is converted to an equivalent motor speed taking into account the gear ratio and tyre radius, so that the controller gain is not affected and it allows for easier comparison.

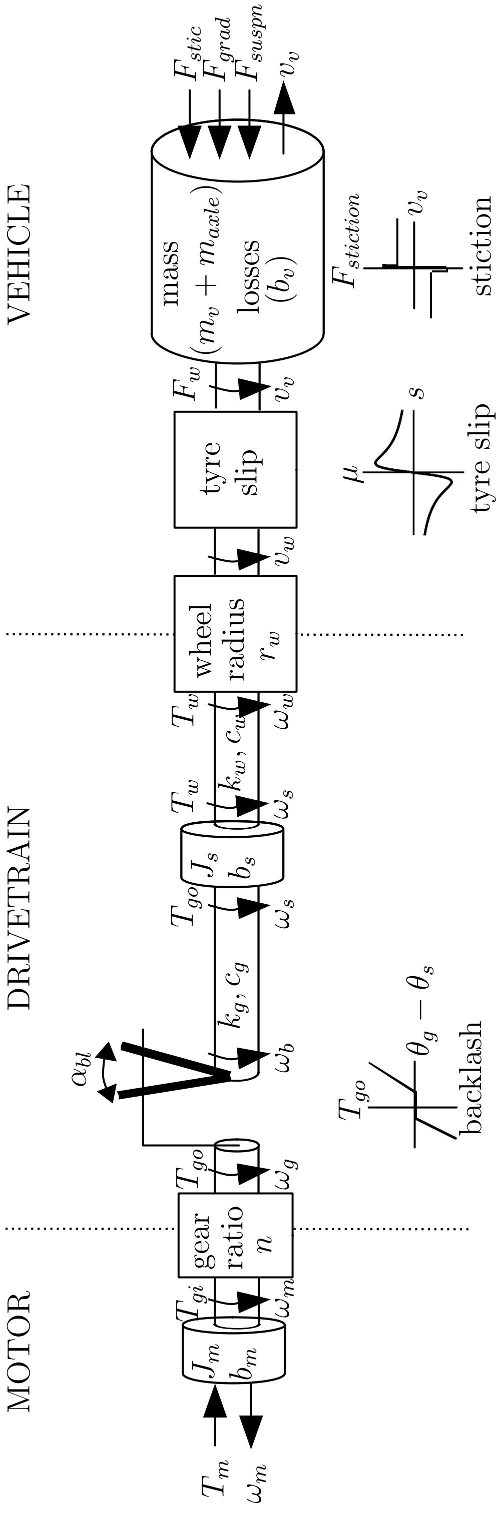

Figure 2.17: Diagram of vehicle model used in simulation

In Fig. 2.18 a simulation of the PI speed control is shown, using motor speed feedback as this is normally used. The control performance looks acceptable but the speed lags the demand and overshoots. In Fig. 2.19, where the vehicle speed is now used for the control feedback. It can be seen that the performance is the same as the previous result with the same gains.

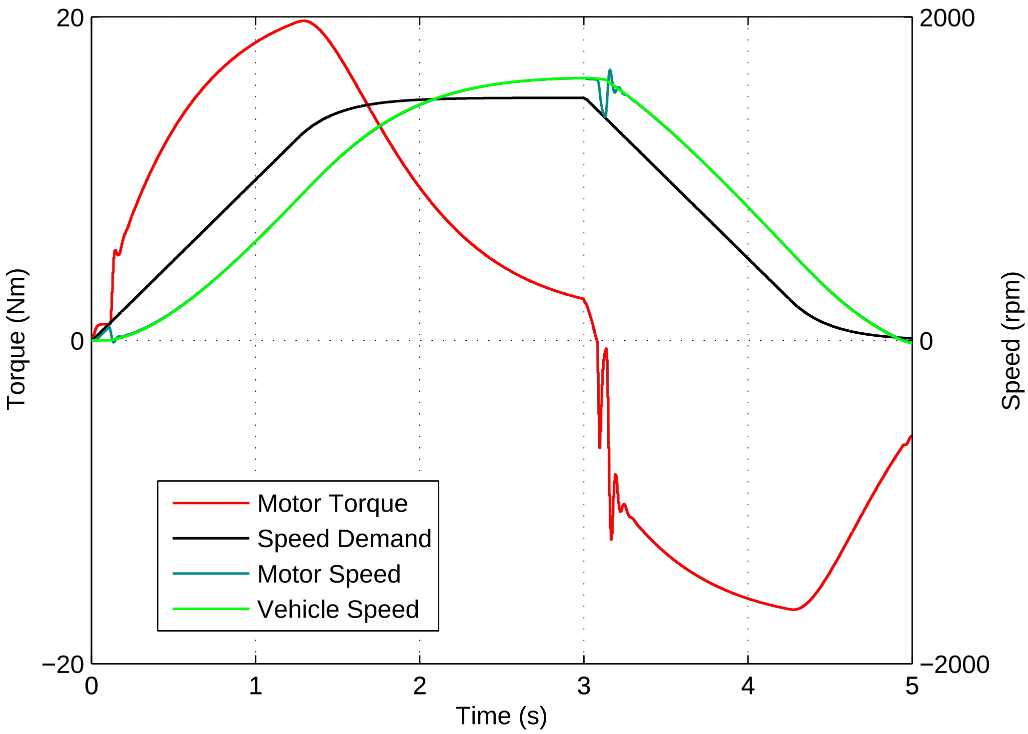

Figure 2.18: Simulation results using motor speed feedback with low gains

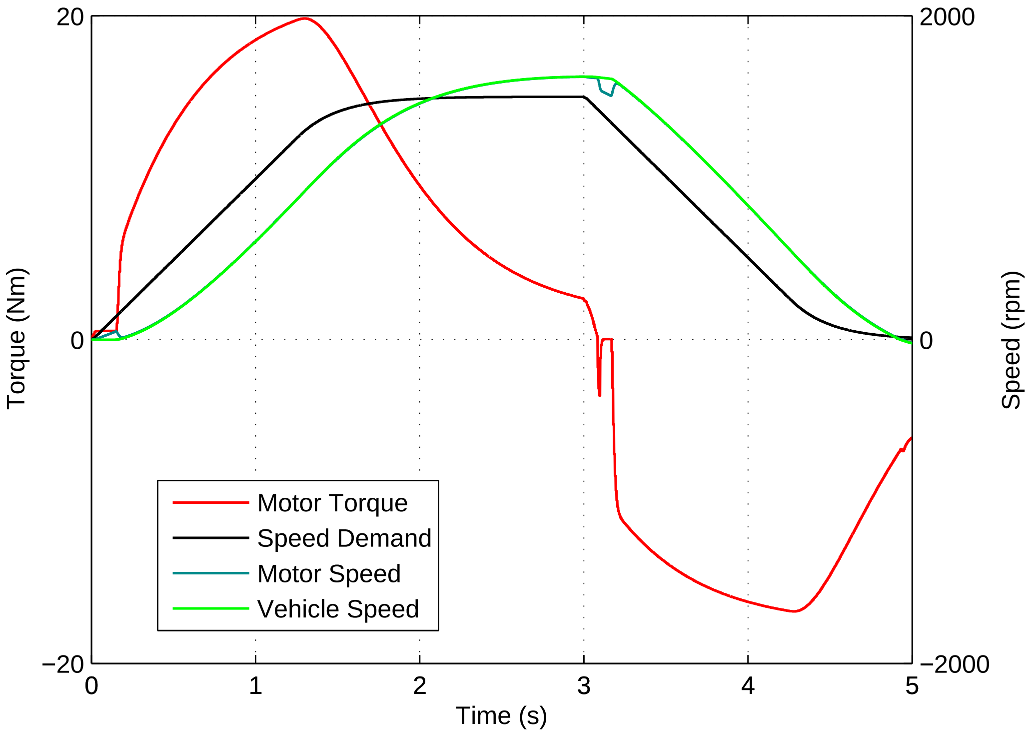

Figure 2.19: Simulation results using vehicle speed feedback with low gains

With vehicle speed feedback much higher gains (of the order of 10 times larger for $K_p$ and 100 times larger for $K_i$) can be used, as shown in Fig. 2.20. If the same gains are used with motor speed feedback the performance is unstable as shown in Fig. 2.21.

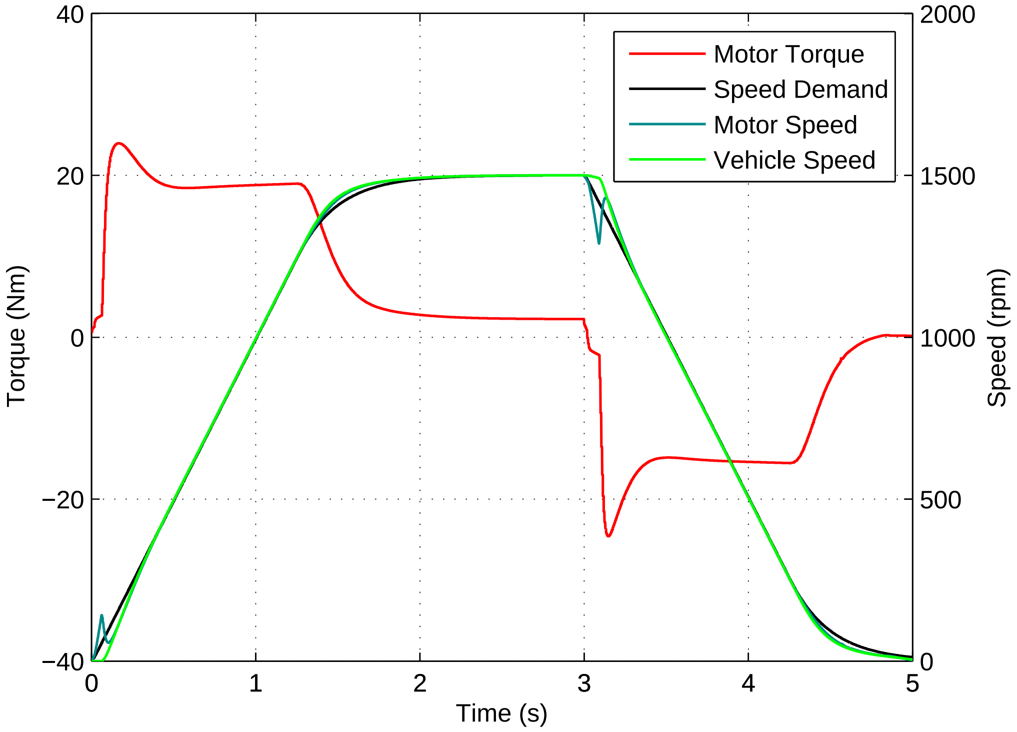

Figure 2.20: Simulation results using vehicle speed feedback with high gains

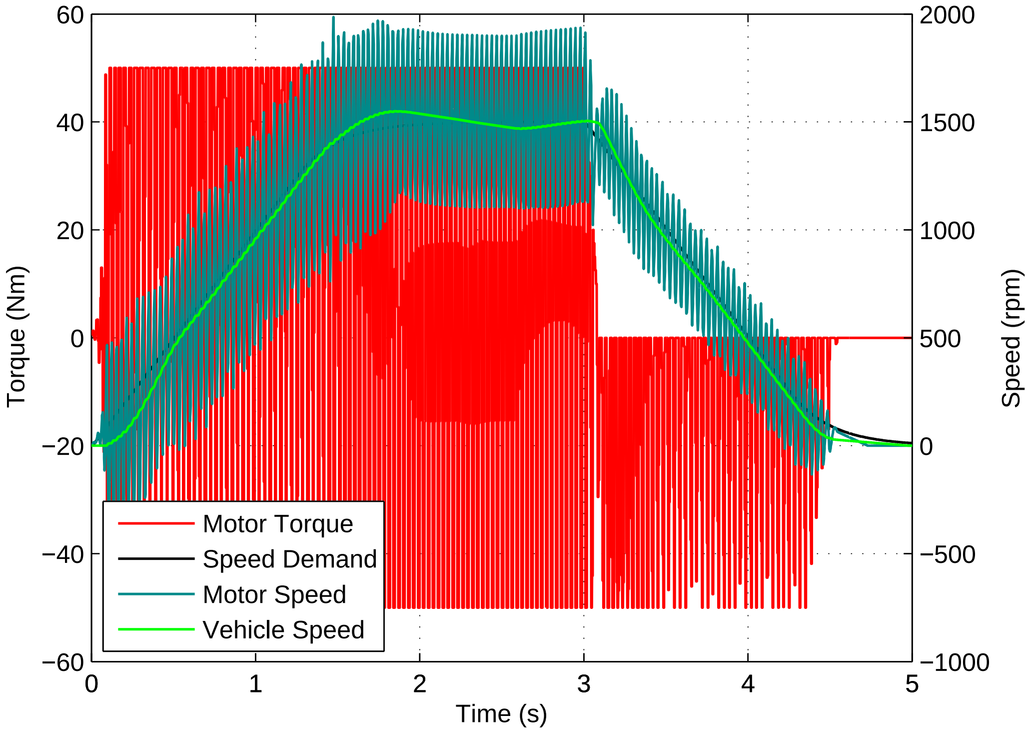

Figure 2.21: Simulation results using motor speed feedback with high gains

## 2.5 Summary

The issues first mentioned in the introduction were looked in more detail in this chapter. It was found that there are a number of undesirable characteristics that ideally need to be solved through improved estimation and control. The main undesirable characteristic is the smoothness of vehicle response as torque transitions from drive to braking can lead to osculations in motor speed which are then felt in the vehicle acceleration. Other issues shown above in the previous sections are loss of traction due to tyre slip and poor speed control response.

# Chapter 3

# Vehicle Drivability - A Literature Review

## 3.1 Introduction

This chapter will look at other published work in the same field to the research carried out for this thesis. The aim is to see what work has already been carried out, there advantages and disadvantages and what new work could be done to improve on it.

This chapter will initially look at the general area related to the project in section 3.2 with specific interest in drivability issues encountered and vehicle parameter estimation is briefly looked at in section 3.3. The next three sections are based on estimation: vehicle speed estimation is considered in section 3.4, although this can also be compared to a two-mass (inertia) industrial drive and a number of the papers discussed have this as the application, vehicle mass estimation in section 3.5 and tuning Kalman Filter noise matrices in section 3.6. The last three sections look at control: smooth vehicle speed (or load speed for two-mass industrial systems) response in section 3.7, backlash control in section 3.8 and finally preventing loss of traction in regenerative braking in section 3.9.

## 3.2 Project Overview - Improving Drivability

Vehicle drivability is a general term used to describe all aspects of how a vehicle drives and responds to driver commands. It is nearly always a requirement for a vehicle to have both a fast response time and a smooth acceleration response [2]. Although these two requirements work against each other and it is common to compromise between them [3].

Many issues related to vehicle response are caused by the drivetrain, which is also an issue on traditional internal combustion (IC) engine driven vehicles. Although electric motors bring advantages in terms of a faster response time compared to an IC engine, they have the disadvantage that the lower inertia motor can lead to a poorer response, as it is more prone to oscillation due to less damping from the inertia (no requirement for a flywheel) and the lower frictional losses.

The gearbox and specifically its backlash are the cause of most drivetrain issues. The disconnect between the input and output side of the gearbox allows the motor to speed up whilst in the backlash and the subsequent collision leads to undesirable audible noise and vehicle driving feel as the rotational kinetic energy in the motor is rapidly transferred into the rest of the drivetrain and the vehicle acceleration leading to a jerk or shunt [4, 5]. This is especially true when going from low torque to high torque demands “tip-in” or the reverse “tip-out”.

For equivalent two-mass models for industrial systems such as paper machines it is possible to use demand ramps to remove any steps in the reference [6] and something similar to this has been shown to be successful in [7] for a vehicle application, where the acceleration demand profile is used for smooth acceleration after gear changes. Generally any limit on the vehicle’s response time will lead to an undesirable delay to the driver, making the vehicle feel sluggish [3].

## 3.3 Estimating Vehicle Drivetrain Parameters

Knowledge of the vehicle parameters is important to correctly estimate unmeasured states and to make the simulation work accurately. For the work in this thesis and typically for two-mass models the states are the speed and position of the motor and also the speed and position of the load (vehicle speed and distance for vehicle applications). If the model includes gearbox dynamics, the backlash angle will also be included as a state in the model. In [8], the use of Genetic Algorithms (GA) has been investigated for estimating the parameters for various two-mass systems and has been found to be successful even with noisy measurement data. In [2] estimating the drivetrain parameters including backlash for an Internal Combustion Engine (ICE) driven vehicle is carried out using estimated engine torque and measured engine and wheel speeds. It uses a switched Kalman Filter (KF) for state estimation, with a separate open loop mode whilst in the backlash. An optimisation scheme is used to find the parameters such as the backlash, engine inertia, shaft stiffness and damping. The work is similar to [9], but only estimating the backlash size using the KF directly.

## 3.4 Estimating Vehicle Speed and Drivetrain Dynamics Modelling

In the previous chapter it was shown in simulation that improved speed control could be achieved if vehicle speed was measured. As fitting additional sensors is undesirable, using estimation would be a better solution. A model of the vehicle is required along with a suitable estimation scheme.

A detailed model of an IC engine driven vehicle drivetrain is given in [10], it highlights the advantage of "Power-Oriented Graphs" for modelling a drivetrain, where the power is maintained in each block of the model, in this case usually as torque and rotational speed. It also shows how a complicated model can be simplified to a lower order system and still be a valid representation.

In [5] and [9] a two-mass (inertia) model is used to represent a drivetrain with backlash as it was found to be sufficient as the main focus of the work was on backlash. It also showed how a switched Kalman Filter (KF) could be used for estimating states of a non-smooth non-linear system, in this work the backlash size was the focus of the estimation scheme using engine and wheel speed sensors as feedback. The KF used here switches between two different models depending upon if in the backlash or not.

The estimation of vehicle speed is considered in [11], along with cornering dynamics. Feedback comes from acceleration in two axes, yaw rate (rotation around a vertical axis), steer angle (yaw rate demand) and wheel speed sensors. This is used to estimate vehicle speed in longitudinal and latitudinal directions under the assumptions that the terrain is flat, the tyre-road friction model is known and fixed and the mass does not change. It was found that when relying on the acceleration data, the speed drifted away due to integrating any zero bias in the sensor; this was also found in [12]. Using an assumed fixed tyre-road model in [11] was found in the low friction cases such as ice to give poorer estimation performance.

In [13] a number of different algorithms are considered for estimating vehicle speed. They all calculate the speed directly from the four wheel speed sensors and an accelerometer measuring longitudinal acceleration, without using an estimator/observer. It uses a vehicle status indication to switch between the sensors and found that if the accelerometer was calibrated and compensated for changes in vehicle pitch when accelerating or braking it could be successfully used to integrate to give vehicle speed.

Using more than one feedback gives the option of using the different feedback sources under different conditions, [14]. In this paper a Fuzzy Logic (FL) supervisor was used to tune the KF noise matrices of the estimator to favour different sensors, for instance at low accelerations the wheel speed sensor is used, to avoid the zero offset of the sensor (accelerometer), and at wheel lock conditions using the accelerometer over the wheel sensors, as it was estimating longitudinal vehicle speed. This does require the full Kalman Filter (KF) algorithm to run on line to recalculate the Kalman gain with changes in noise matrices, **Q** and **R** at each iteration.

An alternative to the Extended Kalman Filter (EKF) for non-linear estimation is the Unscented Kalman Filter (UKF) [15]. It was used to estimate wheel speed from noisy low resolution wheel speed sensors. It was found that the UKF gave less noise and more accurate results that an equivalent EKF, it also avoided some stability issues. Although both of these methods required more calculations than a standard KF.

To convert a continuous system model to a discrete system the Taylor expansion is often applied. In [16] it is considered that if small sample times are required, only the first approximation needs to be used as the accuracy improves with decreasing sample times (faster sample rate). If the model is to be updated on line such as in this example then the conversion time is important, but if the model is fixed then the highest order possible can be used.

The issues when using a fixed point processor are discussed in [17] and [18]. The first paper shows how it is important to include the calculation error from using fixed point as part of the process noise **Q**, so that the Kalman Filter (KF) estimator can take into account the inaccuracies. The second paper states that it is important to convert the model to a per-unit (PU) system, to avoid saturation and rounding errors in the estimator.

There are a number of different ways of estimating vehicle speed discussed above. Some required a lot of calculations such as EKF/UKF and some used additional sensors to improve the estimate. As for this work the vehicle doesn't permanently have any additional sensors and has limited processing resources in the controller, a switched KF using a fixed Kalman gain is chosen.

## 3.5 Estimating Vehicle Mass and Gradients

Changes in the vehicle mass can significantly change the dynamics of the vehicle and therefore the operation of the estimator and controller. The use of a Kalman Filter (KF) has been considered in [19] for estimating mass and gradients of large diesel trucks. It uses estimated engine torque and measured wheel speed and acceleration (from an accelerometer); known losses such as aerodynamic drag are also considered. It compares its results to other work, where the mass is estimated by measuring the acceleration difference between when in neutral during a gear change and then immediately after, to give a large difference in acceleration. Oscillations, which are common during a gear change can lead to poor results with this method and it does not work with automatic vehicles that do not disconnect the drivetrain during a gear change or electric vehicles with no gearbox. It is also important to avoid estimation when the friction brakes are used, as is difficult to model the losses involved here. Four different estimation methods are investigated; all use the estimated engine torque output. The methods are: estimating the slope from the measured speed only, it considers the frequency of gradient changes on a typical road and includes unmeasured force disturbances as a state; estimating slope and mass from measured speed only, the Extended Kalman Filter is now required due to the model being non-linear; slope estimation using measured acceleration only; and mass and slope estimation with measured acceleration, it allows estimating the gradient directly without any knowledge of mass. The results show that if the vehicle losses cannot be measured accurately, an accelerometer is required for estimating both slope and mass.

Some work considers calculating a gradient from accelerometers directly without using an estimator. In [20] an accelerometer is used with measured wheel speed, which is then differentiated and filtered to give wheel acceleration. The difference between the two gives the acceleration due to gravity, as the accelerometer will be also measuring gravitational acceleration. Acceleration due to gravity can then easily be used to calculate road gradient angle.

An alternative feedback source for mass and gradient estimation is the Global Positioning System (GPS). In [21] GPS is used for estimating mass and road gradient of large vehicles along with other vehicle losses such as rolling resistance and aerodynamic drag. The road gradient is initially found using two GPS sensors at each end of the vehicle, the difference in altitude of both sensors will give the road gradient, if filtered to remove the higher frequency vehicle pitch changes due to suspension, although this requires the vehicle to be fitted with two GPS antennas. A better scheme involves just using the GPS velocity in both horizontal and vertical axes to give gradient. It was found that the other parameters to be estimated required variations in vehicle speed to be able to identify the different losses.

The use of Recursive Least Squares (RLS) has previously been shown to be successful for estimating the mass and road gradients of heavy duty diesel trucks [22]. It was found that both states (vehicle mass and road gradient) could be estimated using RLS with multiple forgetting to give different estimation response to both mass and gradients, using only measured speed and estimated engine torque. The results show that the mass can be accurately estimated but the gradient can only be estimated if the estimator inputs are changing. The estimator had to be disabled around gear shifts as the oscillations lead to poor estimation.

Recursive Least Squares (RLS) has been shown suitable for correcting a Kalman Filter (KF) for low inertia or low resolution encoder [23, 24] applications. Although in these examples only a single inertia load is used, with no gearbox and shaft dynamics, making the inertia identification simpler. The RLS estimated inertia is then used to compensate the KF speed estimator.

Including the load inertia as a state and using the Extended Kalman Filter (EKF), with only motor side feedback, has been shown to successfully estimate the required states in [25] for industrial drives. But the EKF has to recalculate its matrices at every step, including the Kalman gain matrix, adding a large amount of additional processing overhead.

An alternative to the Extended Kalman Filter (EKF) for both parameter and state estimation, other than using a separate Recursive Least Squares (RLS) for parameter estimation, is to use more than one Kalman Filter (KF) [26, 27]. In [27] a dual KF is used, one for vehicle parameter estimates such as mass and the other for vehicle states such as speed. It shows the advantage that the parameter estimator can be disabled once suitable values have been found, reducing the calculations required during normal operation.

In [28] EKF is used for estimating vehicle mass and gradients, but in order to overcome the issue of the conditions not always being ideal for parameter estimation as mentioned in some of the earlier papers considered it uses a Model Predictive Control (MPC) to create suitable modified acceleration profiles for the estimation to be more successful. The results show good estimation but are only done in simulation and do not consider drivetrain dynamics.

The electric vehicle used in this thesis has a fixed ratio gearbox, so detecting acceleration differences during gear change is not possible. It is also not preferable to permanently add additional sensors such as GPS or accelerometers, so schemes that use these are avoided. Estimators for non-linear systems such as the Extended Kalman Filter require a large number of calculations, the processor used for this work is already performing a lot of functionality and therefore its usage is high. The use of two estimators, one for parameters and one for states was the best solution as it could be implemented with less resources.

## 3.6 Tuning the Kalman Filter Noise Matrices

The estimation accuracy of the Kalman Filter (KF) requires the correct process noise matrix $\mathbf{Q}$ and measurement noise matrix $\mathbf{R}$. In many cases it is common to use trial and error methods for tuning the matrices [29], although this can depend on the person doing the tuning and not give consistent results. In [30] a single factor was used to scale the **Q** matrix to tune the bias between using the measurements and the model for the estimated states. In [31] trial and error is used to find the optimal matrices for a particular system (motor), and then normalised to rescale for other motors, so that the trial and error procedure does not need repeating.

For calculating **Q** a recursive method was first proposed over 40 years ago, [32] it is commonly referred as innovation-based adaptive estimation, where the innovation sequence is assumed to be only white noise if correct **Q** and **R** matrices are used. Another common method is Multiple Model Adaptive Estimation (MMAE) [33], where multiple models are used on-line and evaluated to find the most accurate solution. Both of these methods, however, have the disadvantage that the exact system dynamics are required and they do not always guarantee finding the best **Q** matrix [34]. The MMAE scheme in [33] tries to overcome this by having a bank of adaptive models rather than fixed. In [34] a new scheme is proposed called optimisation based adaptive estimation that does not require any prior knowledge of the system dynamics.

Genetic Algorithms (GA) have been shown to successfully tune the noise matrices for an Extended Kalman Filter (EKF) for a two mass system in [25, 35]. GA also has the advantage that it works well with non-linear stochastic systems, it can avoid local minima and there is no derivative required in the cost function (as this can be difficult to calculate), unlike other optimisation techniques [36]. GA can have the disadvantage that the optimisation process can be very time consuming and take many hours to find the solution, but it is used off-line in this case so this is not an issue.

It is quite common to only look at decreasing the estimation error when optimising a Kalman Filter (KF) [37] and [38], and not consider the noise in the signal (measured and estimated). If the noise is extracted from the signal and its variance measured, the noise level can be included in the cost function (of estimator performance) to be optimised. In [39] and [40], methods for obtaining noise from images and speech respectively are shown. In [40] the noise estimation is used on-line with a KF to improve the speech. The method looks for a number of peak frequencies in a given power spectrum, from the dominant energy band the noise variance can be calculated.

Least squares has been used to estimate the **Q** matrix in [41] and [42]. They work similar to earlier schemes such as innovation-based adaptive estimation that use the innovation to calculate the noise level.

As the matrices **Q** and **R** are only required off-line, GA is used in this work as it has advantage of avoiding local minima. However it is common to only look at decreasing the estimation error in the cost function, this work also looks at decreasing the noise of the estimated signals.

## 3.7 Improving Vehicle Response

In most vehicle cases it is not desirable to fit additional sensors, but in these examples [43, 44] load side speed or acceleration sensors have shown improvement for a two-mass system; with similar dynamics to a vehicle drivetrain. In these examples the applications are industrial systems, in one a paper cross cutter and the other a machine tool. In [45] the effects of load inertia are considered when tuning gains for PID and Linear Quadratic (LQ) controllers for a two-mass system, but in this case there is encoder feedback from the load side of the system, rather than estimating all the load side states.

The drivetrain oscillations can be limited by simply limiting the rate of change of torque, which has been shown to work successfully here [6] for an industrial paper machine system, but this gives a sluggish driving response [3] for a vehicle and will limit the dynamic performance of the speed loop, therefore active damping methods are required. A common solution to damping oscillations is to estimate the gearbox output torque and use this to compensate the torque demand, but some schemes require the wheel speed to be measured [46]. It is possible to use an estimator instead of sensors and it has been shown that improved speed control can be obtained by estimating the speed of the load and the shaft/gear torque (or wheel torque in a two-mass vehicle system) [47–49], but the effects of the load inertia changing significantly have not been considered, which is quite common for a vehicle application, especially industrial vehicles. In [50] the model is expanded to a three-mass model to account for gearbox inertia, it also shows how important parameters such as natural frequency and damping factor can be calculated from the inertias, shaft stiffness and damping.

Other examples of using an estimator with a two-mass system to provide multiple feedbacks such as load torque, shaft torque and load speed are found in [51] and [52]. In [51] the estimator was found to be robust to reasonable load inertia changes and in [52] LQ control was used for control. In both cases the performance was found to be superior to standard PI speed control whilst still only using motor speed feedback; the load torque estimation gave improved disturbance rejection. Another example of using a Linear Quadratic Regulator (LQR) is [53, 54], which changes the cost function and therefore controller during backlash stages to reduce impact speed and oscillations.

Other solutions for improving speed control for a dual inertia model system include Model Predictive Control (MPC), which has been compared to standard Proportional Integral (PI) speed control and PI control using addition estimated feedbacks [48]. The predictive scheme shows dynamic improvement over the standard

PI control, and a lower drive shaft stress compared to the PI with multiple feedbacks (PI based state space control), but with a much higher calculation time.

An optimisation to MPC was tested in [55] where the explicit form of MPC is used for control of a two-mass elastic drive system. The algorithm is solved off-line and stored in a look up table, to greatly reduce the on line processing requirement. Similar to other methods the load side unmeasured states are estimated using a KF.

An example considering load inertia changes (of a two-mass industrial system) is [25]. EKF is now used unlike previous papers, with the reciprocal of the load inertia as a state. Trying to estimate both load torque and load inertia though cannot be done simultaneously. The speed control error is used to determine which state to estimate, with large errors requiring the inertia to be estimated.

It is possible to estimate gear torque using a deterministic observer using only motor acceleration and the known motor inertia (subtracting the multiple of these two from the motor torque demand). This technique has been used in these papers for improving speed control [56, 57] and torque control in [58] which is expanded to a three-mass system. In the speed control example motor speed is used for speed feedback, rather than trying to estimate or measure load side speed.

For engine driven vehicles, the engine's slower torque response can lead to difficulties in damping oscillations. The Smith predictor is ideal as it accounts for time delay when estimating states [3]. In [59] it is used for a diesel vehicle with a controller designed through pole placement, although similar to a Proportional Derivative (PD) controller to reduce jerks when requesting large steps in torque demand. In [60] a pole placement torque controller for an engine driven vehicle was also considered, although in this case Genetic Algorithm (GA) was used to enhance the pole placement.

In order to overcome instabilities around the backlash, previous papers considered have mostly all estimated unmeasured load states. In [61] an alternative is used that switches between two linear controllers. The high gain controller that give the best response during normal operation but is unstable around the backlash is used normally, but a lower gain detuned controller is used around the backlash.

## 3.8 Gearbox Backlash Traversal Schemes

The gear backlash can cause an undesirable vehicle feel and drivetrain oscillations which can then be compensated using methods described in section 3.7. But it is possible to reduce or cancel out the effect of the backlash, so that the collision between the cogs is reduced. In [62] a detailed comparison of backlash control schemes is carried out; it groups backlash control into two sections: strong action in the backlash gap and weak action in the backlash. The strong action refers to control that tries to traverse the backlash quickly such as control based on the inverse [63], the weak action refers to control that detunes whilst traversing the backlash. Another comparison is carried out in [64], but this time grouping the control methods to linear and non-linear. The best performing controller was found to be an active switching controller; where a different scheme is used to get across the backlash than what is used when the cogs are in contact. During contact mode a PID controller that is closed around vehicle acceleration is used along with a torque compensator, where gearbox output torque is estimated using a linear observer. In the backlash mode it controls the position of the motor gearbox cog to close the backlash gap, although by using feedback from the load (wheel) side.

A position control scheme in [65] is shown to give improvement as they have used two position control loops, one for the motor side of the backlash and an outer control loop for the load. The outer loop ignores the backlash and a feedforward position offset is given to the inner loop to cancel it out. Although the work was only investigated in simulation and it requires load position feedback as well as motor.

There are a few open loop backlash schemes in the literature, specifically in [66] and [67]. Both schemes look at traversing the backlash by accelerating and decelerating the gearbox input, to traverse the backlash quickly but minimise the impact speed when contact is regained. Although in [66] the experimental setup has hardly any inertia on the load side of the gearbox and in [67] the work is carried out in simulation only and the optimisation problem requires feedback.

In [68] a Model Predicative Controller (MPC) is applied to the problem of traversing a backlash. The MPC is solved off-line to allow it to be implemented in real time on a microprocessor. The application is for an engine driven vehicle so the control response time is much slower than the previous two papers and also it is not possible to generate braking torques apart from frictional losses.

There are a number of different gearbox backlash schemes but the majority are closed loop and so require accurate feedback. There is not much published material on open loop backlash schemes, so that will be investigated in this thesis.

## 3.9 Preventing Wheel Lock Caused by Regenerative Braking

Loss of traction is always a dangerous situation for any vehicle as it can greatly increase the distance required to stop, or lead to loss of control whilst cornering. Nearly all road cars are now fitted with anti-lock brakes which prevent the locking and loss of traction due to the hydraulic brakes. It operates by comparing the measured speed of all the drive wheels; if one is seen to decelerate more than the others the hydraulic pressure is released greatly reducing the braking force, this force is then reapplied when traction is regained effectively alternating between large braking forces and low braking forces [69].

An improvement on the standard anti-lock schemes involve controlling the brake pressure and therefore traction force to give an exact value of slip to optimise the braking force to the peak of the torque slip curve, rather than simply pulsating the brake force level, although this usually requires accurate knowledge of the vehicle speed. In this case tyre slip is the ratio between the wheel speed and the road speed. Nearly 25 years ago in [69], work was published for a scheme that regulated hydraulic brake levels to separately control front and rear brake forces, it did require knowledge of the vehicle speed which was assumed to be measured from a sensor. A better mechanism in [70] where Linear Quadratic Regulator (LQR) control for each wheel is used for friction brake control, it uses an Extended Kalman Filter (EKF) for estimating the vehicle speed and tyre slip ratio so that no additional sensors (apart from wheel speed sensors) are required. The experimental work is carried out on a standard on-road ICE vehicle.

Electric vehicles add the additional complication that regenerative braking is desired which can affect the brake balance especially when the motor drives the rear axle [71]. In [72] a new control method called the slip trial method that aims to combine the control of the anti-lock brakes and the regenerative braking using fuzzy logic, without knowledge or estimation of road condition or braking force. Although it is not clear that the vehicle speed feedback (measured or estimated) has been considered as the results were carried out in simulation.

Many on-road vehicles already have wheel speed sensors for anti lock braking functionality. In [73] they are used to for estimating vehicle speed and learning of road conditions, which is used during loss of traction to help track the peak traction coefficient through an open loop mechanism. It looks at combining the forces from hydraulic mechanical brakes and regenerative braking. A better scheme in [74] aims to completely remove the effect of regenerative braking on the rear wheels, so the vehicle drives like a non-electric vehicle whilst maximising the regenerated energy, using a PID controller on the wheel slip.

It has been shown the the tyre diameter can have a significant effect on tyre slip estimation [75], most other work assume it to be constant. In [75] the tyre radius changes are estimated slowly over time using GPS, which is then used to estimate the tyre slip assuming vehicle speed is equal to the speed of the non-driving wheels. This however adds the cost of a GPS sensor.

Similar to work above, in [76] it has been considered to control tyre slip directly using simple control methods such as PI controllers. In this case the vehicle speed is not measured and the tyre slip is estimated directly from an acceleration measurement from an accelerometer.

The use of more than one feedback sensor is considered in [14] and focusses on just estimating the vehicle speed, which can then be used for slip estimation. It uses a Kalman Filter (KF) to pre-filter measurements from an accelerometer and wheel speed sensors, using a third KF to estimate the vehicle speed. In order to compensate for situations such as loss of traction a fuzzy logic rule mechanism is used to tune the KF noise matrices **Q** and **R** to give emphasis on the most appropriate sensor in different conditions, such as locked wheels. It also compensates when the signals from each sensor have poor signal to noise ratios, such as when the acceleration or speeds are quite low.

In [77], no additional sensors are used for tyre slip estimation and control. A disturbance observer and parameter estimator are applied, using recursive least squares to estimate load changes. Although the experimental results were produced on an artificial laboratory set-up using a dynamometer and belts to give slip, rather than a tyre on a road or other surface, it also does not consider all of the unknowns encountered by vehicles.

Regenerative braking can have an effect on cornering dynamics as it can change the front rear brake balance from the ideal values normally controlled by the hydraulic brakes [71]; especially for rear wheel drive vehicles, as it is preferable to usually have most braking effort at the front wheels.

## 3.10 Summary

This chapter has considered a large range of published work related to vehicle drivability. In section 3.2 the definition of drivability was defined and it was noted that the gearbox and its backlash are the main cause of the issues and that significantly limiting the ramp rate of the motor torque output is undesirable.

The next section 3.3 briefly looks at estimating the drivetrain parameters. Most of the previous work looked at used Kalman Filtering for on-line estimation of parameters, but there was a paper that used Genetic Algorithms (GA) for a non-vehicle application. As most of the parameter estimation (apart from mass) only needs to be carried out off-line, GA was chosen to be applied to the work in this thesis.

Vehicle speed estimation is looked at in section 3.4. It has been shown that the steady state fixed Kalman gain filter can be applied to non-smooth systems with backlash. Other estimation schemes were looked at but used additional sensors or required a lot of computation.

Estimating vehicle mass is investigated in section 3.5. Some schemes involve fitting additional sensors or use algorithms that require a large amount of calculations such as EKF. Previous work has considered using two KF for estimating states and parameters separately for vehicle applications, or using RLS for inertia estimation with a KF speed estimator for non-vehicle applications - the work in this thesis will attempt to apply this to a non-linear vehicle application.

The best method for tuning the Kalman Filter noise matrices in this work is GA, see section 3.6. It allows for tuning under a number of different driving conditions simultaneously, so that the estimator is robust to mass changes. The time taken for the process to complete is not important and other iterative methods can converge to local minima.

In section 3.7 methods for improving the smoothness of the driving response whilst maintaining a fast response time are looked at. There is previous work for using estimated states to improve this for two-mass systems (many examples are not for vehicles) but not with large load inertia changes.

There are a number of gearbox backlash schemes looked at in section 3.8. The majority are closed loop and so require accurate feedback. There is not much published material on open loop backlash schemes, so this will be investigated in this thesis.

In section 3.9 preventing wheel lock is considered. It has been shown that if the vehicle speed and other unmeasured states are estimated, the loss of tyre traction can be avoided.

# Chapter 4

# Vehicle Dynamics Model

## 4.1 Vehicle Model Overview

This chapter aims to describe how all of the main vehicle characteristics have been modelled to create the entire vehicle model. The model includes the drivetrain, suspension, tyre slip and also losses acting upon the vehicle, such as rolling resistance, gradients and aerodynamic drag. Approximations have been used leaving the most significant features that affect the vehicle's performance, these will be discussed in each section. The content of this chapter has been used to create a vehicle simulation within MATLAB and Simulink in order to help with the testing and development of the work in the remainder of this thesis.

### 4.1.1 Force Direction Conventions

It is important to define the axes and force directions used throughout the rest of this thesis, they are all referenced to the perspective of the driver: the forward longitudinal direction is the x axis, the left/right direction is y axis and the downwards direction towards the road surface is the z axis. There are three rotations defined around these axes: roll, pitch and yaw, where there positive direction is defined in Fig. 4.1.

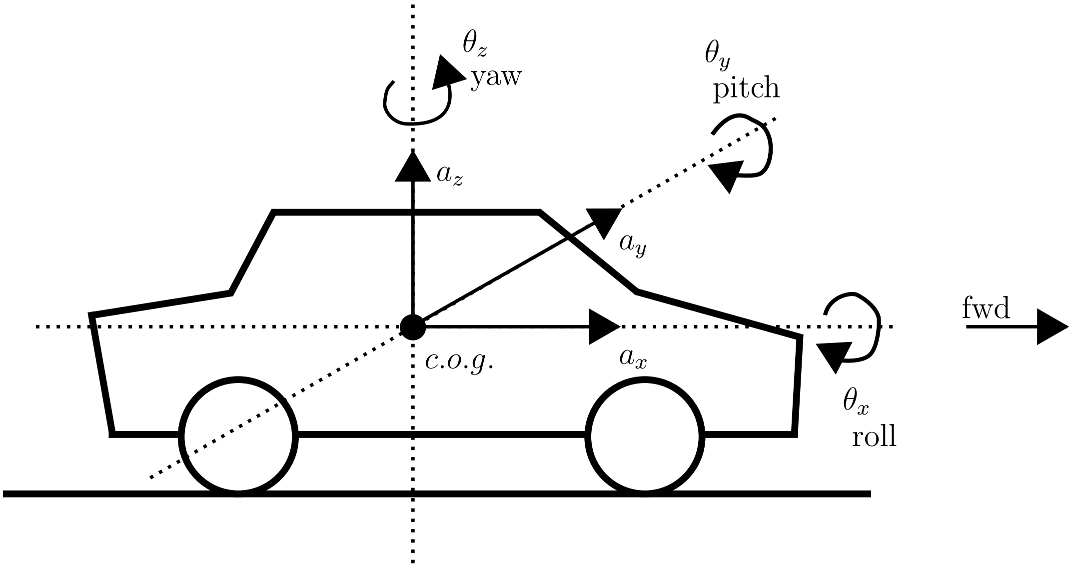

Figure 4.1: Force direction and rotational axes

The following subscripts are commonly used to represent the wheel being referred to: $_{fl}$ front left, $_{fr}$ front right, $_{rl}$ rear left, and $_{rr}$ rear right, when it is an entire axle $_f$ is used for front and $_r$ for rear.

### 4.1.2 Basic Model Fundamentals

The vehicle model is based upon the resultant torque ($T_w$) at the wheels, after the mechanical losses in the transmission system between the motor and wheels, causing a longitudinal force ($F_w$) depending upon the tyre radius ($r_w$) (4.1). This force is then reduced (or increased) depending upon the aerodynamic drag ($F_{drag}$), gradient force ($F_{grad}$), rolling resistance ($F_{roll}$) and longitudinal suspension force ($F_{suspn}$) acting upon the vehicle, giving the remaining force ($Fx_{accel}$) that is available for acceleration, (4.2) and (4.3). It can then be converted to an acceleration based upon the vehicle's mass ($m_{total}$), with the velocity and distance travelled calculated by integrating this acceleration (4.4). The force generated by the wheel is calculated simply based upon the torque applied and its radius. All forces $F$ are measured in $N$, all torques $T$ are measured in $Nm$, velocity $v$ in $ms^{-1}$, acceleration $a$ in $ms^{-2}$ and mass $m$ in $kg$. The standard power based diagrams are used with torque along the top and speed along the bottom, the multiple of both lines should almost always give power in watts [10].

$$Fx_{w-r} = \frac{T_{w-r}}{r_w} \tag{4.1}$$

$$Fx_{road-r} = Fx_{w-r} + Fx_{roll-r} \tag{4.2}$$

The resultant driving force is then given by adding the forces acting on the vehicle, as they are signed to oppose against the direction of travel:

$$Fx_{accel} = Fx_{road-r} + Fx_{road-f} + Fx_{drag} + Fx_{roll} + Fx_{grad} + Fx_{suspn} \tag{4.3}$$

The acceleration is simply calculated by dividing by the mass (4.4):

$$a_v = \frac{Fx_{accel}}{m_{total}} \tag{4.4}$$

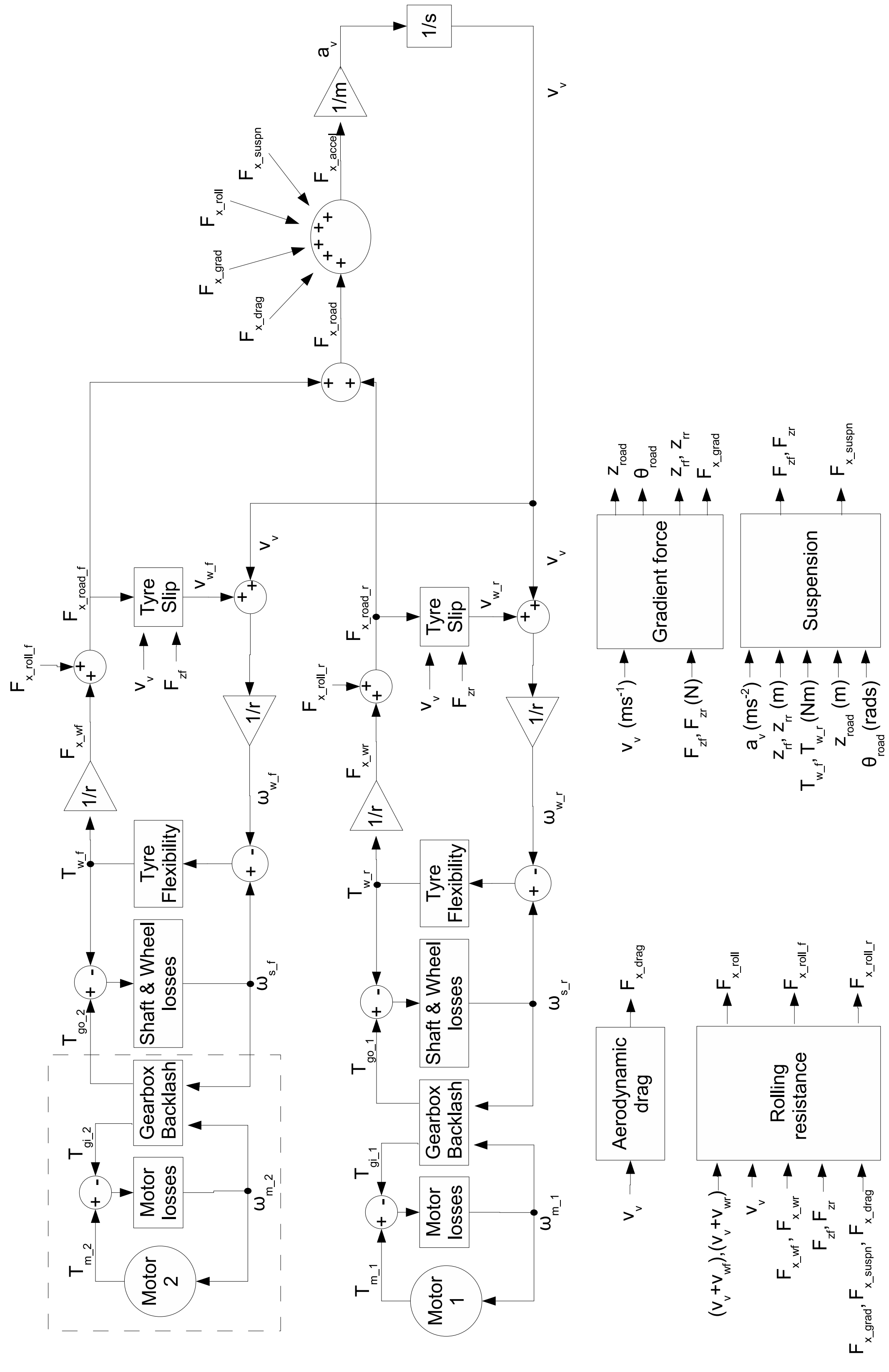

Figure 4.2: Complete vehicle model - with separate front and rear dynamics

The above diagram shown in Fig. 4.2 contains the entire vehicle model. It is often easier to represent the drivetrain with a rotating parts model shown in Fig. 4.3, in this case the front and rear, and left and right, have been lumped into a single lumped axle. Within this digram there are a number of new parameters defined: torque $T$, rotational speed $\omega$, linear force $F$, linear velocity $v$, torsional stiffness $k$, torsional damping $c$, inertia $J$, frictional loss $b$ and backlash angle $\alpha_{bl}$. The following subscripts are also used: motor $_m$, gearbox-input $_{gi}$, gearbox-output $_{go}$, gearbox $_g$, backlash $_b$, wheel $_w$, vehicle $_v$, gradient $_{grad}$ and suspension $_{suspn}$.

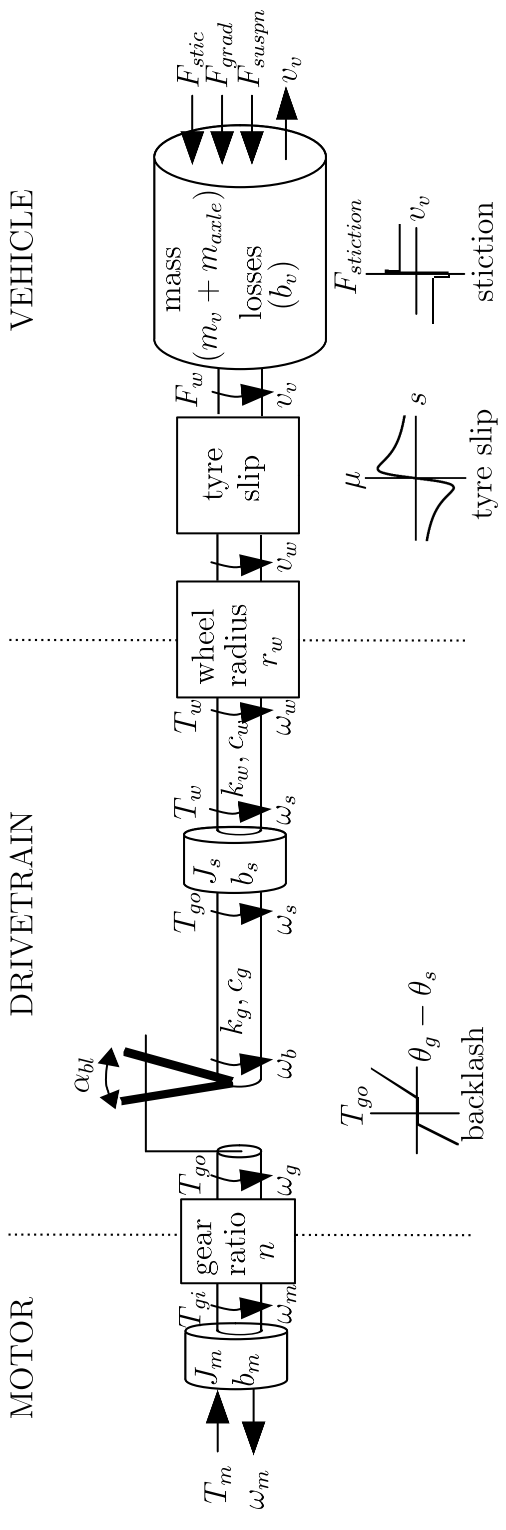

Figure 4.3: Complete (but simplified) vehicle model shown as rotating parts

The vehicle model can be split into a few key sections, which makes up the remainder of this chapter: drivetrain 4.2, vehicle losses 4.3, suspension 4.4 and tyre slip 4.5.

## 4.2 Drivetrain

### 4.2.1 Drivetrain Overview

The drivetrain includes all rotating parts of the vehicle's transmission, from the motor to the drive wheels. For most electric vehicles it commonly includes a combined fixed ratio gearbox and differential to then drive either the front or rear wheels. No parts within the drivetrain section are assumed to be affected by cornering dynamics, or external forces to the vehicle, and all forces take the form of torque $T$ in $Nm$ and speeds $\omega$ in $rads^{-1}$. Depending upon the subsection of the drivetrain the torque and speeds will be subscripted with $m$ for motor, $g$ for gearbox, $s$ for driveshaft and $w$ for wheel.

### 4.2.2 Typical Electric Vehicle Drivetrains

The drivetrain of different electric vehicles can vary significantly from one to the other, in terms of whether a gearbox is used or the number of traction motors present, this section will give a brief comparison of some common electric vehicles: an industrial fork lift truck, an on-road scooter and an on-road quadricycle.

Key for diagrams are: M=motor, W=wheel, G=gearbox

#### 4.2.2.1 Fork lift truck - industrial vehicle

The fork lift truck is considered first, it uses 2 traction motors to drive both the left and right front wheels though their own fixed ratio gearbox, as shown in Figs. 4.4, 4.5, 4.6 and 4.7 .

Figure 4.4: Fork lift truck - photo

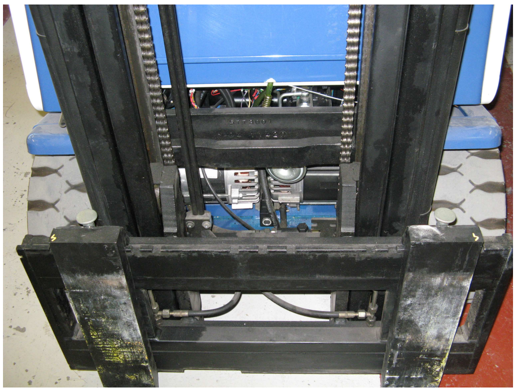

Figure 4.5: Fork lift truck - drivetrain photo

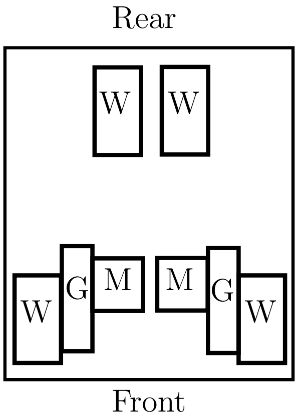

Figure 4.6: Fork lift truck - diagram

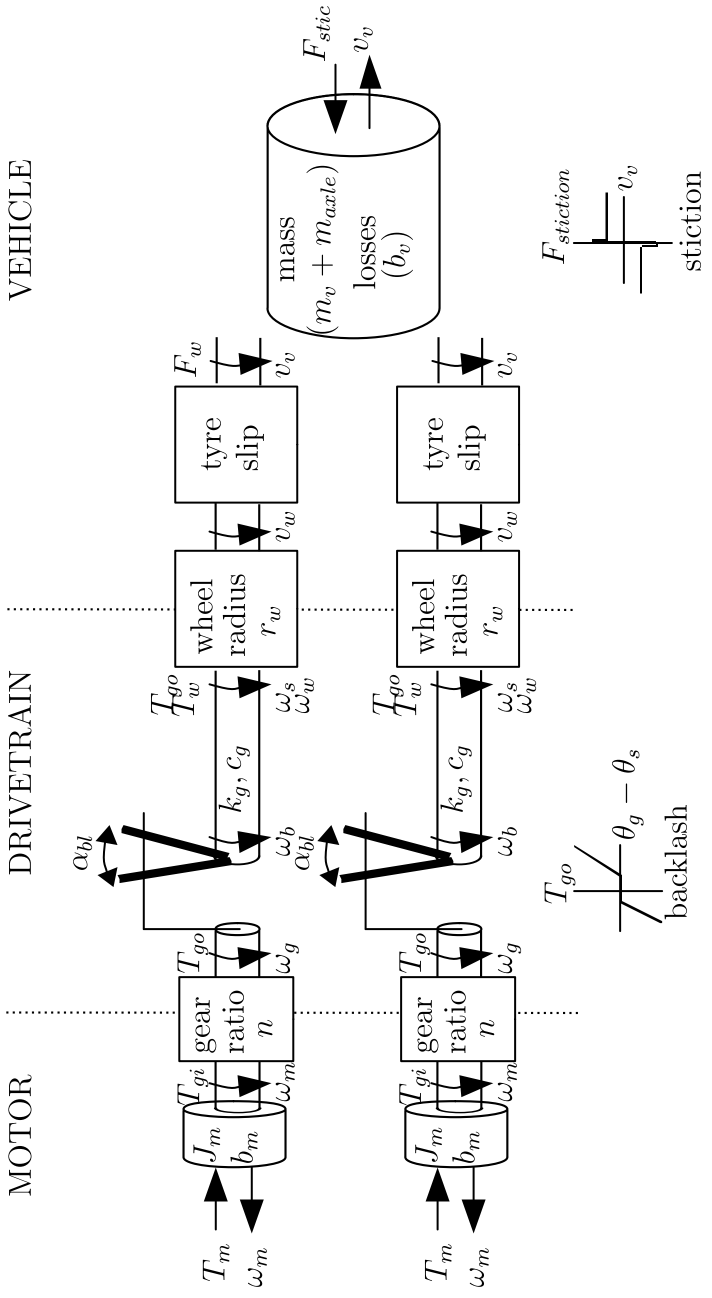

Figure 4.7: Fork lift truck - rotating model

#### 4.2.2.2 Scooter - on-road vehicle

The scooter has no gearbox, and has the simplest drivetrain. As it has a wheel mounted direct drive motor, there is only tyre slip separating the motor inertia from that of the vehicle, as shown in Figs. 4.8, 4.9 and 4.10.

Figure 4.8: Direct drive scooter - photo

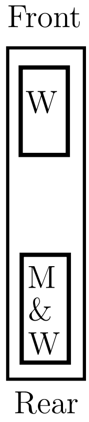

Figure 4.9: Direct drive scooter - diagram

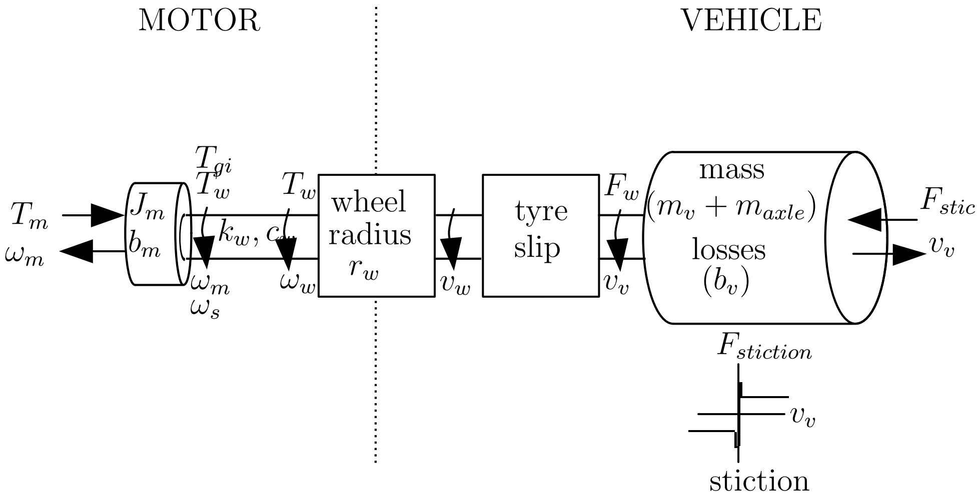

Figure 4.10: Direct drive scooter - rotating model

#### 4.2.2.3 Quadricycle - on-road vehicle

It is common to use a differential to apply equal torques to the left and right drive wheels whilst allowing for different rotation speeds for cornering. As motors have a large range of speeds where peak torque is possible, a fixed ratio gearbox is used allowing the motor to be mounted directly onto the differential, as shown in Figs. 4.11, 4.12 and 4.13. The overall drivetrain is similar to the vehicle used for the majority of testing in this work, refer to section 5.1.

Figure 4.11: Single traction motor rear wheel drive - photo

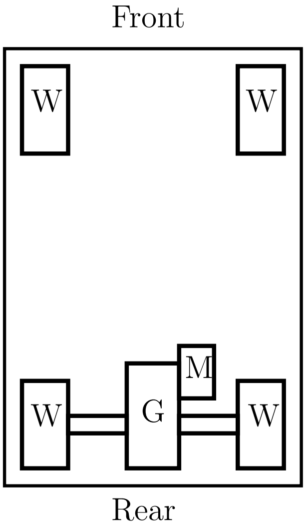

Figure 4.12: Single traction motor rear wheel drive - diagram

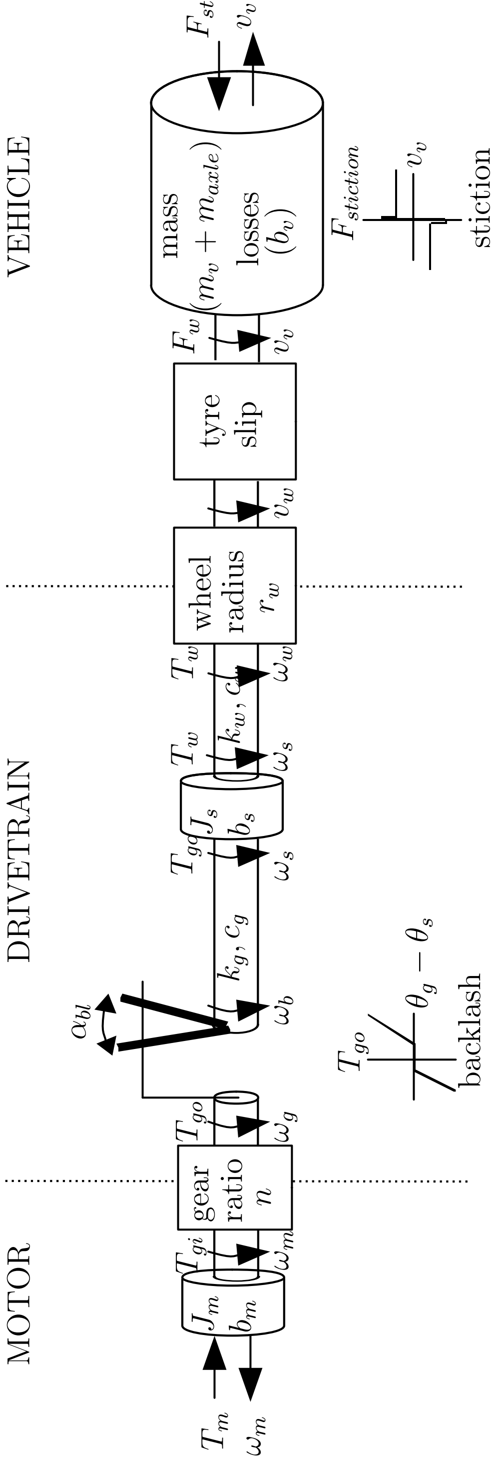

Figure 4.13: Single traction motor rear wheel drive - rotating model

### 4.2.3 Inertia, friction and stiction

All rotating parts within the drivetrain have an energy loss component and a kinetic energy storage component. The loss component is made up of the friction and stiction parts, and the energy storage is the inertia, as defined in (4.5). The friction part is represented as a torque linearly proportional to the rotation speed ($b$) and always opposing the direction of rotation. The stiction is a fixed loss torque $T_{stic}$ that opposes the direction of rotation, or if stationary opposes the applied torque, but cannot be greater in magnitude than the applied torque.

$$\omega = \int \frac{T_{in} - T_{stic} - b.\omega}{J} \mathrm{d}t \tag{4.5}$$

The stiction term $T_{stic}$ (4.6) can be equal to one of three possible values depending upon the magnitude of two other inputs: the applied torque $T_{in}$ and the rotational speed $\omega$. If $\omega$ is zero and the applied torque, $T_{in}$, is less than the stiction torque, $T_{stiction}$, then the stiction is simply equal to the applied torque $T_{in}$. If $\omega$ is zero and and the applied torque, $T_{in}$, is greater than the stiction torque, $T_{stiction}$, it is fixed at $T_{stiction}$ but signed depending upon applied torque direction. If $\omega$ is non-zero then the stiction is equal to $T_{stiction}$ but signed depending upon $\omega$.

$$T_{stic} = \begin{cases} T_{in}, & \text{if } \omega = 0 \text{ AND } T_{in} \leq T_{stic} \\ T_{stiction} \times sgn(T_{in}), & \text{if } \omega = 0 \text{ AND } T_{in} > T_{stic} \\ T_{stiction} \times sgn(\omega), & \text{if } \omega \neq 0 \end{cases} \tag{4.6}$$

The three stages that the stiction is applied are shown in figure 4.14; where stage (status) 1 equates to the first part of (4.6), stage 2 equates to the second part and stage 3 equates to the third part.

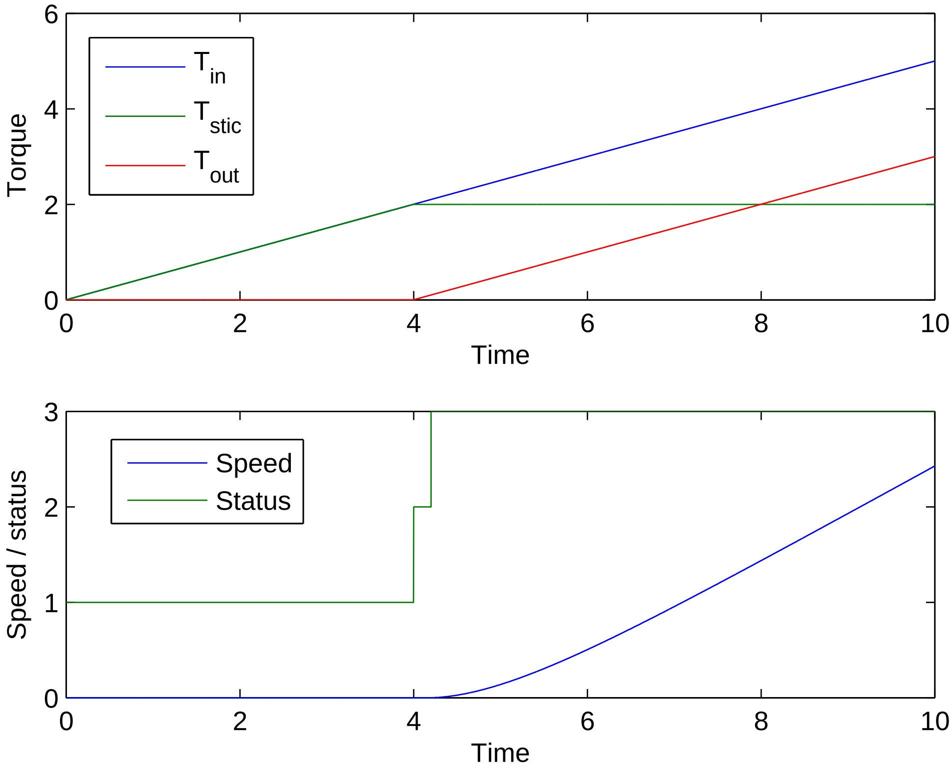

Figure 4.14: Stiction graph

There are two drivetrain parts that use the inertia friction model: motor and gearbox input shaft, and gearbox output shaft and wheel - but not including tyre rolling resistance losses.

### 4.2.4 Gear Backlash and Transmission Stiffness and Damping

The gearbox is used to increase the motor torque to the higher value required to accelerate the vehicle as traditionally higher torques required the motor to be larger and heavier, and so it was more appropriate to use a lower torque but higher speed motor. In the case of a differential the torque is split to the left and right wheels equally but allowing the wheels to rotate at different speeds for cornering. The gears inherently have play between the cogs, as this is required to reduce losses in the gearbox and minimise wear by allowing sufficient clearance. This gear play is commonly referred to as backlash and is a major cause of the 'non-smooth' response during torque direction transitions. This is because it causes the motor and vehicle to become disconnected for a brief period, allowing a large speed difference to develop as the motor accelerates quickly due to the large inertia differences between the motor and the vehicle. The collision between the cogs then rapidly transfers all of the motor kinetic energy (from the difference in speeds) into the vehicle giving a large change in vehicle acceleration or 'jerk' (rate of change in acceleration) in its response. A linear representation of gear cogs is shown in Fig. 4.15. The diagram shows what is meant by gear backlash.

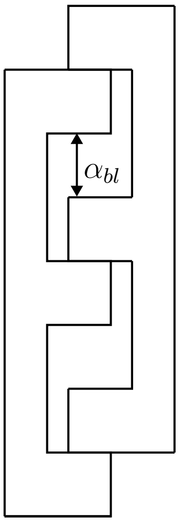

Figure 4.15: Gear cog diagram - drawn linear rather than circular

The gearbox model includes stiffness $k_g$ and damping $c_g$ terms, although as most of the flexibility will be within the drive shafts on the output of the gearbox, and not the gearbox itself, they can be lumped into single terms $k_g$ and $c_g$, as shown in Fig. 4.16. The gear ratio is applied on the motor side of the gearbox, before the backlash and flexibility components. This gives a conversion for gear input speed (4.7) and gear input torque (4.8), also shown in Fig. 4.16.

$$\omega_g = \frac{\omega_m}{n} \tag{4.7}$$

$$T_{go} = T_{gi} \times n \tag{4.8}$$

In the following Fig. 4.16 there are also equivalent position states $\theta$ (not shown) for each of the speed states $\omega$.

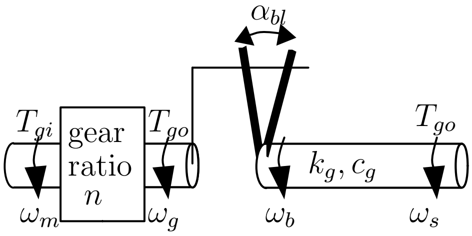

Figure 4.16: Backlash rotational diagram

If two additional states are introduced, $\theta_{bl}$ and $\theta_{ds}$, it can simplify the analysis of the gearbox. The angle within the backlash $\theta_{bl}$ is simply given by the difference in position of the two states either side of the backlash, $\theta_g$ and $\theta_b$, as shown in Fig. 4.16. This is calculated as (4.9). The total difference in angle across the drive shaft part of the drivetrain is given in $\theta_{ds}$. This is calculated as (4.10).

$$\theta_{bl} = \theta_g - \theta_b \tag{4.9}$$

$$\theta_{ds} = \theta_g - \theta_s \tag{4.10}$$

### Dead-zone gearbox model

The simplest representation of a backlash is the dead-zone model; where the gearbox output torque is set to zero when the position difference of the two inputs is within a set range, the dead-zone/backlash range. In some cases the damping term is calculated from the speed error of the two inputs speeds $\omega_g$ and $\omega_s$, but this leads to a torque output during the backlash period, which is incorrect as the inertia within the flexible part of the axle is assumed to be zero. The Fig. 4.17, shows an equivalent diagram for the dead-zone backlash model along with the diagram in Fig. 4.18.

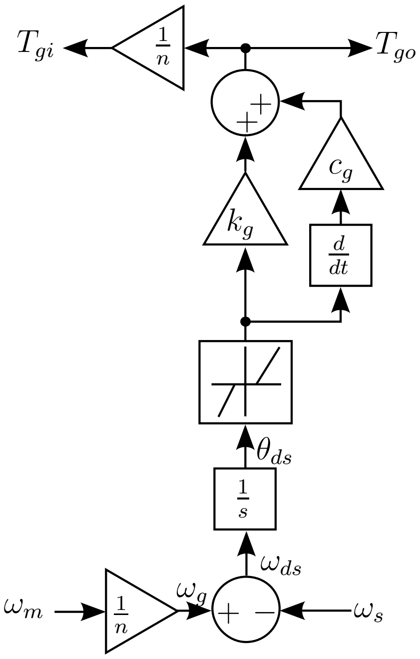

Figure 4.17: Dead-zone gearbox model

The dead-zone based model has 2 modes of operation, backlash $_{bl}$ and contact $_{co}$, which are determined by the following conditions (4.11):

$$\omega_{bl} = \begin{cases} \omega_{ds}, & \text{if } |\theta_{bl}| < \frac{\alpha_{bl}}{2} \\ 0, & \text{if } |\theta_{bl}| \geq \frac{\alpha_{bl}}{2} \end{cases} \tag{4.11}$$

The diagrams in Fig. 4.17 and the flow diagram in Fig. 4.18, gives a graphical equivalent of the equation above (4.11). The diagram in Fig. 4.18 is drawn to compare with the one of the physical model in Fig. 4.19.

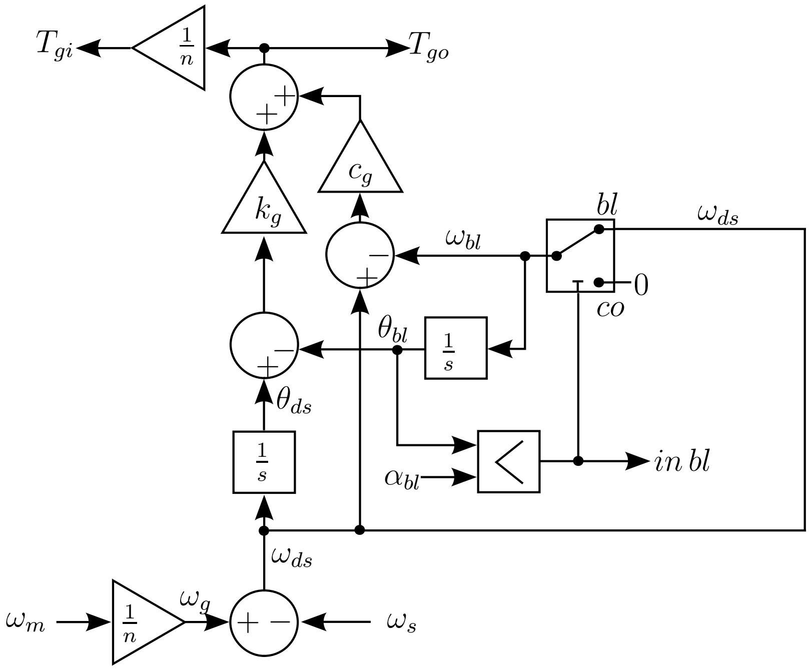

Figure 4.18: Dead-zone gearbox model (to compare with physical model)

If the damping term $c_g$ is very small the dead-zone model is valid, but at higher damping values, typical of real vehicles it is shown to be incorrect. Figs. 4.23, 4.26 and 4.29 show a comparison discussed in section 4.2.5. The main disadvantages with the dead-zone model is that as the damping term is incorrectly modelled the output torque and the indication when in the backlash is incorrect, especially during transients.

### Physical gearbox model

A more accurate model was developed by M. Nordin [78], it is based on the physical representation of the gearbox and accurately models transient conditions for systems with damping. Unlike the dead-zone model above (4.11), this model adds a speed contribution from the damping to $\omega_{ds}$ when in the backlash to give (4.12) [9]:

This effectively means that when in the backlash ($-\frac{\alpha_{bl}}{2} < \theta_{bl} < \frac{\alpha_{bl}}{2}$) the speed within the gap is governed by: $\omega_{ds} + \frac{k_g}{c_g}(\theta_{ds} - \theta_{bl})$, which is the total speed difference across the drive shaft and also a speed contribution from the shaft flexibility.

$$\omega_{bl} = \begin{cases} max(0, \omega_{ds} + (\frac{k_g}{c_g})(\theta_{ds} - \theta_{bl})), & \text{if } \theta_{bl} = -\frac{\alpha_{bl}}{2} \\ \omega_{ds} + (\frac{k_g}{c_g})(\theta_{ds} - \theta_{bl}), & \text{if } |\theta_{bl}| < \frac{\alpha_{bl}}{2} \\ min(0, \omega_{ds} + (\frac{k_g}{c_g})(\theta_{ds} - \theta_{bl})), & \text{if } \theta_{bl} = \frac{\alpha_{bl}}{2} \end{cases} \tag{4.12}$$

This can be simplified to (4.13) if $\omega_1 = \omega_{ds} + (\frac{k_g}{c_g})(\theta_{ds} - \theta_{bl})$ [9]:

$$\omega_{bl} = \begin{cases} max(0, \omega_1), & \text{if } \theta_{bl} = -\frac{\alpha_{bl}}{2} \\ \omega_1, & \text{if } |\theta_{bl}| < \frac{\alpha_{bl}}{2} \\ min(0, \omega_1), & \text{if } \theta_{bl} = \frac{\alpha_{bl}}{2} \end{cases} \tag{4.13}$$

When the cogs are meshed ($\theta_{bl} = -\frac{\alpha_{bl}}{2}$ or $\frac{\alpha_{bl}}{2}$), $\omega_{bl}$ is usually zero unless the cogs are in the process of unmeshing: $\omega_{ds}$ must be greater in magnitude than the speed contribution from the shaft flexibility $\frac{k_g}{c_g}(\theta_{ds} - \theta_{bl})$ for this to happen. If the damping $c_g$ is large, it is possible for the backlash to be entered whilst the shaft is still twisted [9], as shown in Fig. 4.23.

The flow diagram in Fig. 4.19, gives a graphical equivalent of the equation above (4.12), where $\omega_1 = \omega_{ds} + (\frac{k_g}{c_g})(\theta_{ds} - \theta_{bl})$.

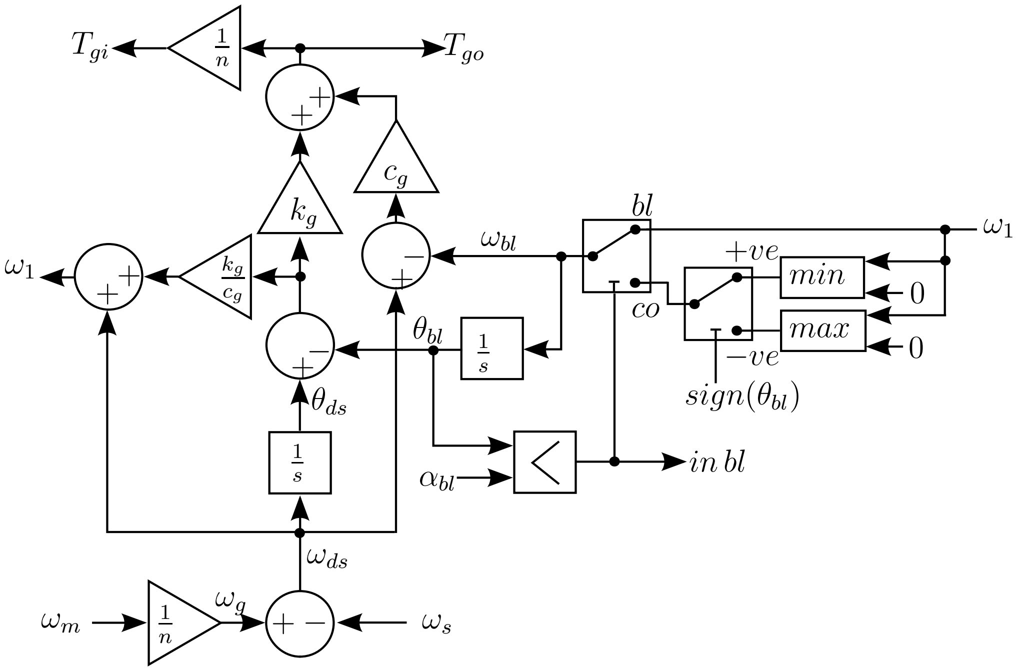

Figure 4.19: Physical gearbox model

The torque output is then given by (4.14), which is applicable to both gearbox models (dead-zone and physical):

$$T_{go} = k_g(\theta_{ds} - \theta_{bl}) + c_g(\omega_{ds} - \omega_{bl}) \tag{4.14}$$

Which can be simplified to (4.15), when applying equations (4.9) and (4.10):

$$T_{go} = k_g(\theta_b - \theta_s) + c_g(\omega_b - \omega_s) \tag{4.15}$$

### 4.2.5 Comparison of Gearbox Models

#### 4.2.5.1 Simulation model

The two potential gearbox models are compared in simulation to select the most appropriate model. A simulation was constructed based around a two mass model, which will be derived later in this chapter, shown in Fig. 4.20.

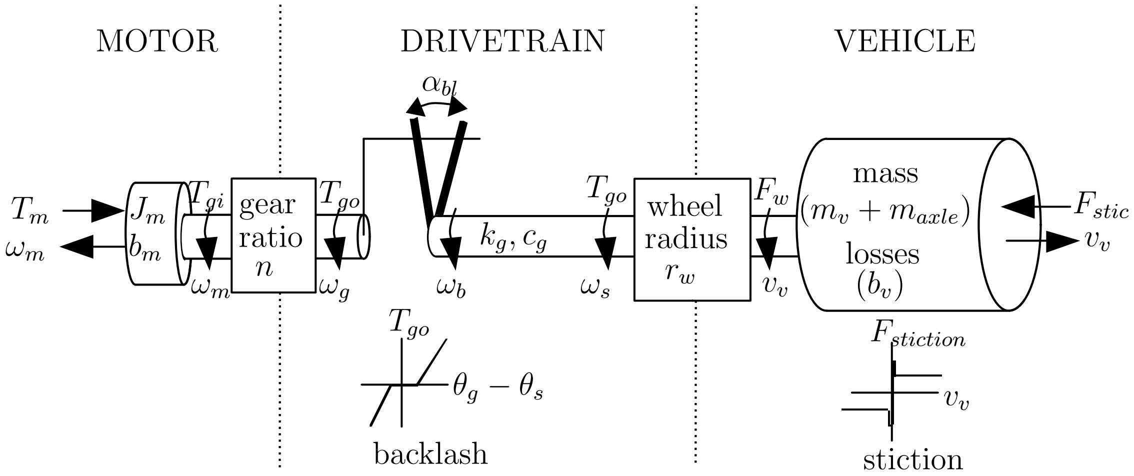

Figure 4.20: Rotating two mass model

The simulation is better shown in Fig. 4.21, where the block labelled gearbox backlash is replaced with either the dead-zone or physical model. The torques $T_{go}$ and $T_w$ are equal as this is only a two mass model and the speeds $\omega_s$ and $\omega_w$ are also equal.

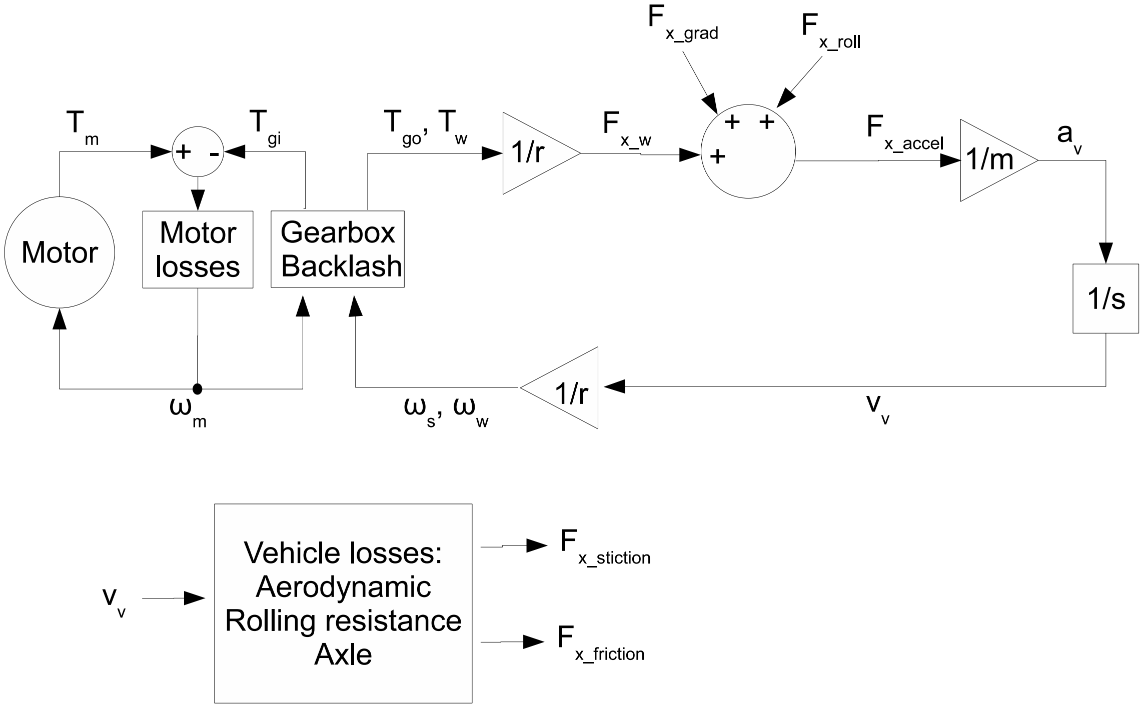

Figure 4.21: Simulation model for gearbox comparison

The simulation uses parameters similar to that of the test vehicle, see section 5.2, but with a number of different levels of axle/gearbox damping tested. A simple two-mass model was used with a fast torque ramp rate applied to the vehicle, initially applying a positive torque of $25Nm$ to accelerate the vehicle to $2000rpm$ and then a negative braking torque of $25Nm$ causing it to slow down; shown in Fig. 4.25 for the torque and speed profile with standard damping for that vehicle. Three damping levels are compared: practically zero damping, typical damping for the test vehicle used in this thesis and higher damping.

#### 4.2.5.2 Low damping

The first simulation is carried out with minimal axle damping ($c_g = 0.15Nm/rads^{-1}$), the speed response is shown in Fig. 4.22. The motor speed is seen to oscillate for nearly a second when accelerating and after the torque direction change there is almost $800rpm$ speed difference between the motor and vehicle speed (vehicle speed in this case has been converted back to an $rpm$ equivalent taking into account the gear ratio and tyre radius). The two gearbox models compared give an identical response suggesting that they are both suitable for gearbox modelling, shown in Fig. 4.23, and with a close up of the transient shown in Fig. 4.24.

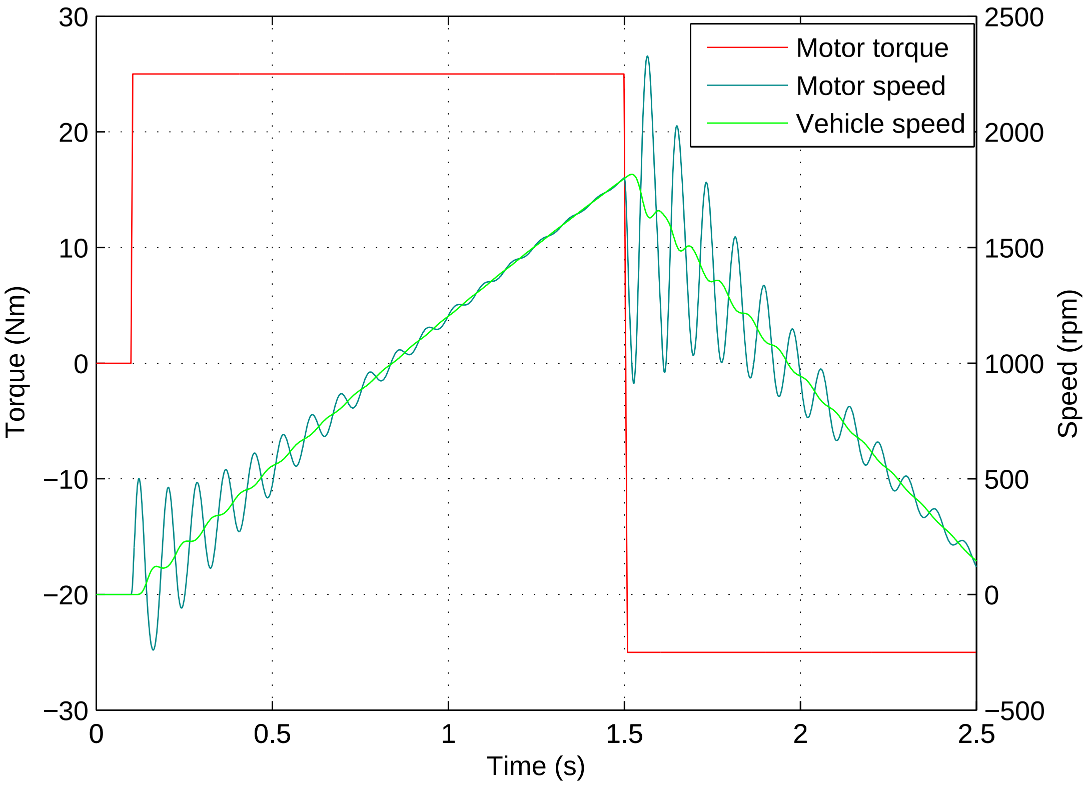

Figure 4.22: Gearbox model speed response graph with very low damping

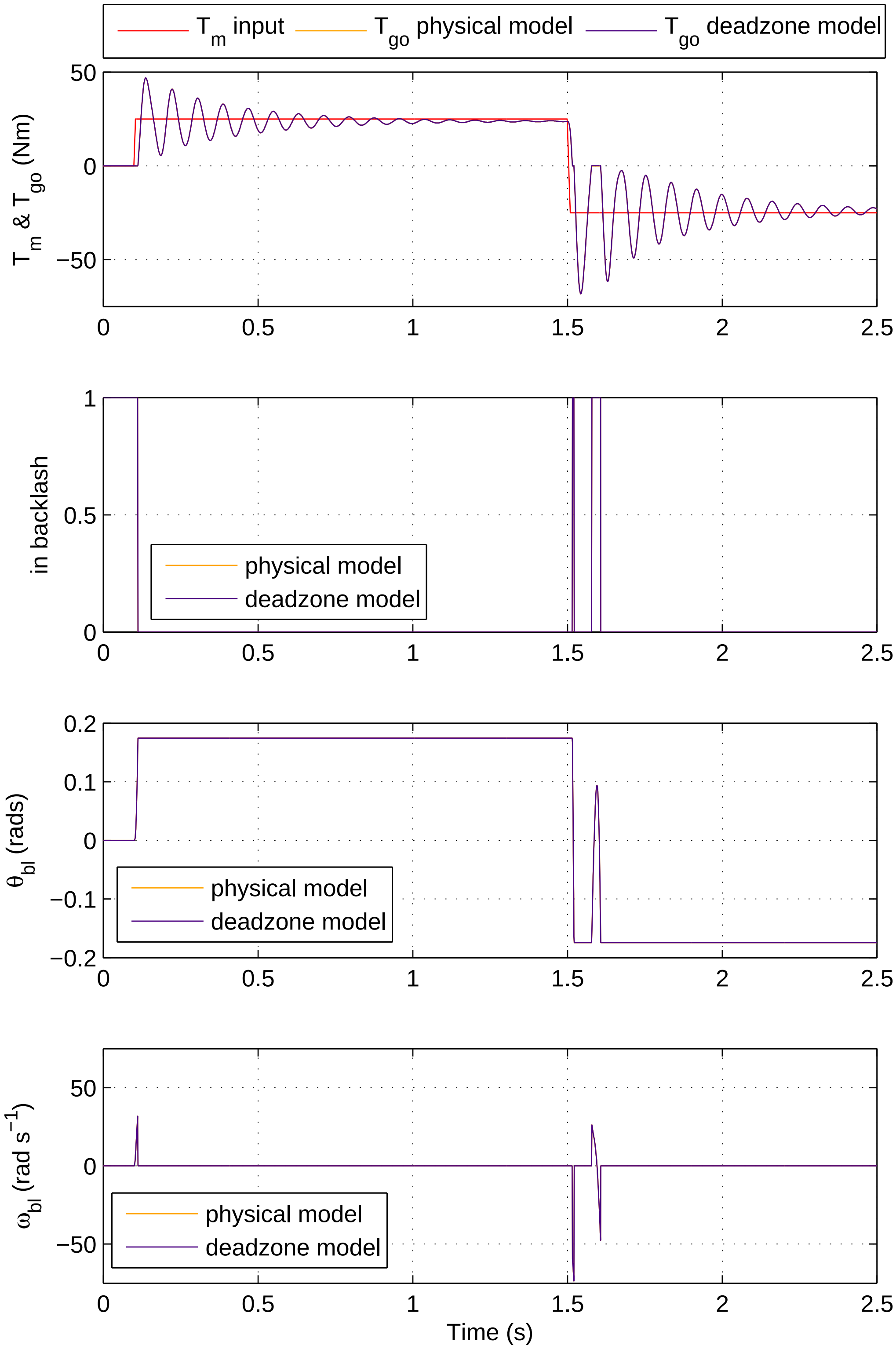

Figure 4.23: Gearbox model comparison graph with very low damping

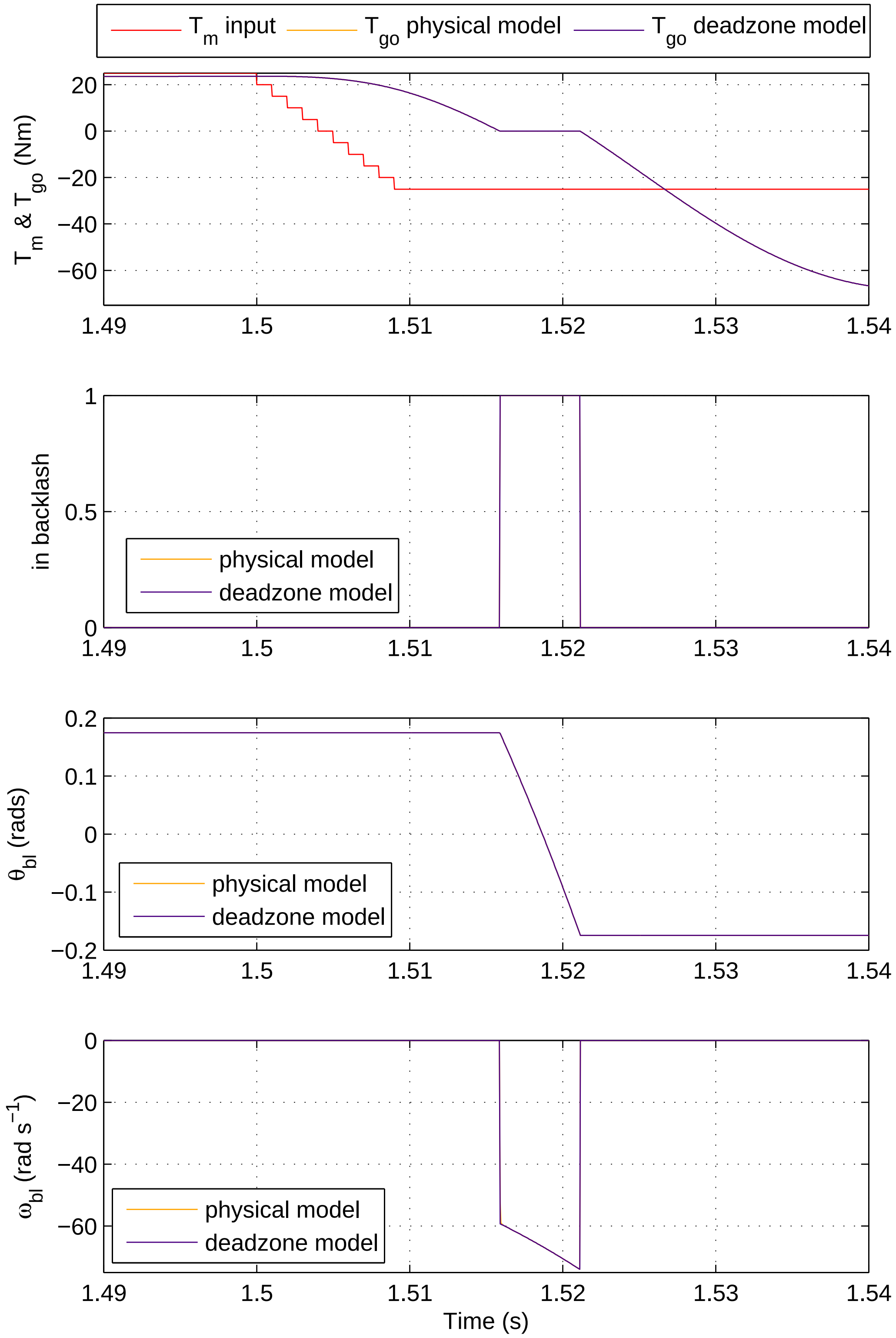

Figure 4.24: Gearbox model comparison graph with very low damping (zoomed around transient)

#### 4.2.5.3 Typical damping

With standard ($c_g = 15Nm/rads^{-1}$) damping, approximately at the level for the test vehicle used, the speed oscillation can be seen to reduce much faster in Fig. 4.25, than the case with no damping shown in Fig. 4.22. The two models, dead-zone and physical, look to give a fairly similar response when compared in Fig. 4.26, although the close up graph around the transient when moving in Fig. 4.27, shows a significant difference. Although when comparing the overall effect of the vehicle the change in models is fairly small, the motor and vehicle speed oscillations look the same, it is important that the time entering the backlash can be accurately predicted. The dead-zone model gives an incorrect backlash entry time due to the errors within the model.

The initial transient when starting to move at 0.1 seconds is identical, this is because there is no energy stored in the axle when stationary, so the damping has no effect when entering or exiting the backlash.

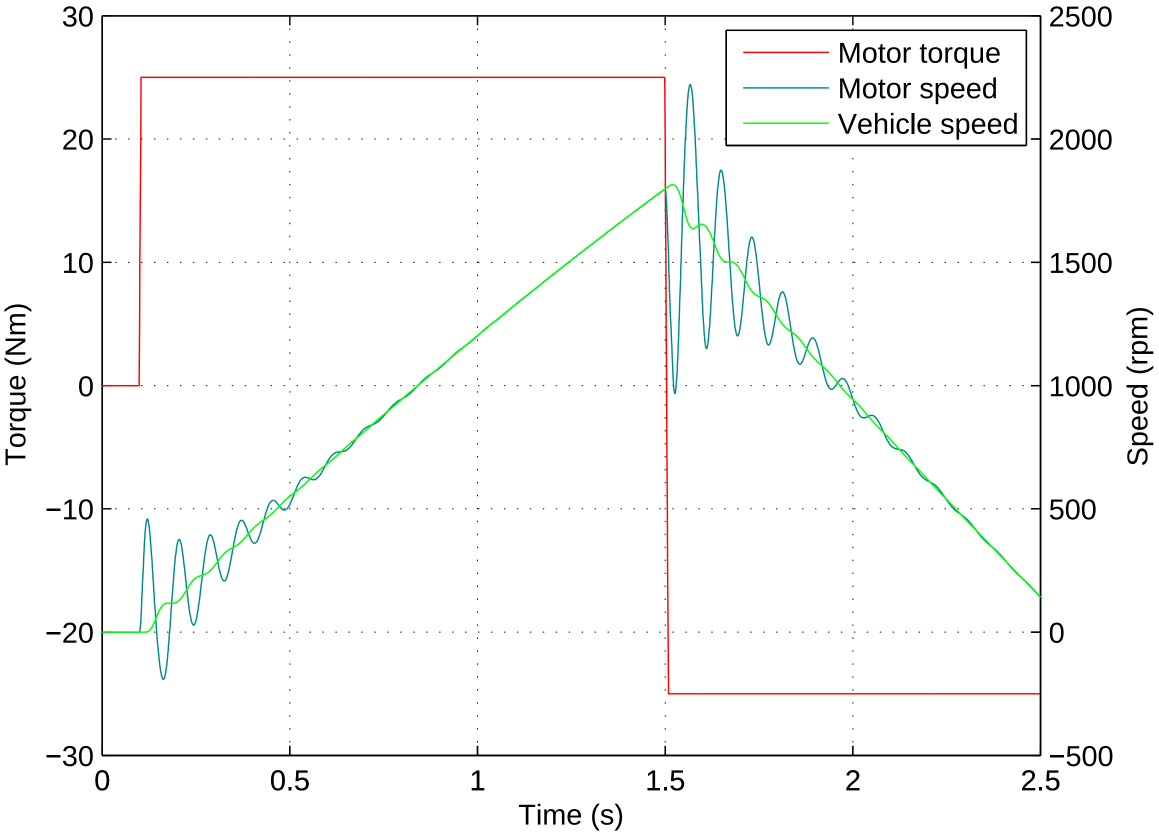

Figure 4.25: Gearbox model speed response graph with typical damping

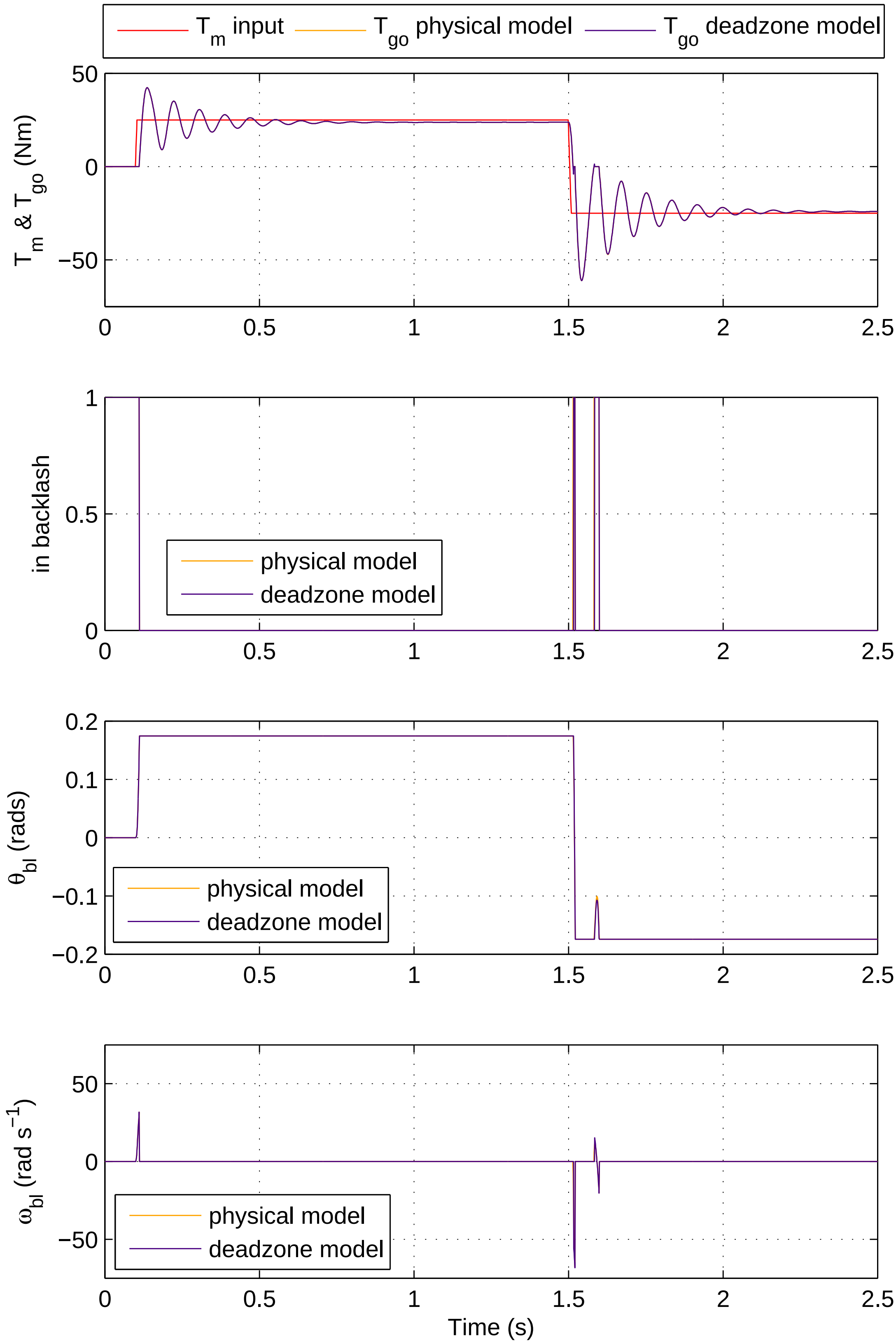

Figure 4.26: Gearbox model comparison graph with typical damping

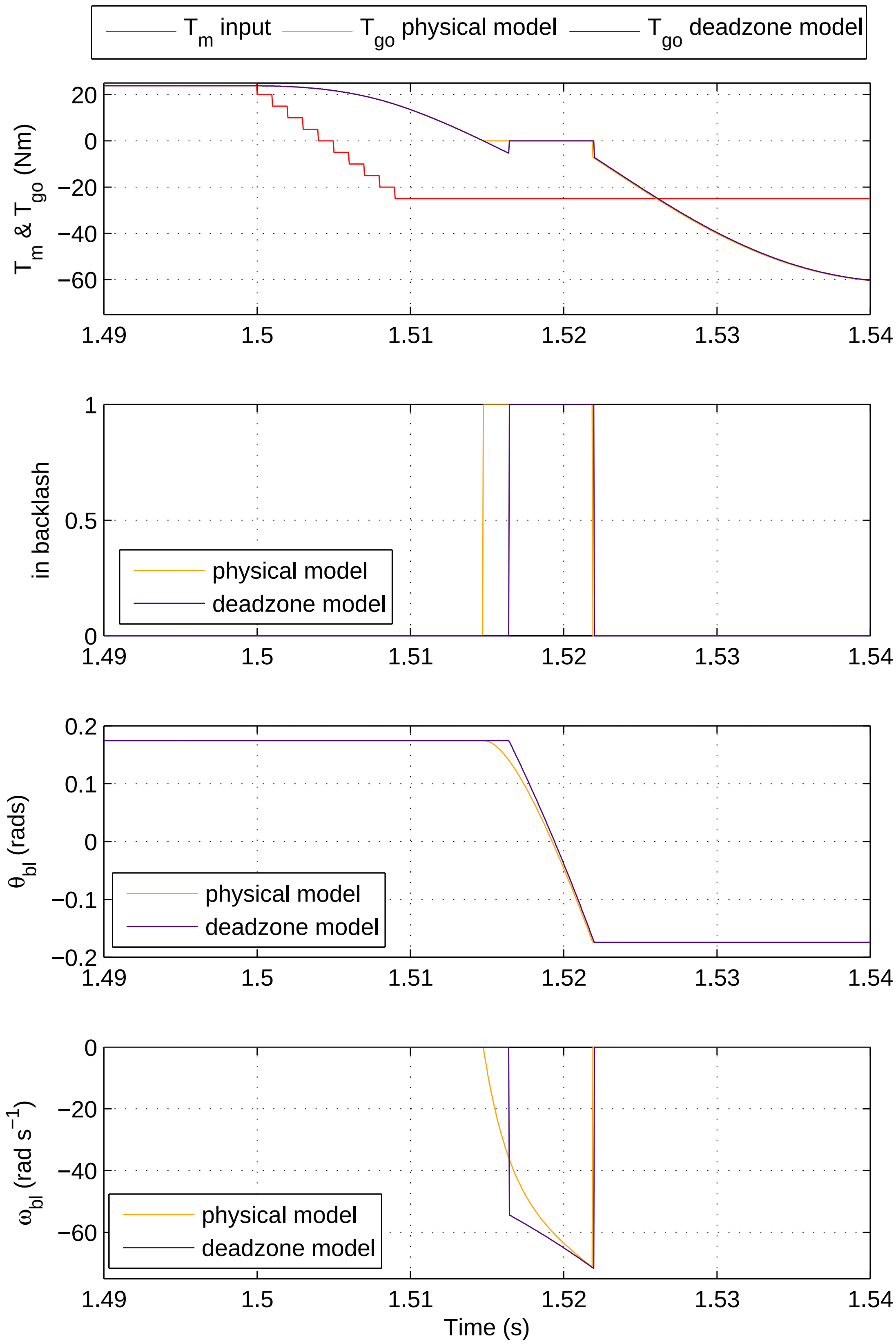

Figure 4.27: Gearbox model comparison graph with typical damping (zoomed around transient)

#### 4.2.5.4 Higher damping

With higher levels of damping ($c_g = 150Nm/rads^{-1}$), the speed response graph shows practically no oscillation, as shown in Fig. 4.28. Although the damping level for this test has been increased greatly to emphasise the effect, other vehicles will have damping levels higher than the golf buggy test vehicle and so will have a response between the previous test level ($c_g = 15Nm/rads^{-1}$) and the level in this test ($c_g = 150Nm/rads^{-1}$).

The increase in damping has caused a significant difference between the models, shown in the graph in Fig. 4.29 and a close up of the torque reversal transient is shown in Fig. 4.30.

One of the most significant errors with the dead-zone model is that a torque in the opposite direction is seen before crossing the backlash as shown in Fig. 4.30, which of course is impossible as it would suggest a pull force between the cogs [78]. This can also be seen to a lesser extent in the graph in Fig. 4.27 for the case with typical damping.

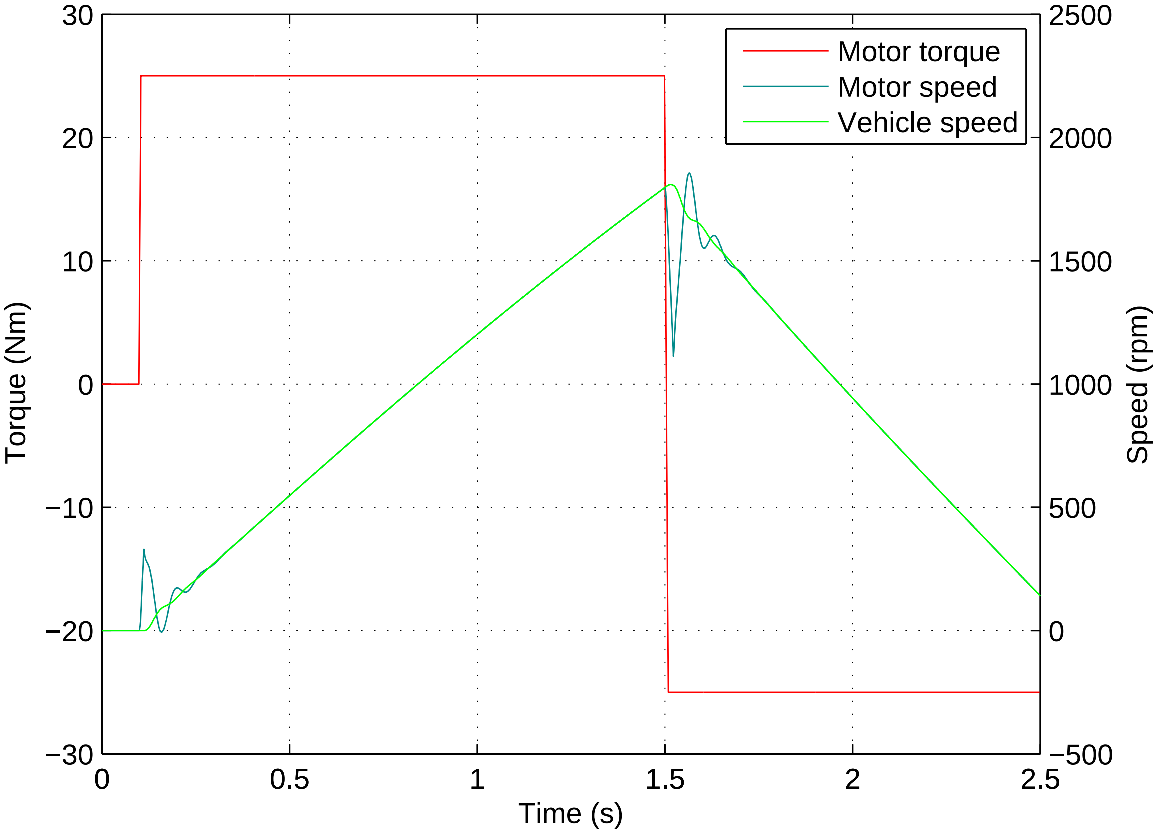

Figure 4.28: Gearbox model speed response graph with relatively high damping

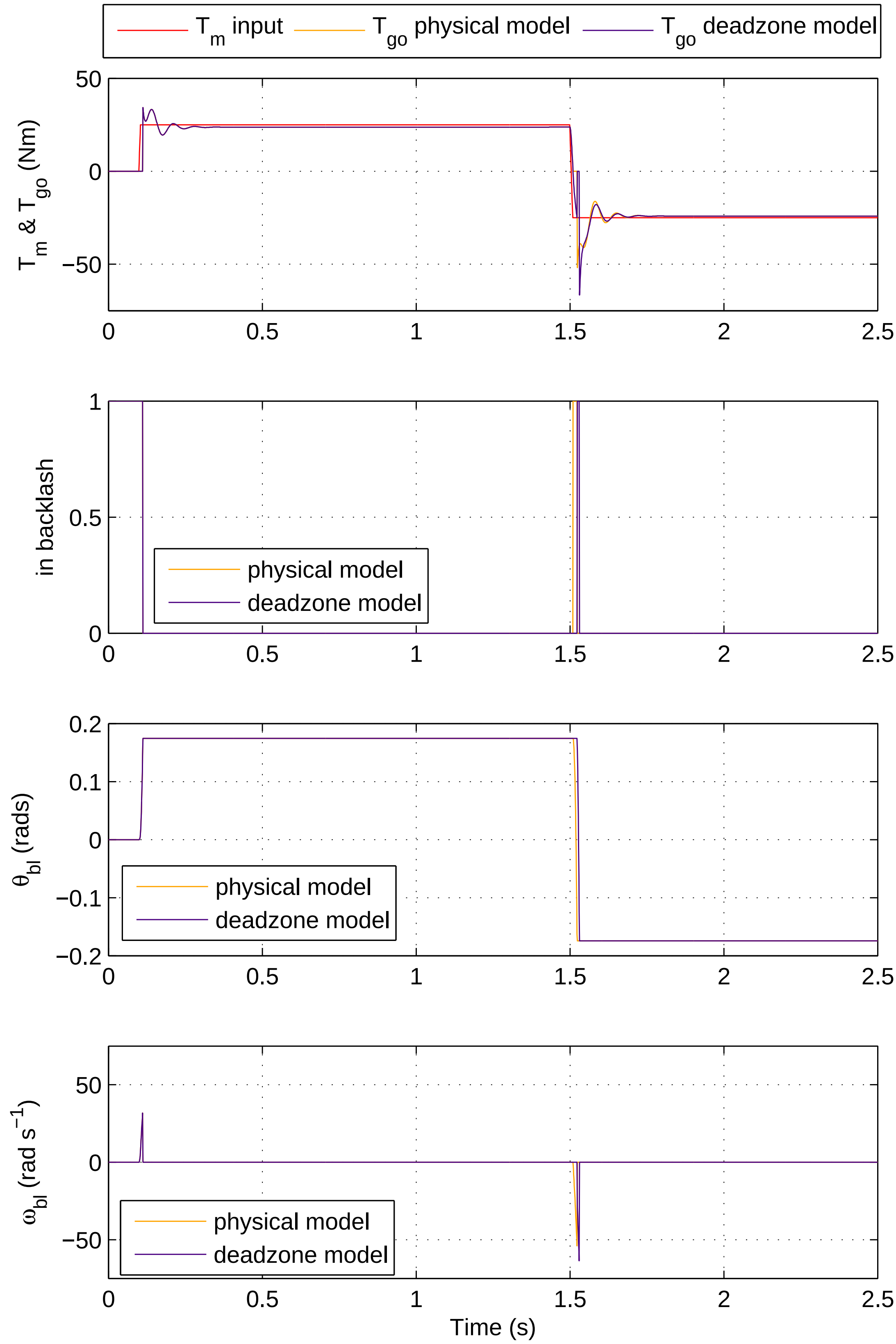

Figure 4.29: Gearbox model comparison graph with relatively high damping

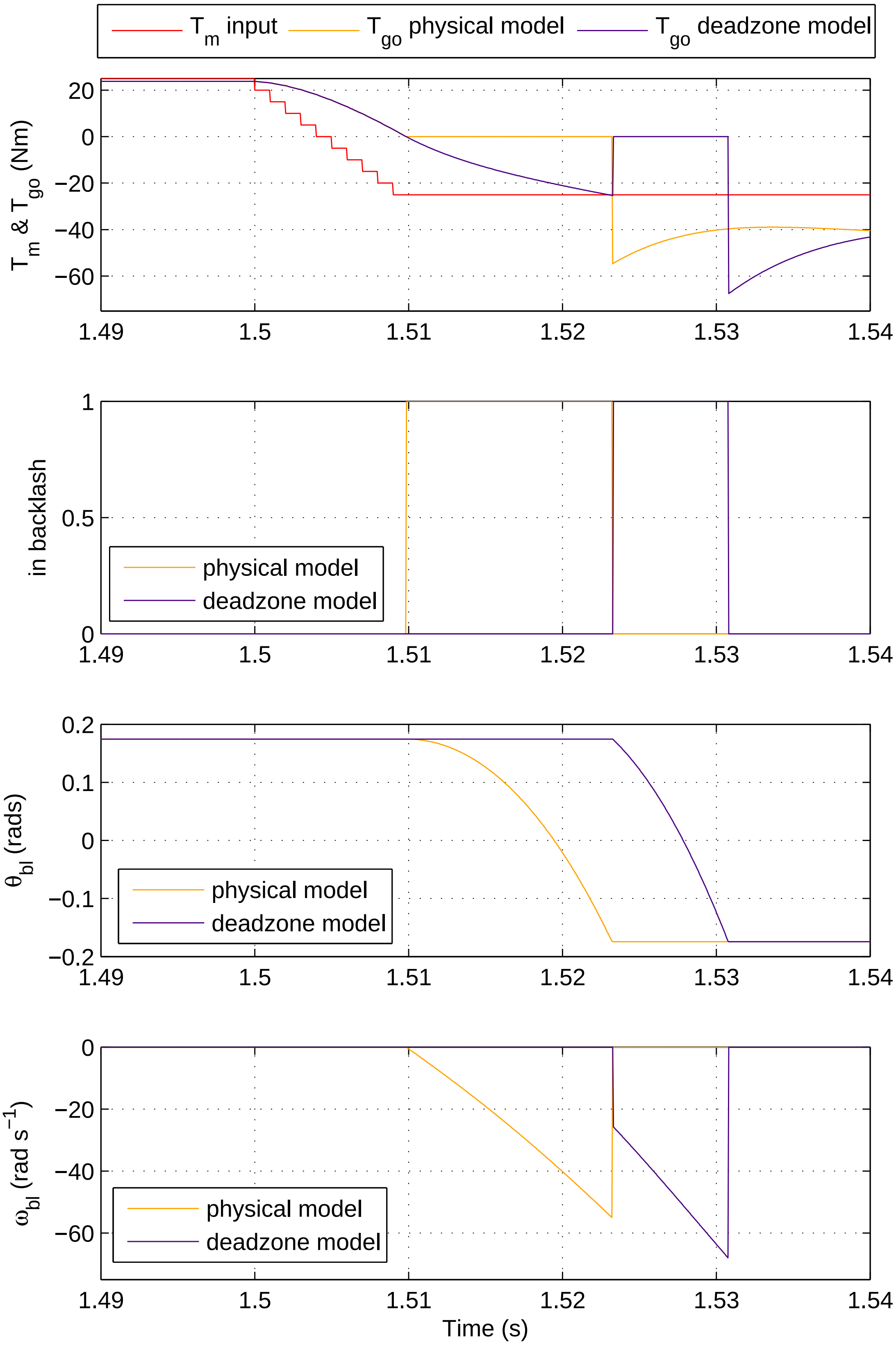

Figure 4.30: Gearbox model comparison graph with relatively high damping (zoomed around transient)

#### 4.2.5.5 Comparison

The most appropriate model to use is the physical model as it correctly represents the dynamics of the gearbox. Two errors were seen in the dead-zone gearbox simulation, both of which which increased in magnitude the higher the damping was. The first error is the time the backlash is entered and its duration, which is incorrect for the dead-zone model as it estimated that the backlash is entered later and for a shorter time. This is shown in the typical damping case in Fig. 4.27 and in the higher damping case shown in Fig. 4.30. Backlash entry time is important for the scheme discussed later in the thesis in section 7.5. Additionally the dead-zone model shows a pull force between the cogs before entering the backlash which is not physically possible. There is greater calculation required for the physical model but the difference is quite small when compared to the dead-zone model.

### 4.2.6 Tyre Flexibility

Similar to the flexibility of the gearbox and axle, the tyres will also be able to deform when a torque is applied across them. This is defined as a single stiffness $k_t$ and damping term $c_t$. Generally this value is much higher than the axle flexibility and in some instances can be ignored. When the vehicle parameters are found in the next chapter the magnitude of $k_g$ (gear/axle stiffness) and $k_t$ (tyre rotational stiffness) will be compared to see if the effect of $k_t$ should be included in the model.

## 4.3 Vehicle Losses

### 4.3.1 Overview

The drivetrain outputs a torque at the wheels, which has a number of losses subtracted before the resultant torque acting on the vehicle is determined. There are four main forces that act upon the vehicle, these are aerodynamic drag, rolling resistance, the suspension and a gradient force, as shown in Fig. 4.31. The drag force is created by the vehicle having to move through stationary (or moving if windy) air, the rolling resistance is caused by having to continually deform the shape of the wheels as different areas of the tyre contact the road surface, and the gradient force is caused by ascending or descending inclines against the force of gravity. The suspension force will be discussed in section 4.4, the force naming convention is given in Fig. 4.36. Most of the vehicle losses operate over three directional dimensions and three rotational dimensions, but this leads to an unnecessarily complicated model, as shown in Fig. 4.1. Most of the vehicle losses have been approximated to only operate in the longitudinal forward driving direction (x axis), as the majority of the testing is done under these conditions. An additional approximation is shown in Fig. 4.31; when calculating the tyre rolling resistance, the vehicle z axis is always equal to the direction of gravity.

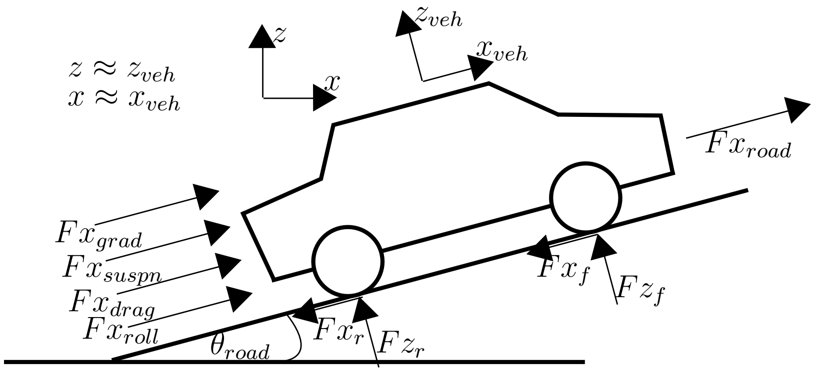

Figure 4.31: Vehicle loss directions

### 4.3.2 Aerodynamic Drag Force

As any vehicle is moving through air, it gives a viscous fluid resistance, the loss force is proportional to the square of its velocity, (4.16) [79]. If the vehicle is stationary but there is still wind acting in the linear (x axis) direction, it will still cause a positive or negative drag force, depending upon the direction. As the majority of the testing is done indoors, the wind velocity $v_w$ will be set to zero initially. Sidewinds in the y axis have also been ignored as these would normally cause a change in the vehicle's roll angle, changing the left and right suspension forces. Downforce has been assumed to be zero for the suspension model, so all drag forces will just cause a loss in the x direction, but no increase in the downwards (z axis) force.

$$F_{drag} = C_{d-f}.A_v.\rho.(v_v + v_{wind})^2.sgn(v_v + v_{wind}) \tag{4.16}$$

### 4.3.3 Road Gradients

Unless travelling along a level surface, gravity causes an additional force which either acts to accelerate or decelerate the vehicle depending upon the direction of travel and the incline angle. A lot of industrial vehicles only operate on flat terrain, such as Fork Lift Trucks (FLT) in warehouses or tow tractors at airports. Some such as mining vehicles operate over large inclines and with large load mass added and also on-road vehicles generally travel over continually varying gradients; this causes a significant change in the vehicle's dynamics.

The gradient model has inputs of front and rear force in the z axis. This can either be fixed depending upon the steady state weight distribution of the vehicle, or it can be calculated on-line from the suspension model, if increased accuracy is required; taking into account the changing vehicle weight distribution with changing vehicle speed and gradient. The other inputs are vehicle distance (this can be in both the x and z axis depending upon gradient) and the road height information. This can either be in the form of a level surface followed by a fixed gradient ramp for the indoor lab tests, or continuing varying gradient lookup table taken from GPS for the on-road tests.

In order to calculate the road gradient on the vehicle, the height of the roads at the front and rear axle of the vehicle are used (4.17), as shown in Fig. 4.32.

$$\theta_{veh} = \arcsin\left(\frac{z_{road-f} - z_{road-r}}{l_{total}}\right) \tag{4.17}$$

The gradient force is now calculated from trigonometry (4.18). The sum of the forces towards the road at the front and the back of the vehicle, ($Fz_f$ and $Fz_r$), will be equal to $m_{total} \times g$. The forces ($Fz_f$ and $Fz_r$) as shown in Fig. 4.31.

$$F_{grad} = (Fz_f + Fz_r).\sin(\theta_{veh}) \tag{4.18}$$

Or this can be simplified without calculating the road angle of the vehicle (4.19). There is a single output force applied to the entire vehicle for gradient force and only works for gradients in the vehicle's longitudinal direction.

$$F_{grad} = \ (Fz_f + Fz_r)\,.\left(\frac{z_{road-f} - z_{road-r}}{l_{total}}\right) \tag{4.19}$$

In order to obtain the road heights for the fixed angle ramp tests, the displacement in the z axis, $z_{road-f}$ and $z_{road-r}$, is calculated as follows (4.20):

$$z_{road-f} = \int v_v.\sin(\theta_f).dt \tag{4.20}$$

Where $\theta_f$ is set to the fixed angle if the front axle is between the $x_{grad-start}$ and $x_{grad-end}$, shown in Fig. 4.33. The same calculation is performed for the road height at the vehicle rear axle (4.21):

$$z_{road-r} = \int v_v.\sin(\theta_r).dt \tag{4.21}$$

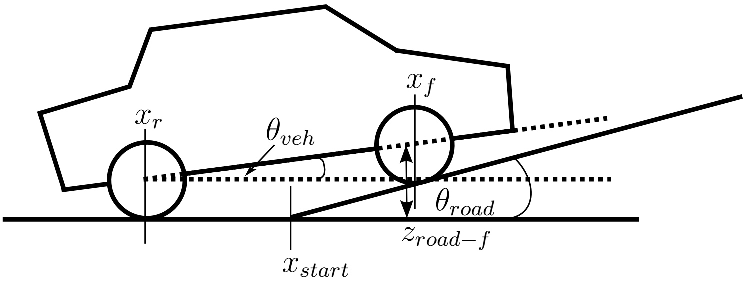

Figure 4.32: Gradient forces when partially on a fixed gradient ramp

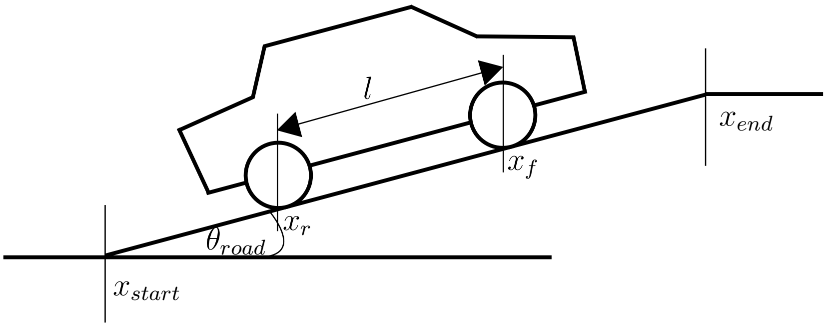

Figure 4.33: Gradient forces when on a ramp with wheelbase shown ($\theta_{veh} = \theta_{road}$)

### 4.3.4 Rolling Resistance

When the vehicle is moving, the tyres are continually being deformed, depending upon which part contacts the road; this causes a force acting against the direction of travel of the vehicle. The rolling resistance force is calculated from 3 coefficients, one fixed $C_{r0}$, one dependent on the velocity magnitude $C_{r1}$ and one dependent on the velocity square $C_{r2}$. They remain constant provided the tyre pressure and temperature is constant. This is unlikely during normal operation as the tyre will heat up, causing the pressure to increase and therefore the coefficients will decrease. This change is assumed to be quite small and fixed constants have been used throughout the simulation. No additional losses have been included for when the tyres are turned and the vehicle is going round a corner.

Rolling resistance is normally calculated on a per wheel basis, as the z axis force varies for each corner of the vehicle, and the rolling resistance varies linearly with the normal force applied to the wheels. If the model does not have independent wheel motors, or the differential is ignored, the z axis force can be combined to give less rolling resistance outputs.

When stationary an additional term $Fx_{roll}$ is used, which is the combined front and rear rolling resistance, and acts in the same place in the simulation as the aerodynamic drag, gradient and suspension longitudinal forces. This has been done due to the difficulty of calculating front and rear rolling resistance forces when there is also gradient forces acting on the vehicle as the gradient force acts on a per vehicle level and cannot be proportioned between wheels.

Rolling resistance when stationary and $|Fx_{drag} + Fx_{grad} + Fx_{road} + Fx_{suspn}| < (C_{r0} \times Fz)$ [80]:

$$\begin{aligned} Fx_{roll} = -(Fx_{drag} + Fx_{grad} + Fx_{road} + Fx_{suspn}) = -F_{wheel} \\ Fx_{roll-f} = 0 \\ Fx_{roll-r} = 0 \end{aligned} \tag{4.22}$$

This is signed so it opposes the resultant of the gradient, drag, wheel and suspension forces. Where $Fz = Fz_f + Fz_r$ and $Fx_{road} = Fx_{w-f} + Fx_{w-r}$

Rolling resistance when stationary and $|Fx_{drag} + Fx_{grad} + Fx_{road} + Fx_{suspn}| >= (C_{r0} \times Fz)$:

$$\begin{aligned} Fx_{roll} &= 0 \\ Fx_{roll-f} &= -C_{r0} \times Fz_f \times sgn(Fx_{drag} + Fx_{grad} + Fx_{road} + Fx_{suspn}) \\ Fx_{roll-r} &= -C_{r0} \times Fz_r \times sgn(Fx_{drag} + Fx_{grad} + Fx_{road} + Fx_{suspn}) \end{aligned} \tag{4.23}$$

This is signed to ensure it opposes the resultant force acting on the vehicle ($Fx_{drag} + Fx_{grad} + Fx_{road} + Fx_{suspn}$)

Rolling resistance when the vehicle is moving is calculated by [81]:

$$\begin{aligned} Fx_{roll} &= 0 \\ Fx_{roll-f} &= -(C_{r0} \times C_{r0-scaler} + C_{r1} \times |v_w + v_v| + C_{r2} \times (v_w + v_v)^2) Fz_f \times sgn(v_v) \\ Fx_{roll-r} &= -(C_{r0} \times C_{r0-scaler} + C_{r1} \times |v_w + v_v| + C_{r2} \times (v_w + v_v)^2) Fz_r \times sgn(v_v) \end{aligned} \tag{4.24}$$

This is signed to ensure it opposes the direction of travel. When moving $C_{r0}$ is multiplied by $C_{r0-scaler}$ to take into account the fact that the stiction decreases once the vehicle is moving.

It is common for the rolling resistance to be reduced by the cosine of the road gradient [79]; to take into account that gravitational force acts perpendicular to a flat road surface and any gradient will mean that the vehicle z axis will no longer align with this gravitational axis. The effect of this is quite minimal though and for a typical 10 degree gradient this effect is equivalent to a 1.5% reduction in force, so it has been ignored.

## 4.4 Suspension

### 4.4.1 Half Vehicle Model

The vehicle's suspension has an effect on the dynamics of most other subsections of the model and has a large impact on how the vehicle drives. The main effect is caused by the change in front and rear normal forces at the tyres, due to acceleration or braking. This change in normal force determines the amount of grip that is available; it therefore controls, along with the road surface coefficient whether the wheels will lose traction and move at a different speed relative to the road (see section 4.5). The wheel normal force also determines the gradient force and the rolling resistance losses. Lastly the suspension causes a longitudinal force when the vehicle acceleration changes direction, caused by the action of weight transfer during acceleration and braking. The suspension, especially on vehicles with a low suspension stiffness and a high centre of gravity, can cause the vehicle to roll back after heavy braking. This happens when the front suspension becomes compressed and the weight (and therefore centre of gravity) is transferred forward during braking. When the braking stops, this weight is then transferred back to the steady state position. Some of this force will act in the longitudinal direction, causing the vehicle to roll back.

Some parts of the suspension dynamics have been ignored in order to simplify the simulation and analysis. The following have been assumed to be insignificant for the model: the effect of the anti-roll bar - transferring force from the left side of the vehicle to the right or vice versa when cornering; the change in normal force with gradients reducing rolling resistance - only depends on the cosine of the gradient (therefore quite small); and any downforce effects - caused by aerodynamic drag leading to increased normal force or in-balance in the front and rear of vehicle normal forces.

As only a half vehicle model is used the suspension only simulates rotation about the y axis (shown as vehicle pitch angle in Fig. 4.1), shown in Figs. 4.34 and 4.36, giving a force change to the front and rear. The left and right force change when cornering, giving a rotation around the x axis (vehicle roll angle) has not been included as the focus has been on longitudinal driving tests. The suspension has inputs of road height (front $z_{road-f}$, rear $z_{road-r}$ and below the Centre of Gravity (C.O.G) $z_{road}$) from the gradient subsystem and vehicle acceleration ($a_v$) and torques ($T_{wf}$ and $T_{wr}$) from the main system.

$$h = h_{steady-state} + (z_v - z_{road}) \tag{4.25}$$

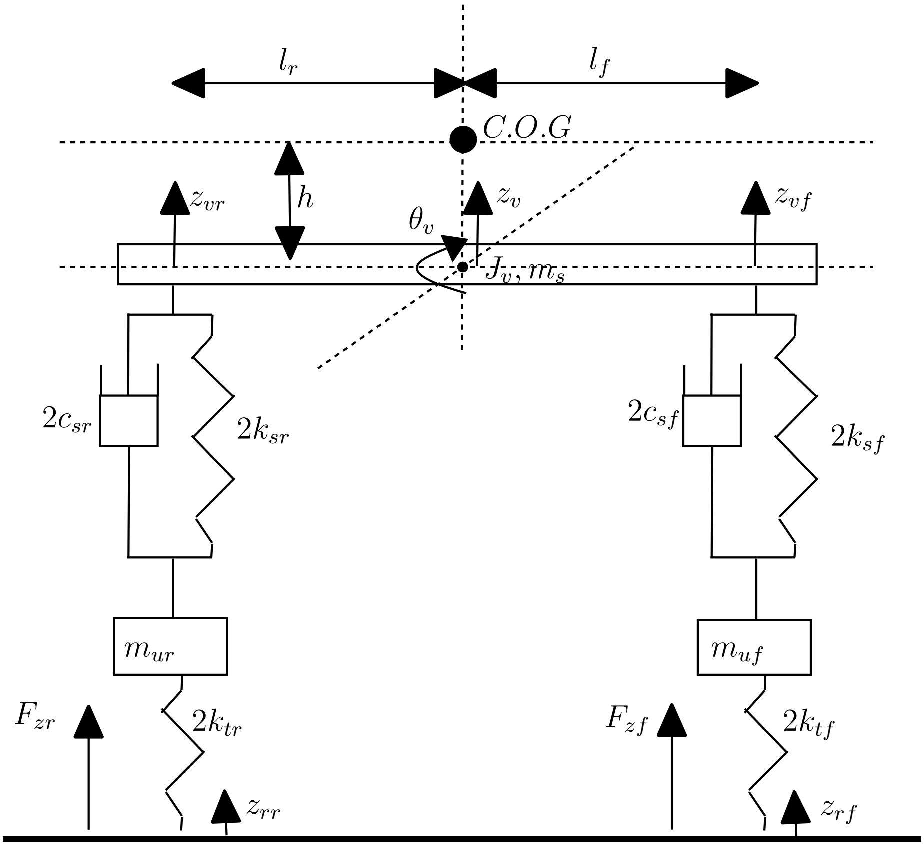

Figure 4.34: Half-car suspension model

The suspension simulation is based upon the diagram in Fig. 4.34. The vehicle suspension is only taken into account in the longitudinal driving direction (x axis) and so is represented by two springs with damping, front and rear. The tyres are also represented by two springs, although with a much larger stiffness and no damping. The mass of the vehicle and its inertia around its centre of gravity determines how much the front, centre and rear of the vehicle are displaced ($z_{vf}$, $z_v$, $z_{vr}$). The z axis is always assumed to be in the same direction as gravity, even when on gradients where there will be slight difference between the two (depending on how steep the gradient is). Similarly with the x axis it is always assumed to be the driving direction and also perpendicular to gravity. This is shown in Fig. 4.35 below.

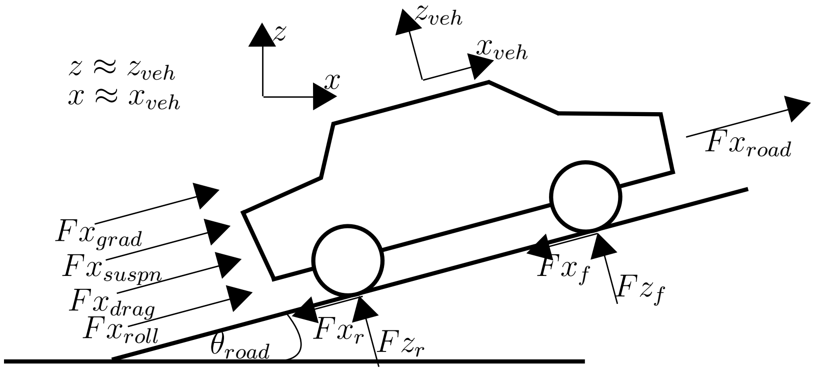

Figure 4.35: Forces with a vehicle on a gradient

The suspension forces are calculated by the following equations [82, 83], (4.26) and (4.27):

$$Fz_{sn-f} = -2 \times k_{sn-f}\left(z_{v-f} - z_{usp-f}\right) - 2 \times c_{sn-f}\left(\dot{z_{v-f}} - \dot{z_{usp-f}}\right) \tag{4.26}$$

$$Fz_{sn-r} = -2 \times k_{sn-r}\left(z_{v-r} - z_{usp-r}\right) - 2 \times c_{sn-r}\left(\dot{z_{v-r}} - \dot{z_{usp-r}}\right) \tag{4.27}$$

Where the front and rear displacements are given by, (4.28) and (4.29):

$$z_{v-f} = z_v - l_f \sin\theta_v \tag{4.28}$$

$$z_{v-r} = z_v + l_r \sin\theta_v \tag{4.29}$$

And the forces for the tyres (this is shown as $Fz_f$ and $Fz_r$ on the diagrams), (4.30) and (4.31):

$$Fz_f = Fz_{ty-f} = -2 \times k_{ty-f}\left(z_{usp-f} - z_{road-f}\right) \tag{4.30}$$

$$Fz_r = Fz_{ty-r} = -2 \times k_{ty-r}\left(z_{usp-r} - z_{road-r}\right) \tag{4.31}$$

The acceleration of the unsprung mass, (4.32) and (4.33):

$$\ddot{z_{usp-f}} = \frac{Fz_{ty-f} - Fz_{sn-f} - m_{usp-f}.g}{m_{usp-f}} \tag{4.32}$$

$$\ddot{z_{usp-r}} = \frac{Fz_{ty-r} - Fz_{sn-r} - m_{usp-r}.g}{m_{usp-r}} \tag{4.33}$$

The front and rear normal forces ($Fz_f$ and $Fz_r$) represent the force in the z axis, and they are used for calculating tyre friction, gradient forces and rolling resistance.

The difference between the normal force and the acceleration/gravity force, see section 4.4.2, is used to find the linear and rotational acceleration of the vehicle chassis, (4.34) and (4.35):

$$\ddot{z_v} = \frac{(Fz_{sn-f} - Fz_{a-f}) + (Fz_{sn-r} - Fz_{a-r})}{m_s} \tag{4.34}$$

$$\ddot{\theta_v} = \frac{(Fz_{sn-r} - Fz_{a-r}) \times l_r - (Fz_{sn-f} - Fz_{a-f}) \times l_f}{J_v} \tag{4.35}$$

This can then be integrated to find $\dot{z_v}$ and $z_v$, $\omega_v$ and $\theta_v$, which is used for calculating the normal force.

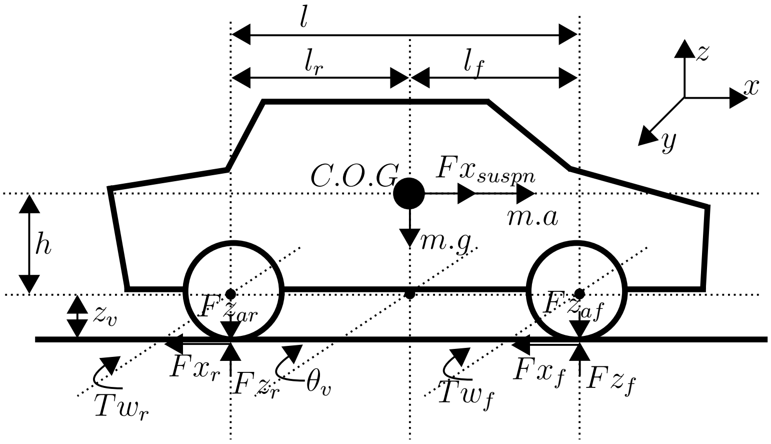

Figure 4.36: Complete vehicle forces and torques

### 4.4.2 Torque Reaction and Acceleration

The two effects acting on the suspension are the vehicle acceleration and the torque reaction of the torque at the wheels. The torque reaction force is caused by the resultant of the vehicle applying a torque against the road and the vehicle being able to rotate around the axis along the axle. These are used to calculate the forces $Fz_{a-f}$ and $Fz_{a-r}$ that are used in the equation above. The acceleration is used to calculate a change of the force in the z axis [82], the parameters used are labelled on the diagram in Fig. 4.36.

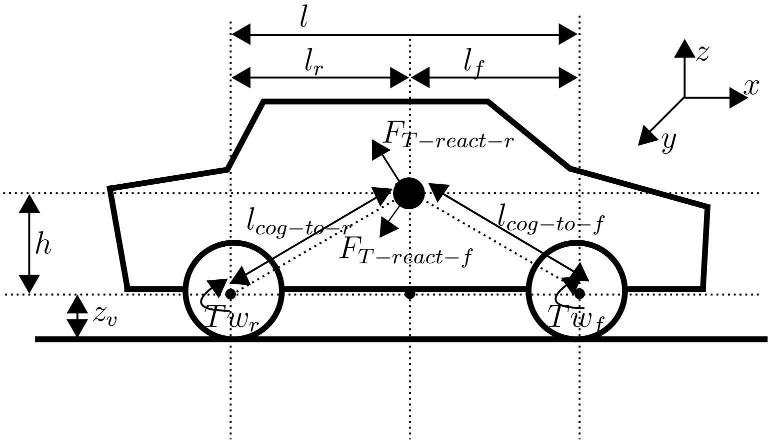

Figure 4.37: Torque reaction force on suspension

The torque is used to calculate a force acting on the centre of gravity, pivoting around the centre of the wheel, shown in Fig. 4.37. The reaction torque is equal is magnitude but opposite in direction to the wheel torque.

The distance between the axle and the vehicle's centre of gravity is calculated from (4.36) and (4.37):

$$l_{cog-to-f} = \sqrt{l_f^2 + h^2} \tag{4.36}$$

$$l_{cog-to-r} = \sqrt{l_r^2 + h^2} \tag{4.37}$$

This in turn determines the angle between the centre of gravity and the horizontal axis, (4.38) and (4.39):

$$\theta_f = \arctan\left(\frac{h}{l_f}\right) \tag{4.38}$$

$$\theta_r = \arctan\left(\frac{h}{l_r}\right) \tag{4.39}$$

The torque at each wheel can then be translated into a linear force, (4.40) and (4.41):

$$F_{T-react-f} = \frac{T_{w-f}}{l_{cog-to-f}} \tag{4.40}$$

$$F_{T-react-r} = \frac{T_{w-r}}{l_{cog-to-r}} \tag{4.41}$$

The forces can then be broken down into x and z axis components for both front and rear torques, (4.42), (4.43), (4.44) and (4.45):

$$F_{Treact-x-front} = F_{T-react-f} \times \sin\theta_f \tag{4.42}$$

$$F_{Treact-x-rear} = F_{T-react-f} \times \cos\theta_f \tag{4.43}$$

$$F_{Treact-z-front} = F_{T-react-r} \times \sin\theta_r \tag{4.44}$$

$$F_{Treact-z-rear} = F_{T-react-r} \times -\cos\theta_r \tag{4.45}$$

These are then combined into x and z forces, (4.46) and (4.47):

$$F_{Treact-x} = F_{Treact-x-front} + F_{Treact-x-rear} \tag{4.46}$$

$$F_{Treact-z} = F_{Treact-z-front} + F_{Treact-z-rear} \tag{4.47}$$

Finally $Fz_{a-f}$ and $Fz_{a-r}$ can be found, (4.48) and (4.49):

$$Fz_{a-f} = \frac{m_s.g.\left(l_r - h\tan\left(-\theta_v\right)\right).l_r + F_{Treact-z}l_r}{l} - \frac{\left(m.a + F_{Treact-x}\right).h}{l} - F_{Treact-z-front} \tag{4.48}$$

$$Fz_{a-r} = \frac{m_s.g.\left(l_f - h\tan\left(-\theta_v\right)\right).l_f + F_{Treact-z}l_f}{l} - \frac{\left(m.a + F_{Treact-x}\right).h}{l} - F_{Treact-z-rear} \tag{4.49}$$

The front and rear z axis torque reactions are subtracted at the end of each equation above, as the wheels are used as a pivot point but are not actually fixed, otherwise the torque reaction force would cause the vehicle to appear to weigh less, which is not correct.

### 4.4.3 Changing Centre of Gravity and Resultant Longitudinal Force

The longitudinal suspension force ($Fx_{suspn}$) is calculated from the rotational acceleration at the point below the centre of gravity. By multiplying by the height of the centre of gravity, this will give the linear acceleration speed (of the centre of gravity) at this point. Then by simply multiplying by the vehicle mass (assuming it is concentrated at this point), gives the force at this point, which is then applied to accelerate or decelerate the vehicle, (4.50):

$$Fx_{suspn} = \ddot{\theta}_v \times h \times m_s \tag{4.50}$$

## 4.5 Tyre Slip

The tyres are not always in perfect contact with the road and cannot always transmit the full linear wheel force to the road from the vehicle. This allows the tyres to slip and turn at a different speed than the road they are in contact with, depending upon whether the vehicle is accelerating or braking. The tyre slip model works in the longitudinal driving direction only.

If the weight distribution of the vehicle is uneven, for example if the centre of gravity is closer to the rear of the vehicle due to the battery location or a heavy load, this will provide good traction during forward acceleration, as the acceleration will cause the rear wheel normal force to increase. But during regenerative braking it is easy to lock the rear wheels as the weight is transferred to the front wheels, reducing the traction available at the rear.

Tyre slip usually follows the plot in graph in Fig. 4.38, with the tyre slip being proportional to the friction coefficient of the surface (and also the normal force). A slip of 1 is when the wheels are locked and the vehicle is still moving. Slip can be greater than 1 if the wheels are driven in reverse whilst the vehicle is moving forwards, or vice versa.

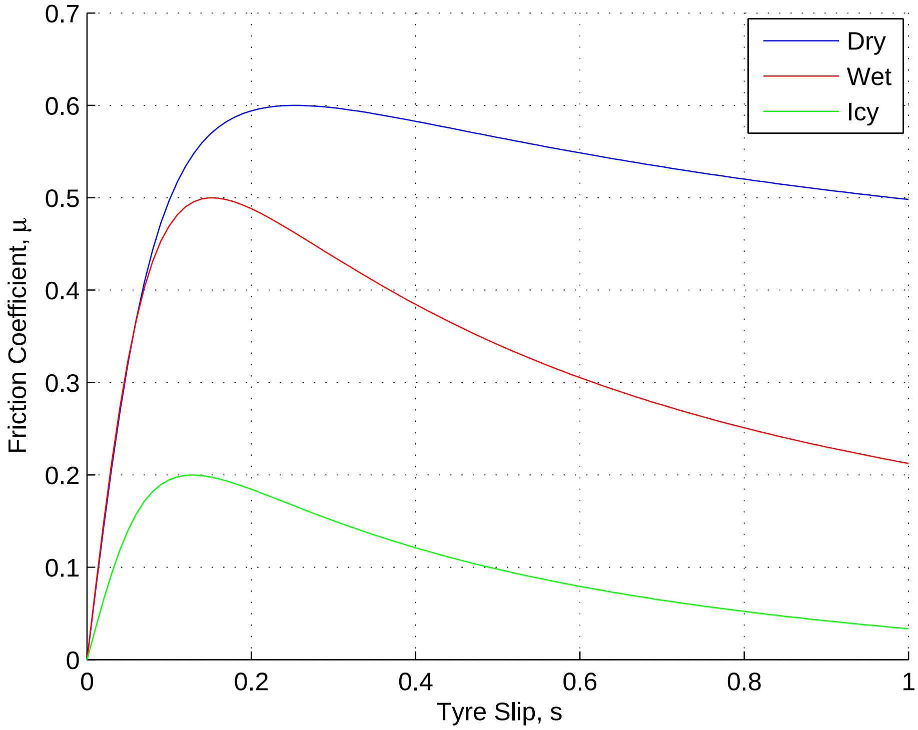

Figure 4.38: Tyre slip coefficient graph

Tyre slip is calculated by (4.51):

$$s = \frac{(r_w \times \omega_w) - v_v}{max\left(|r_w \times \omega_w|, |v_v|\right)} \tag{4.51}$$

Where $r_w \times \omega_w = v_v + v_w$ = vehicle velocity + $\omega_{slip}$ wheel velocity difference (tyre slip speed) from vehicle velocity. The traces in graph in Fig. 4.38 are generated using the following standard tyre slip [82, 84, 85] equation:

$$\mu = D \sin\left[C \arctan\left(B.s - E\left(B.s - \arctan\left(B.s\right)\right)\right)\right] \tag{4.52}$$

B = stiffness factor - this is multiplied by the amount of slip, so determines how much effect it has, will usually increase with increasing vehicle speed
C = shape factor  this parameter can be used to cause the friction coefficient $\mu$ to ramp off faster, it usually is increased with increasing vehicle speed
D = peak value  this is usually equal to $\mu_{max}$, but will usually be reduced with increasing vehicle speed
E = curvature factor - determines the shape around the peak

As the speed of the test vehicle is quite low, the effect of speed on the slip curves is not considered.

The values used to generate the traces in this section are as follows and are similar to values found in this paper [82], but modified to be approximate for the test vehicle and test driving surface and the paper only gives values for dry conditions. In section 5.2.7 $\mu_{max}$ (D) is determined for dry conditions experimentally using the test vehicle, the values for wet and icy conditions are chosen to give an appropriate reduction in maximum tyre friction coefficient. The only other parameters changed from [82] is C as it is increased for the wet and icy conditions so that the friction coefficient reduces faster than the dry conditions.

| **Symbol** | **Dry** | **Wet** | **Icy** |
|---|---|---|---|
| $\mu_{max}$ | 0.6 | 0.5 | 0.2 |
| **B** | 7.527 | 7.527 | 7.527 |
| **C** | 1.6 | 2.0 | 2.2 |
| **D** ($\mu_{max}$) | 0.6 | 0.5 | 0.2 |
| **E** | 0.5 | 0.5 | 0.5 |

Table 4.1: Tyre slip parameters

The driving force in the x axis can then be calculated by:

$$Fx = \mu \times Fz \tag{4.53}$$

$Fx$ here is equivalent to $F_d$ (drive) in the vehicle model.

Simulating the tyre slip in this way causes a number of problems. This is due to the fact that the tyre slip is dependant upon the difference in speed between the tyre and the vehicle, which is determined by the friction coefficient, which is calculated from the slip. The direct relationship prevents Simulink from calculating the response easily without long simulation times.

An approximation has been used (see graph in Fig. 4.39) which does not allow $\mu$ to change with slip until it has exceeded the point where the peak friction coefficient $\mu_{max}$ is reached. After this $\mu$ is calculated as normal, allowing the wheels to lock.

Including the changing tyre slip coefficient still adds a great deal of complexity to the simulation, increasing the calculation time required by many times due to the direct relationship involved. It is often not simulated, as during normal driving conditions it should not have any effect if the above approximation is used.

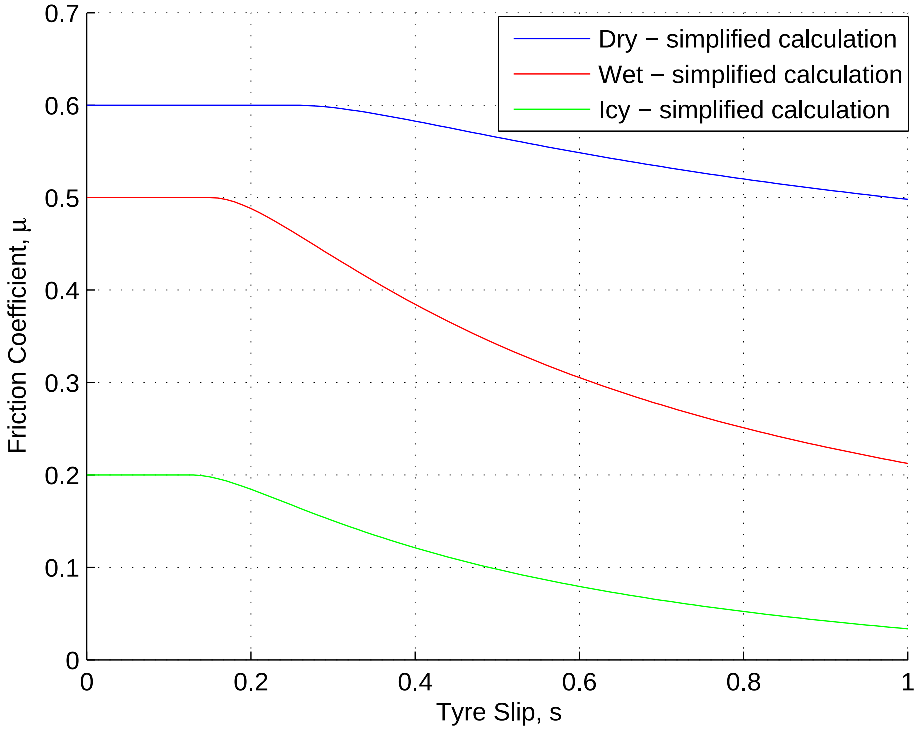

Figure 4.39: Approximated tyre slip coefficient graph

## 4.6 Summary

This chapter has described in sections 4.1 to 4.5 how the model of the vehicle is derived and gives equations for each section. Although there are some approximations, this model should include all areas that have an impact on the results of this thesis so the effect of the approximations should be negligible.

In the next chapter 5 the parameters for the test vehicle are determined and then the model derived in this chapter is simplified so that it can be run on a micro-controller. In section 5.2 specific tests are carried out to determine each of the parameters required for the simulation work and in section 5.3 the model is simplified to be used for the estimation and control work in chapters 6 and 7.

# Chapter 5

# Experimental Setup and Simplified Model

## 5.1 Test Vehicle and Experimental Setup

### 5.1.1 Test Vehicle

The vehicle used for the majority of the experimental work is a commercially available golf buggy, shown in Fig. 5.1.

Figure 5.1: Photo of the test vehicle

This vehicle was selected due to the limited number of vehicles being available, its speed range being appropriate for driving indoors, its ability to easily spin its wheels when accelerating and lock them when regenerative braking, and it being a worst case example in terms of vehicle dynamics - although comparable to other vehicles (including some on-road vehicles). The gearbox within the vehicle has a large amount of slack between the cogs, with the axle having low damping and the suspension springs have a low stiffness coefficient, leading to its effect on the rest of the vehicle being significant. The vehicle has a single traction motor driving the rear wheels through a fixed ratio differential. A block diagram of the vehicle is shown in Fig. 5.2.

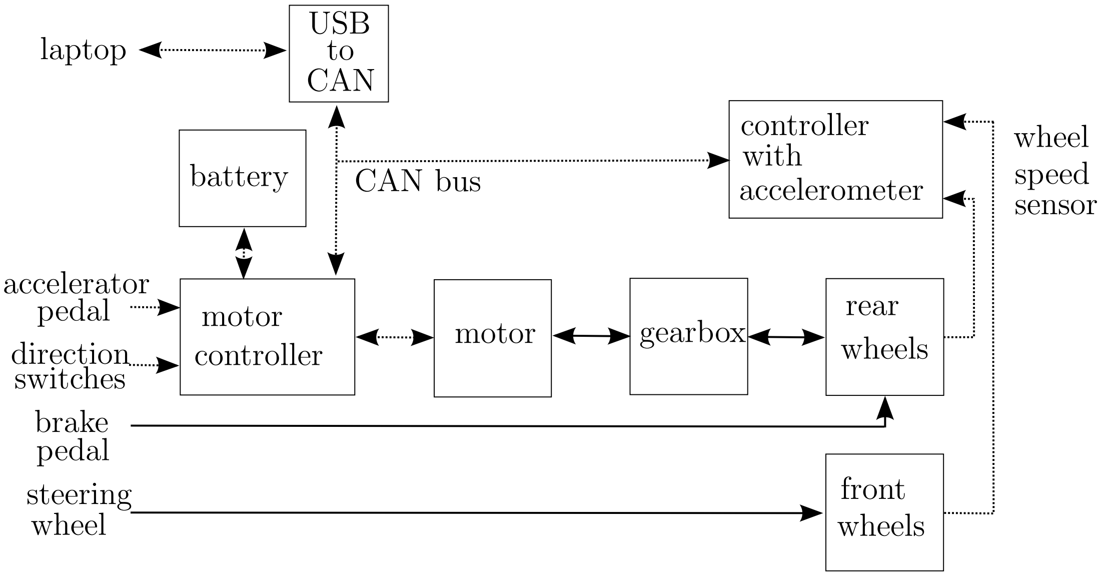

Figure 5.2: Block diagram of the test vehicle

The basic parameters of the vehicle are listed in table 5.1 below, these are all easily measurable without having to perform any additional tests:

| Symbol Name | Value | Units | |
|---|---|---|---|
| $m_{chassis}$ | Unloaded chassis mass | 224.5 | $kg$ |
| $m_{batt}$ | Battery mass | 186 | $kg$ |
| $m_v$ | Unloaded vehicle mass | 482.5 | $kg$ |
| $n$ | Gear ratio | 12.28 | |
| $r_w$ | Tyre radius | 0.21 | $m$ |
| $V_{batt}$ | Nominal battery voltage | 48 | $V$ |
| $v_{v-pk}$ | Top speed | $\sim 7$ | $ms^{-1}$ |
| $F_{x-pk}$ | Peak acceleration force | $\sim 3000$ | $N$ |
| $T_{pk}$ | Peak motor torque | 50 | $Nm$ |
| $\omega_{pk}$ | Peak motor speed | 5000 | $rpm$ |
| $P_{pk}$ | Peak motor power | $\sim 8$ | $kW$ |
| $np$ | Motor pole pairs | 4 | |

Table 5.1: Basic vehicle parameters

### 5.1.2 Motor Controller

The motor controller is a 48V Sevcon Gen4 450A (size 4) AC controller, as shown in Fig. 5.3. It is rated for 540A for 10 seconds, 450A for 2 minutes and 180A continuously. It contains both the power components (MOSFET three phase bridge) and logic including the digital signal processor (DSP) and inputs/outputs to connect to the throttle, direction switches and other inputs along with outputs to power a battery isolation contactor and electromagnetic brakes. The software has been modified to carry out the experimental tests within this thesis; replacing the control algorithms that operate between the accelerator pedal demand and the torque demand input to the lower level motor control. It uses vector control to control the motor, switching at $8kHz$ and the current control loops also run at this frequency. The processor used is a Texas Instruments TMS320F2811 32bit fixed point processor. It has Controller Area Network (CAN) connectivity for updating the software and logging experimental data, using a USB to CAN adapter with a laptop. Data for logging is transmitted at a $20ms$ (millisecond) rate. For faster sample rates down to $500\mu s$ (microsecond) it is possible to buffer the data in RAM and then transmit the stored data, 40 samples, every $20ms$ along with a timestamp for each sample from the CPU clock. For very fast samples rates down to $125\mu s$ all the data has to be saved to RAM and then downloaded after the test, although this limits the test time to $50ms$ if recording two 16bit values.

Figure 5.3: Photo of the motor controller

A block diagram is shown in Fig. 5.4. How the experimental data was recorded over CAN from all the sensors on the vehicle is shown.

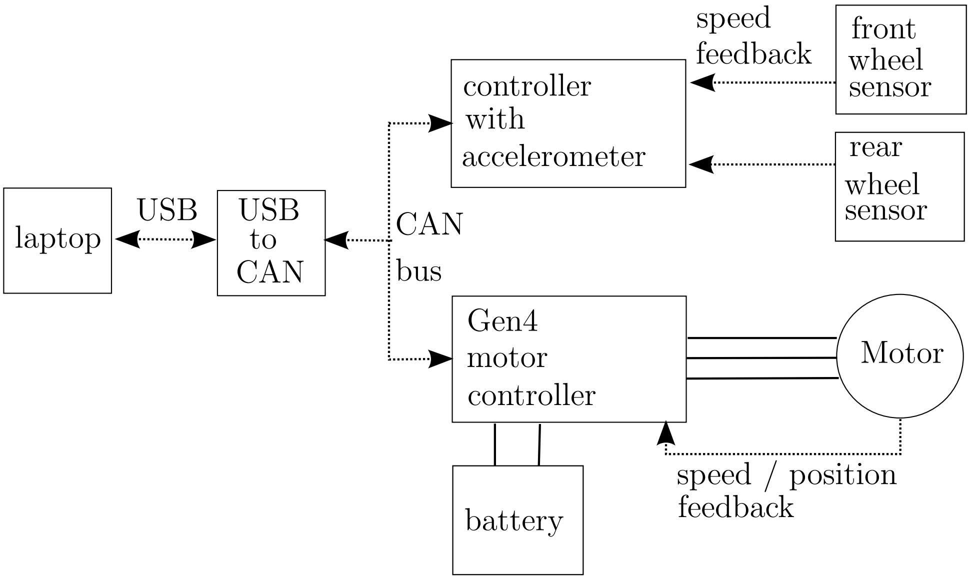

Figure 5.4: Block diagram of controller setup

### 5.1.3 Motor

The motor installed on the vehicle is a PMAC axial flux machine shown in Fig. 5.5 made by Mars Electric (now Motenergy). It is fitted with 2 encoders: a hall effect UVW encoder and a 64 ppr (pulses per revolution) quadrature incremental AB encoder. The UVW encoder gives 24 edges per mechanical revolution (as the motor has 4 pole pairs) and the AB gives 256 edges per revolution, although no absolute position information. Normally the motor is only fitted with the hall effect UVW encoder giving the absolute position information at a low resolution - it is accurate to 60 electrical degrees. The higher precision incremental encoder was fitted for more accurate measurement of motor position and speed for the results in this thesis, but is not required for any of the control or estimation schemes presented.

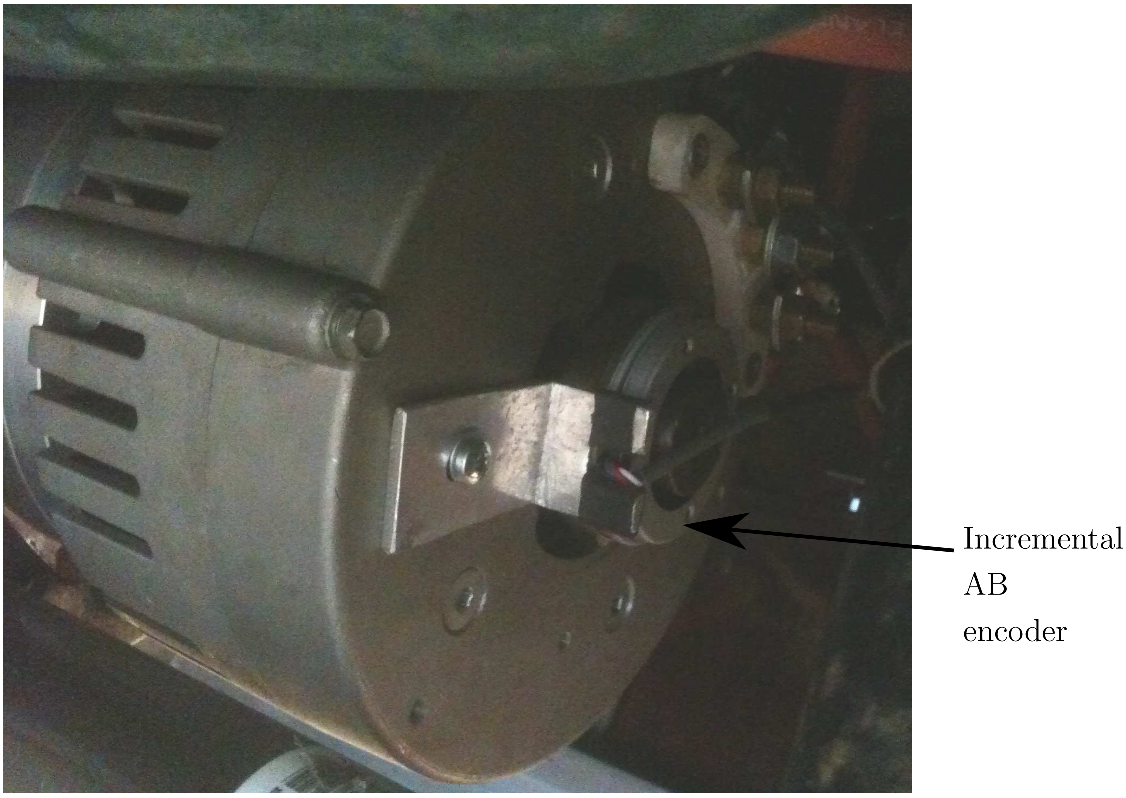

Figure 5.5: Photo of the motor attached to the test vehicle differential gearbox

In order to ensure the motor torque output is accurate, the back-EMF constant has been measured by spinning the motor with an oscilloscope connected across two of the terminals, shown in graph 5.6. By putting this waveform through FFT it is possible to find the magnitude of the fundamental, shown in Fig. 5.7. Dividing this by the electrical frequency, $332.44 rad\, s^{-1}$, for the example shown in Fig. 5.6, gives a $k_e$ of $0.0244 V_{line-rms}/rad\, s^{-1}$

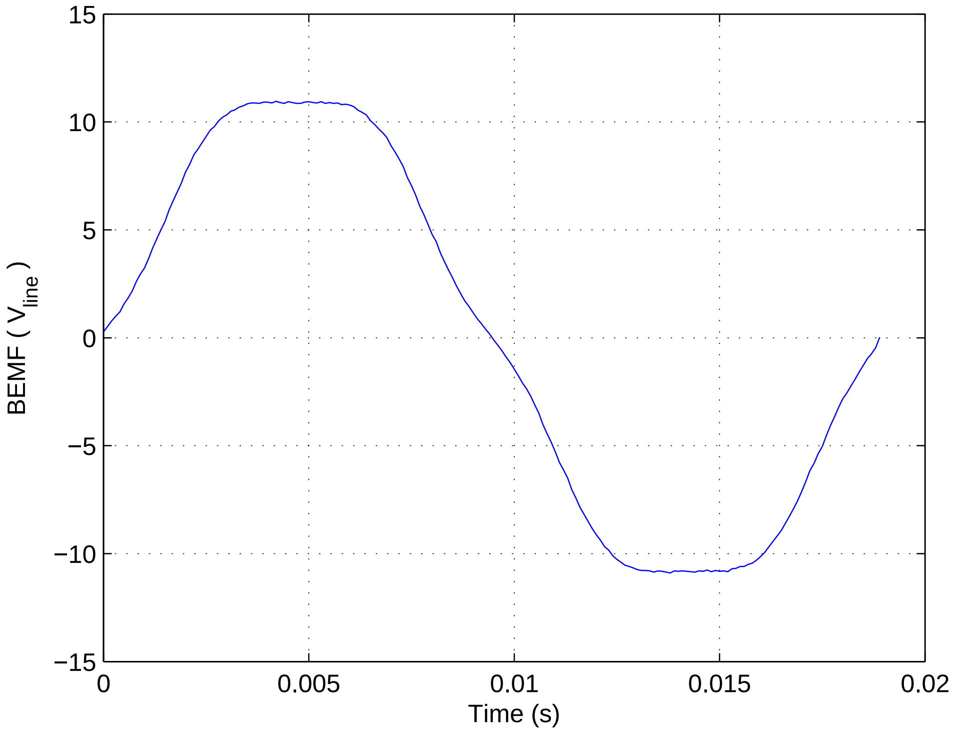

Figure 5.6: Motor line-line back-EMF waveform as measured at $332 rads^{-1}$ electrical frequency

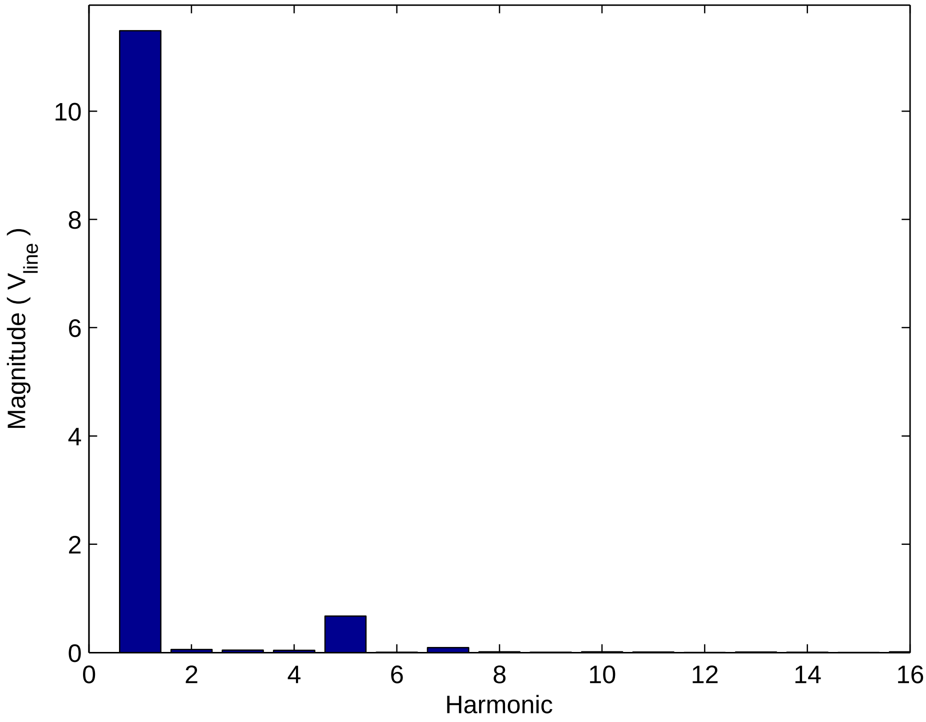

Figure 5.7: Motor back-EMF waveform harmonics at $332rads^{-1}$ electrical frequency

The back-EMF constant $k_e$ is used to calculate the ratio between torque demand and quadrature axis current demand. The motor has then been characterised to ensure the actual torque is close to the measured, as there is no torque measurement on the vehicle. Prior to installing the motor on the vehicle it has been tested with a calibrated torque transducer on a dynamometer to ensure its accuracy. Before the motor enters saturation, the torque output is linearly proportional to the quadrature axis current, shown in the graph in Fig. 5.8. At higher speeds the peak torque out of the motor drops off, as shown in Fig. 5.9, due to the voltage limitation of the battery. For this work the torque demand is assumed to be equal to the actual torque.

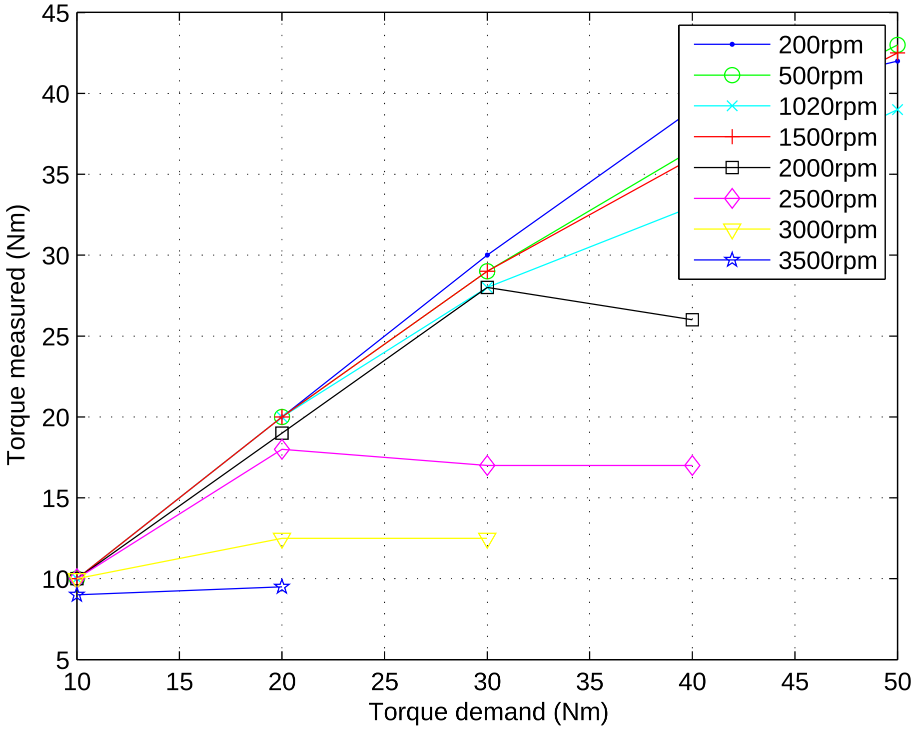

Figure 5.8: Graph showing the motor's measured torque varying with speed and torque demand

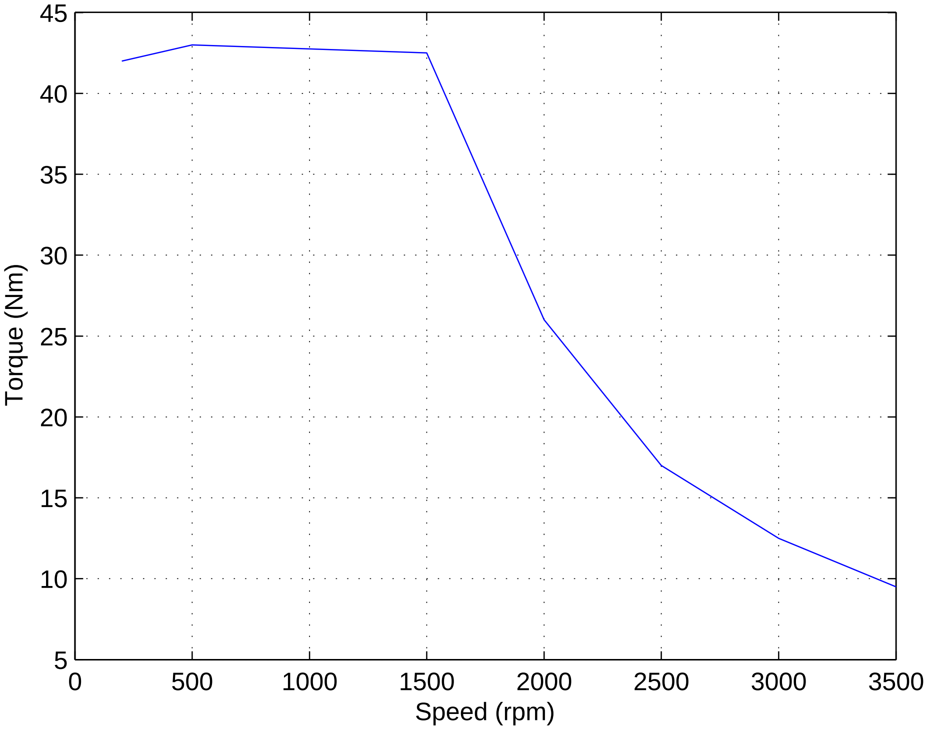

Figure 5.9: Graph showing the motor's peak torque output varying with speed

### 5.1.4 Wheel Speed Sensors

The vehicle was fitted with wheel speed sensors on the front left and rear right wheels, the front wheel sensor is shown in Fig. 5.10. The sensors were fitted to carry out the tests in this thesis and it is not standard to have them fitted to this vehicle. The sensors use Neodymium magnets ($5mm$ diameter) placed with alternate north-south poles and a latching hall effect sensor, mounted very close to the magnets to produce an alternating pulse signal that changes in frequency with the speed of wheel rotation. A separate Sevcon Gen4 motor controller is used to process the signals from these two sensors and then transmit the speed signals over the CAN network. The front left wheel sensor gives speed feedback from one of the non-driving wheels and the rear one gives speed from the driven wheels, subtracting the motor speed will give the speed of the unmeasured rear left wheel. As the wheel internal circumference is not an exact multiple of the magnet diameter, there is a gap of a couple of $mm$ once every rotation. This can be seen on some of the traces as a single sample spike in the speed measured.

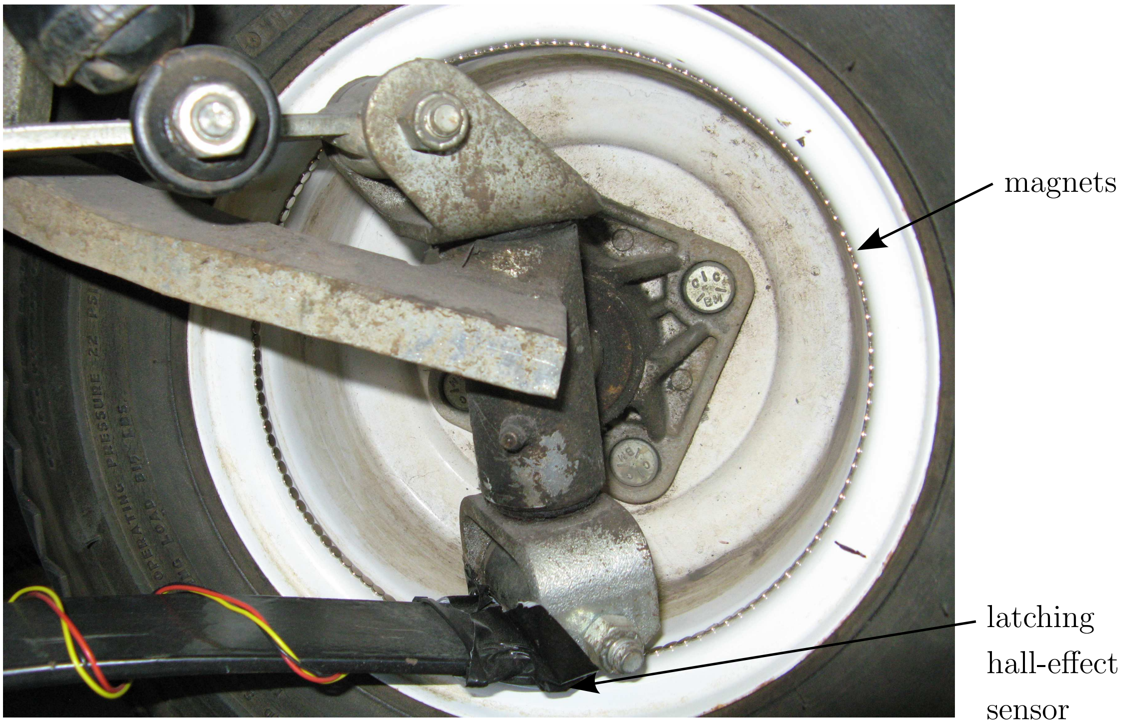

Figure 5.10: Photo showing the wheel speed sensors fitted to the test vehicle front left wheel

### 5.1.5 Accelerometers

An accelerometer was fitted to the vehicle in order to measure vehicle acceleration and determine the smoothness of its response. An Analogue Devices ADXL345 3-axis accelerometer was used but only the x-axis (vehicle driving direction) was measured for this work. A sample rate of $100Hz$ was chosen as the sensor noise increases greatly above this and this is sufficiently faster than the vehicle acceleration oscillations that are seen.

### 5.1.6 Vehicle Driving Area

All vehicle testing was done indoors at Sevcon, shown in Fig. 5.11. The driving area is $35m$ by $5m$ in size which limits the duration and distance of the vehicle tests. As the test vehicle has a relatively low maximum speed this is less of an issue compared to an on-road vehicle.

Figure 5.11: Photo of the vehicle test area

### 5.1.7 Summary

Only one vehicle was used for the experimental work. Although the golf buggy is only one vehicle type, the vehicle drivetrain layout which has a single motor, fixed gear ratio differential driving only the rear axle is quite common amongst small electric vehicles used both on-road and off-road (including industrial use). The vehicle used has a larger backlash gearbox, lower damped drivetrain and very soft suspension. Unfortunately there is a limit to the amount of load mass that can be added to this vehicle and the work within the thesis has not been applied to vehicles that have trailers such as airport tow tractors. As the testing is done indoors the top vehicle speed possible is limited by the space available, although speeds near to the maximum speed of the vehicle used were reached, this speed is very low compared to that of an on-road vehicle. Factors such as the decrease in tyre to road friction with increasing speed cannot be investigated at the driving speeds possible.

## 5.2 Determining the Test Vehicle Parameters

### 5.2.1 Introduction

There are a greater number of model parameters defined in the preceding sections 4.1 to 4.5 than those listed in table 5.1. These are more difficult to obtain and vary from simply applying a torque and measuring the axle flex, to using a genetic algorithm process on vehicle experimental data to find the parameters.

### 5.2.2 Locked Wheel Tests

If the vehicle's rear brakes are applied whilst stationary, it is possible to isolate the drivetrain from all other vehicle dynamics, shown in Fig. 5.12. This makes it easy to identify just the parameters within this part of the vehicle such as: motor inertia $J_m$, motor frictional losses $b_m$, motor stiction losses $T_{stic-m}$, gear ratio $n$, backlash size $\alpha_{bl}$, axle/gear stiffness $k_g$ and axle/gear damping $c_g$.

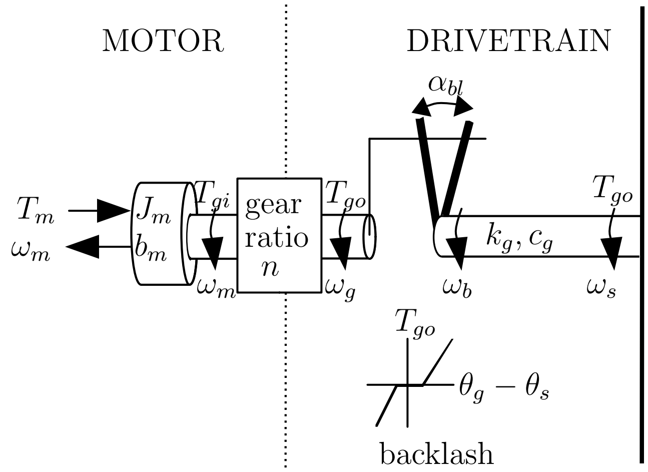

Figure 5.12: Rotational diagram of the drivetrain with brakes applied

The first test is to determine the axle stiffness $k_g$, as the gear ratio $n$ is already known from the vehicle manufacturer's data. This will include the flexibility in the motor's mounting as the vehicle has a 'live' axle, the motor is not attached to the sprung mass of the vehicle, shown in Fig. 5.13. Applying a torque causes a flex in both the axle and the mount as equal torques are applied to both, but this is acceptable as both effects have been combined into a single parameter. By first applying a small torque to engage the gear cogs, then steadily increasing the torque at a fixed ramp (up to a maximum of $50Nm$) and measuring the change in encoder angle, dividing the torque change by angle change will give the stiffness $k_g$.

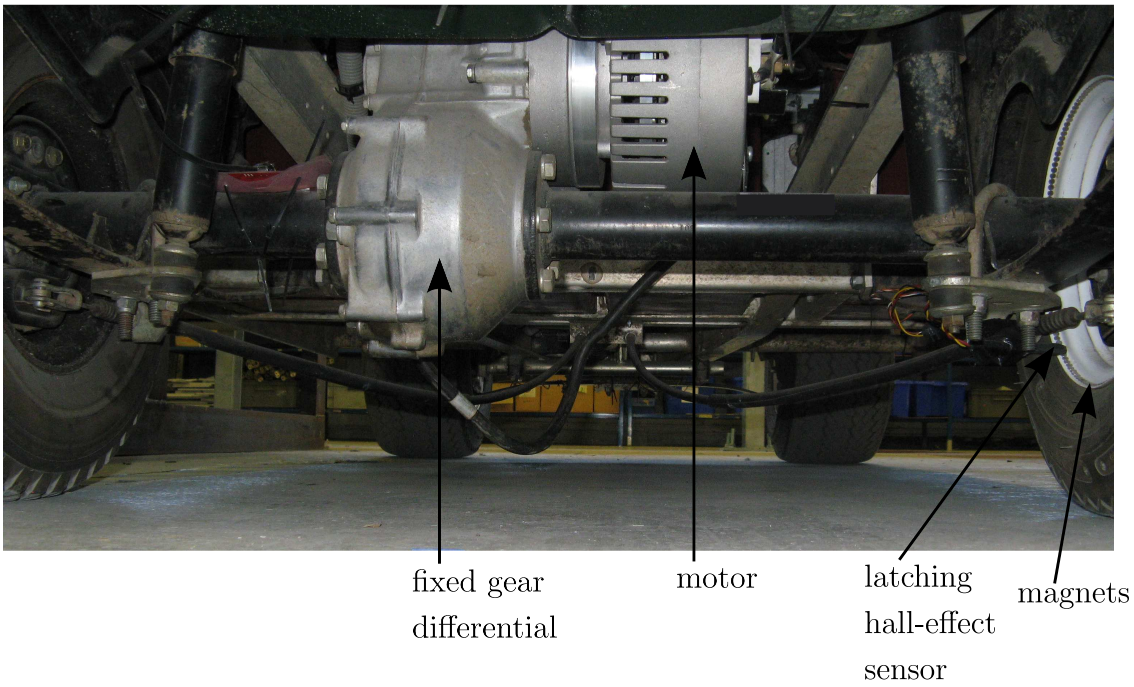

Figure 5.13: Photo of the test vehicle's drivetrain

In Fig. 5.14 the results of 8 tests are shown, with the flexibility calculated at multiple torque levels. It can be seen that the $k_g$ term increases largely at higher torques, but this as expected from the results previous section 5.1.3, as the motor is saturated and not producing the demand torque levels. Before saturation the axle stiffness is almost constant at increasing torque levels, the best fit line drawn in green in Fig. 5.14, although the points in blue do seem to be increasing slightly with torque.

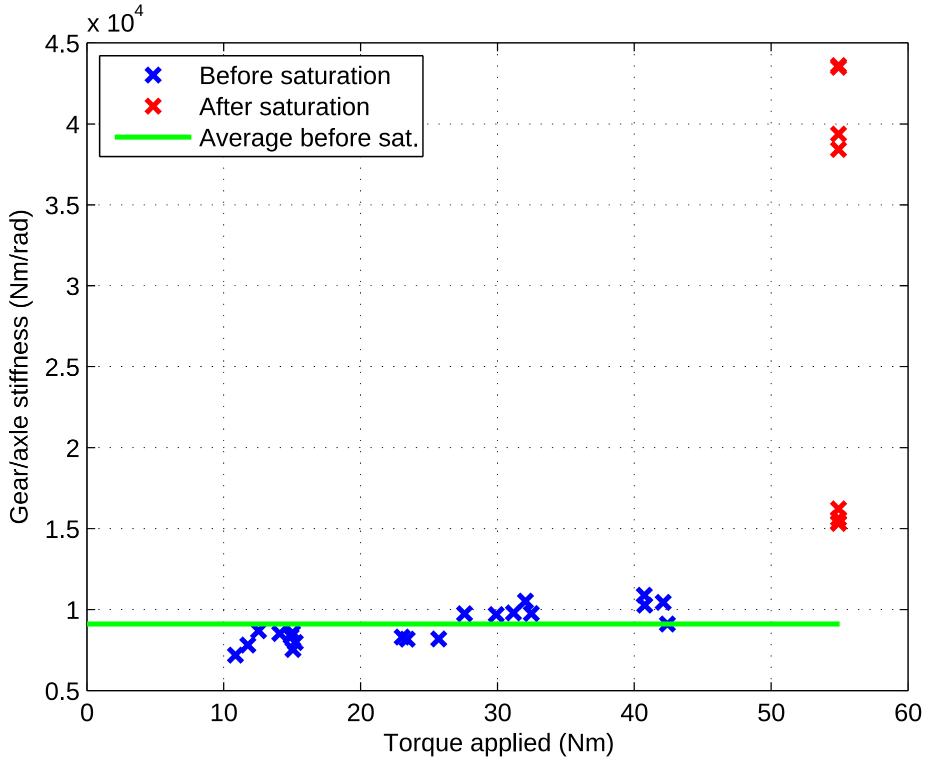

Figure 5.14: Results of the axle stiffness tests

The average axle/gear stiffness $k_g$, ignoring the saturated motor region, is $9100 Nm\, rad^{-1}$.

A further test to include the flexibility of the wheels and mainly the tyres, involves stopping the vehicle from moving (using a sturdy piece of wood between the vehicle and a wall) and, similar to the first test, applying a torque and measuring the total drivetrain flex (including the wheels), shown in Fig. 5.15.

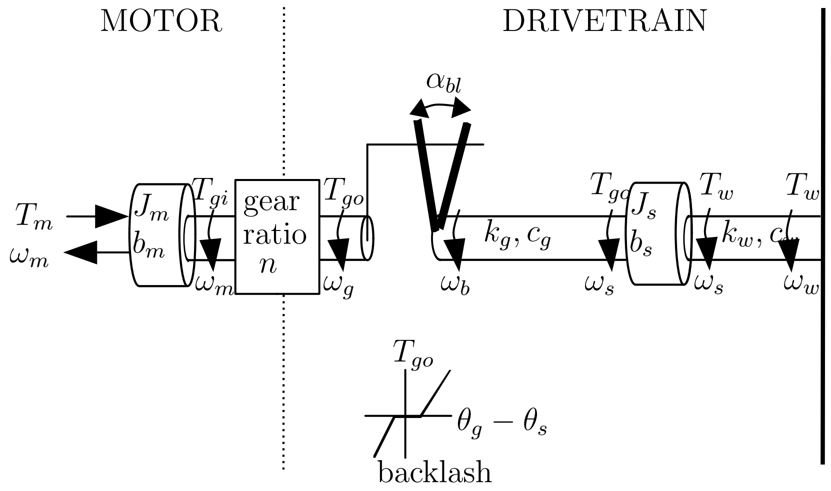

Figure 5.15: Rotational diagram of the drivetrain with the vehicle unable to move

This gave a combined drivetrain stiffness $k_{total}$ of $7500 Nm\, rad^{-1}$.

The relationship combining the two is (5.1), which can be rearranged to give (5.2).

$$k_{total} = \left( \frac{1}{k_g} + \frac{1}{k_w} \right)^{-1} \tag{5.1}$$

$$k_w = \left( \frac{1}{k_t otal} - \frac{1}{k_g} \right)^{-1} \tag{5.2}$$

$$k_w = \left( \frac{1}{7500} - \frac{1}{9100} \right)^{-1} = 40,000 \tag{5.3}$$

A wheel rotational stiffness $k_w$ of around $40,000 Nm\, rad^{-1}$ is then calculated. As this is significantly higher than the remainder of the drivetrain it has been ignored for a large number of tests, allowing a simpler two mass drivetrain model to be used.

### 5.2.3 Tuning of Drivetrain Parameters using Genetic Algorithms (GA)

In order to obtain the remaining drivetrain parameters between the motor and the wheels, a further test is performed with the wheels locked (using the mechanical brakes) by applying a fixed torque and suddenly removing the torque rapidly and recording the position and speed of the resulting oscillation, as shown in Fig. 5.16.

In this example $37.5Nm$ is applied before being released. This torque waveform was chosen instead of stepping from zero torque to $37.5Nm$, as the backlash position will be known before that start of the test whereas if no torque was being applied the start backlash position will be unknown. Also stepping to zero torque can be done by simply disabling the power frame, whereas stepping to a fixed torque level during a large oscillation will require accurate current control to keep the torque constant.

The position information is recorded directly from the edge counts of the incremental AB encoder, the speed is calculated by timing between edges and the the torque is simply the torque demand, which scales directly to a quadrature axis current. As the data is sampled every $500\mu s$, it is buffered into RAM before transmitting on the CAN bus every $20ms$ to a laptop for saving the data.

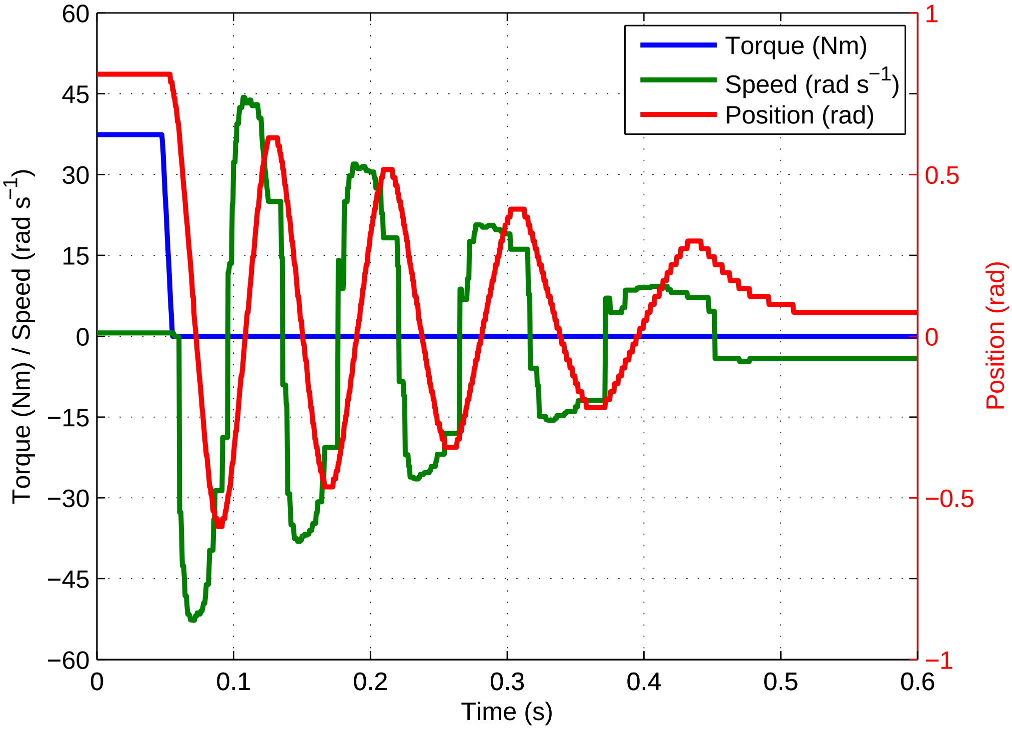

Figure 5.16: Results of the torque removal test

This is done at a number of torque levels, across the full range possible (25%, 50%, 75% and 100%), to help reduce the effect of any non-linearities. The parameters found should give the best overall result across the full torque range. By creating a simulation of the drivetrain it is possible to compare the performance of potential values for the parameters against the measured data.

A cost function (5.4) is used to quantify the accuracy of the parameters using the position and speed feedback, where $n$ indicates the test number out of the 8 performed, $t$ is the time of the measured data and $J_{avg}$ is the score or rating of the parameters accuracy.

$$J_{avg} = \frac{1}{2}\left(\frac{1}{8}\sum_{n=1}^{8}\int_{0}^{t}|\theta_m^n - \hat{\theta_m^n}|dt + \frac{1}{8\times 50}\sum_{n=1}^{8}\int_{0}^{t}|\omega_m^n - \hat{\omega_m^n}|dt\right) \tag{5.4}$$

The cost function for each set of conditions is calculated in the box labelled determine IAE (as it is made up of the integral of the absolute of the error) shown in Fig. 5.17, these are then combined to give $J_{avg}$. This allows the performance of a given set of parameters to be easily quantified in a single number, over a range of conditions.

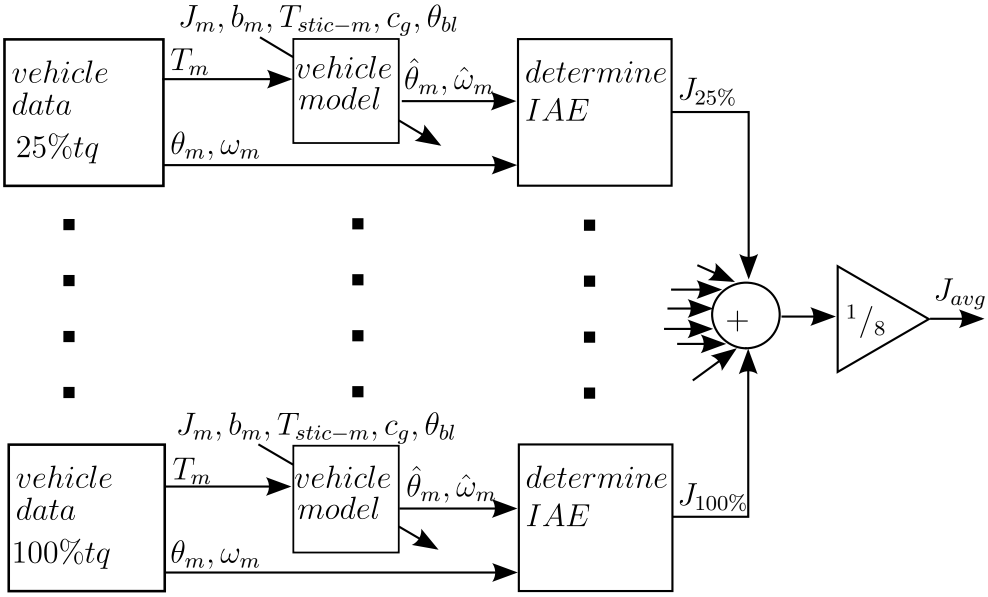

Figure 5.17: Cost function model to evaluate the accuracy of the vehicle parameters

In order to optimise the cost function, to find the most accurate drivetrain parameters, Genetic Algorithm (GA) is used.

The basic operation of the algorithm [36] follows these steps:

- Initialise the population of 50 “chromosomes” with random possible solutions for this parameters vector. These are initialised within a chosen predetermined range, that are the best guess for each parameter. As there are 5 parameters to be found: $J_m$, $b_m$, $T_{stic-m}$, $\alpha_{bl}$ and $c_g$, each chromosome contains 5 “genes”.
- Motor torque $T_m$, motor position $\theta_m$ and motor speed $\omega_m$ recorded from vehicle experiments are used to run the model off line with each of the possible sets of parameter values. This then generates a performance indication of the estimator using a cost function - see (5.4).

- Selection is then carried out based upon the fitness result in the step above. The highest performing (lowest cost function value) chromosomes are selected to for a pool of suitable candidates suitable for creating the next generation.
- In order to create the next generation, a crossover with a probability ratio of 0.6, is performed on the pool of suitable parents above. This takes two of the chromosomes and swaps over some of the genes randomly to try to create a better new pair of chromosomes.
- Non-linear systems can have local minima that are not the most optimal solution. Mutation is therefore performed, with a probability ratio of 0.35, to ensure that the optimisation finds the overall most suitable values. It works by randomly changing a bit in one of the new chromosomes to change its value.
- The process is then repeated with the new set of possible parameter values until either the maximum number of iterations is reached (100 iterations), or the desired performance is attained

The accuracy of the parameters used in the drivetrain model, given by the scores ($J_{avg}$) in Fig. 5.18, is seen to improve at each step. As it gets towards 100 generations, the improvement slows and the best-fit parameters are found, with a score of 0.649.

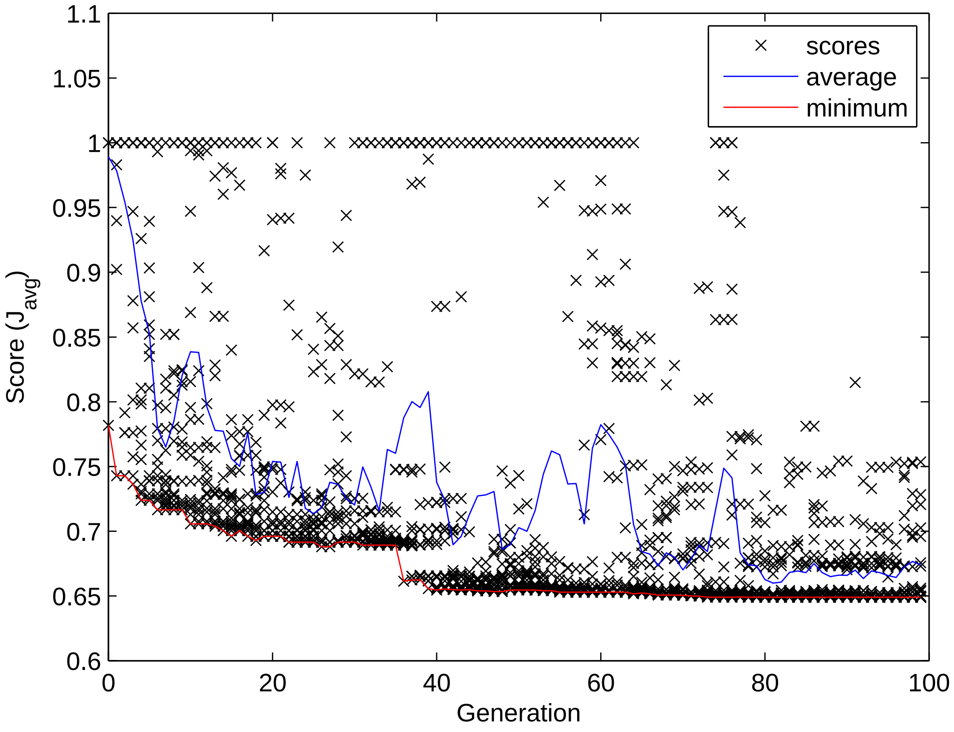

Figure 5.18: Graph showing the improvement in drivetrain parameters with each Genetic Algorithm iteration

The estimated parameters that give the score of 0.649 are: motor inertia $J_m = 6.5 \times 10^{-3} kg\, m^2$, motor frictional losses $b_m = 1.74 \times 10^{-3} Nm\, sec/rad$, motor stiction losses $T_{stic-m} = 0.35Nm$, backlash size $\alpha_{bl} = 20rads$ and axle/gear damping $c_g = 9.6Nm/rads^{-1}$.

### 5.2.4 Wheels in Air Tests

By raising the driving wheels and applying torque from the motor, the total drivetrain inertia, friction and stiction can be found; subtracting the known value for motor and estimated value for the wheels leaves the axle values remaining.

Wheel inertia is estimated from its mass and dimensions:

$J_w = 0.5 \times m_w \times r^2 = 0.5 \times 7 \times 0.115^2 = 0.0463kgm^2$

Front axle, which is just made up of the wheels:

$J_s f = 0.0926kgm^2$

$b_s f = 0.015Nm\, rads^{-1}$

$T_{stic-sr} = 0Nm$

Rear axle, which includes the drive shafts and gearbox output cog:

$J_s r = 1.96kgm^2$

$b_s r = 0.32Nm\, rads^{-1}$

$T_{stic-sr} = 3Nm$

Wheel rotational damping, although this parameter is rarely used due to using a 2 mass model:

$c_w = 70Nm\, rads^{-1}$

### 5.2.5 Other Estimated Parameters

For the aerodynamic drag model in (4.16), all that is required is the surface area of the front of the vehicle and the drag coefficient. The drag coefficient was estimated to be 1; a flat surface moving against the air. As can be seen in the photo in Fig. 5.1; the vehicle screen is almost flat.

$C_{df} = C_{dr} = 1$

$A_v = 2m^2$

The weight distribution is required to give the steady state suspension set up. Scales were not available to measure the front and rear of the vehicle separately, so the weight distribution was estimated based the location of the batteries and driver, assuming the vehicle chassis has its mass split equally to the front and rear.

The wheelbase length of the vehicle is:

$l_{total} = 1.65m$

The mass of the drivetrain components are measured using scales, which then allow calculation of the unsprung mass at the front $m_{usp-f}$ and rear $m_{usp-r}$ of the vehicle:

$m_{axle} = 27kg$

$m_{motor} = 10kg$

$m_{wheel} = 7kg$

$m_{usp-f} = m_{wheel} \times 2 = 14kg$

$m_{usp-r} = m_{wheel} \times 2 + m_{axle} + m_{motor} = 51kg$

The sprung mass components are then individually listed with their height $h$ from the centre of the wheels and their distance $l$ from the centre of the front axle. They are used for calculation of the total vehicle mass $m_{total}$, location of the vehicle centre of gravity ($h_{steady-state}$ and $l_{f-total}$) and the inertia around the centre of gravity $J_{v-s}$ (used for suspension simulation), which are given in table 5.2:

$m_{v(chasis)} = 224.5kg - m_{usp-f} - m_{usp-r} = 159.5kg$

$l_v = \frac{l_{total}}{2} = 0.825m$

$h_v = 0.25m$

$m_b = 186kg$

$l_b = 1.1m$

$h_b = 0.3m$

$m_d = 72kg$

$l_d = 1.0m$

$h_d = 0.6m$

The load added to the vehicle for various tests:

$m_l = 0kg\, to\, 200kg$

$l_l = 0.825m$

$h_l = 0.33m$

A total (unloaded) vehicle mass $m_{total}$ and sprung mass of $m_s$ of:

$m_{total} = 482.5kg$

$m_s = 417.5kg$

### 5.2.6 Acceleration and Coast Down Tests

If the vehicle is driven to a speed and allowed to coast to a stop; applying no foot brake or regenerative braking action, the total vehicle losses can be found. By subtracting all other known losses: aerodynamic drag and transmission, the rolling resistance makes up the remainder of the losses and can be determined.

$C_{r0} = 0.0176$

$C_{r0-scaler} = 0.7$

$C_{r1} = 0.0015m^{-1}s^{1}$

$C_{r2} = 0.0042m^{-1}s^{2}$

### 5.2.7 Tyre Traction

By increasing the driving and regenerative braking torques until traction is lost (by comparing motor speed to wheel speed), the friction coefficient was found.

$\mu_{max} = 0.6$

### 5.2.8 Suspension Parameters

Now that the weight distribution of the vehicle is known, the suspension and tyre linear stiffness can be measured by raising the vehicle and measuring the difference in displacement of either the suspension or the tyre. The suspension displacement change is measured as the distance between the bottom of the vehicle chassis and the ground when the suspension is just fully extended and when the vehicle is compressing it normally. The tyre displacement change is found similarly but by just raising the vehicle so that the tyre is no longer compressed and measuring the change in tyre diameter. The suspension damping was estimated by dropping the vehicle from a jack, whilst measuring the resultant oscillation using accelerometers mounted at the front and rear of the vehicle.

$k_{sf} = 23,500N/m$

$k_{sr} = 20,000N/m$

$c_{sf} = 250N/ms^{-1}$

$c_{sr} = 250N/ms^{-1}$

$k_{tf} = 150,000N/m$

$k_{tr} = 150,000N/m$

### 5.2.9 Vehicle Parameter Table

The complete set of important vehicle parameters have been summarised in the table below 5.2:

| Symbol Name | Value | Units | |
|---|---|---|---|
| $n$ | Gear ratio | 12.28 | no units |
| $r$ | Wheel radius | 0.21 | $m$ |
| $k_g$ | Gear/axle stiffness | 9100 | $Nm/rad$ |
| $c_g$ | Gear/axle damping | 9.6 | $Nm/rad\,s^{-1}$ |
| $\alpha_{bl}$ | Backlash size (at motor) | 20 | $deg$ |
| $J_m$ | Motor inertia | 0.0065 | $kg\,m^2$ |
| $b_m$ | Motor friction | $1.74 \times 10^{-3}$ | $Nm/rad\,s^{-1}$ |
| $T_{stic-m}$ | Motor stiction | 0.35 | $Nm$ |
| $J_s$ | Shaft/wheel inertia | 1.96 | $kg\,m^2$ |
| $b_s$ | Shaft/wheel friction | 0.32 | $Nm/rad\,s^{-1}$ |
| $T_{stic-s}$ | Shaft/wheel stiction | 3 | $Nm$ |
| $m_{total}$ | Vehicle total mass | 482.5 | $kg$ |
| $m_s$ | Vehicle sprung mass | 417.5 | $kg$ |
| $b_v$ | Vehicle combined friction | 18.8 | $N/m\,s^{-1}$ |
| $C_0.g$ | Stiction coefficient | 0.29 | $N$ |
| $h_{steady-state}$ | C.O.G. height | 0.33 | $m$ |
| $l_{f-total}$ | C.O.G. from front axle | 1.02 | $m$ |
| $J_{v-s}$ | Vehicle sprung inertia | 46.2 | $kg\,m^2$ |

Table 5.2: Vehicle parameters

### 5.2.10 Simulation Model and Parameter Accuracy Comparison to Experimental Results

In order to validate the accuracy of the simulation model and the characterised parameters, the recorded torque waveform from experimental data was used as the input to the vehicle model simulation. The resultant motor and vehicle speed output was then compared to the experimental data, shown in Fig. 5.19. The simulated motor speed very closely matches the measured result. The magnitude and the oscillation frequency are almost the same, although during the deceleration the simulated trace slow down at a lower rate than the measured. For the vehicle speed comparison the measured trace has more oscillation, this could be caused by some of the tyre rotational stiffness and damping being wrong. An improvement could be achieved if the Genetic Algorithm approach was also applied to other parameters, rather than just the drivetrain. This could be done as a separate optimisation problem, after the isolated drivetrain ones were found. Suitable parameters for this method would be the tyre rotational components and rolling resistance coefficients.

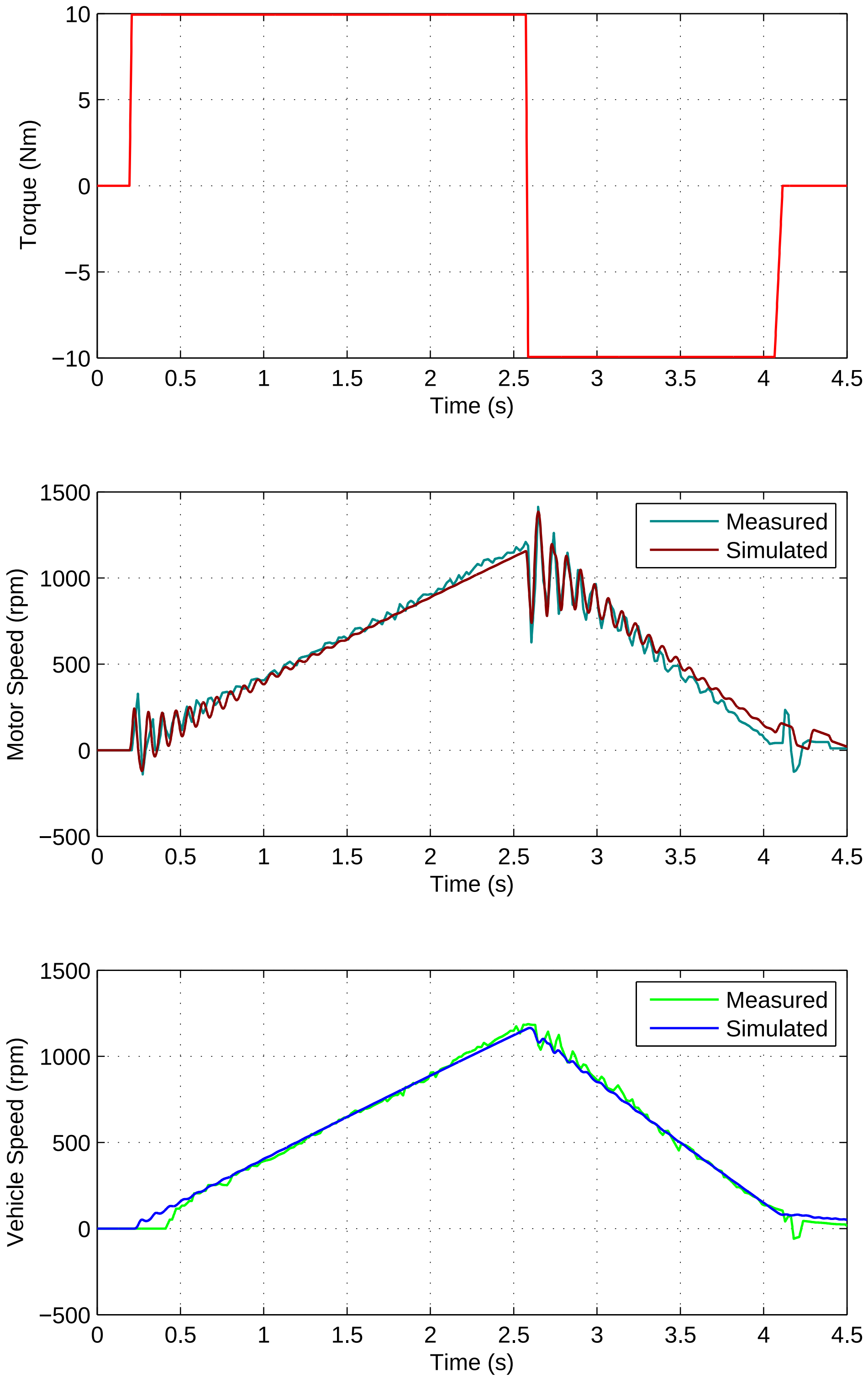

Figure 5.19: Graph comparing simulated vehicle response to experimental results

## 5.3 Simplified Vehicle Model

The full vehicle model derived in chapter 4 is used for simulation work, as it includes all the aspects of vehicle dynamics that have an impact on the results. It is common to use state space to represent complicated dynamical systems. It involves breaking down the model into a number of first order differential equations; the number of differential equations determines how many states are required and the complexity of the calculations. As matrices are used, the calculation required increases with the square of the number of states. It is desired to be able to run the model on the digital signal processor (DSP) so many assumptions are required to simplify it.

The first simplified model is in the next section 5.3.1. All suspension aspects have been ignored and cornering and gradients are only included as an external input added onto the tyre stiction force $F_{stic}$.

The best compromise model is in section 5.3.2, as it still retains the gearbox and backlash but removes the tyre dynamics, as these were found to not be important.

The simplest is in section 5.3.3, it can be used when there is little concern required for the gearbox dynamics.

### 5.3.1 Three-mass Model

The most advanced of the simplified models is given in Fig. 5.20. It focusses on the drivetrain dynamics, ignoring the suspension completely, which will lead to this model giving favourable acceleration response compared to the actual vehicle or the full vehicle model. As the vehicle has no feedback from the suspension states, trying to include any aspects of the suspension would be difficult. The suspension also greatly increases the number of states by many times. The rolling resistance tyre stiction is included as an input that can be calculated outside the model based on the graph shown in Fig. 5.20. It is possible to add gradient forces and cornering losses to this input to include them. This model is not used in any of the estimation or control work but is included as it shows the intermediate step before forming the two-mass model in the next section 5.3.2.

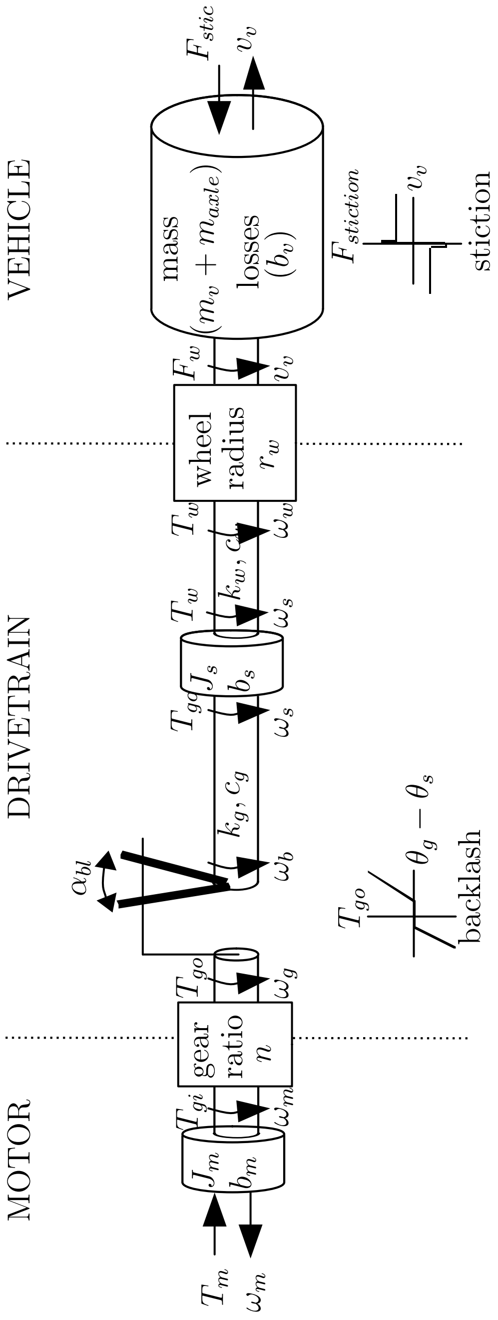

Figure 5.20: Rotating three mass model

The stiction force part of the rolling resistance is calculated outside of the state space model and fed in as an input ($F_{stic}$); the stiction is either zero, equal to the force being applied at the wheels ($F_w$) or one of two levels applied in either direction. A higher stiction level is applied whilst stationary ($F_{stic-max-stationary}$), but it is clamped to ensure that it does not exceed the force at the wheels; it is equal and opposite to the force at the wheels until it reaches the max level ($F_{stic-max-stationary}$). When the vehicle then starts to move it reduces to the vehicle moving level ($F_{stic-max-moving}$) and is applied to oppose the vehicle velocity direction.

The maximum stiction levels are calculated in (5.5) based upon the equations developed in 4.3.4:

$$\begin{aligned} F_{stic-max-stationary} &= C_{r0} \times m_v \times g \\ F_{stic-max-moving} &= C_{r0} \times C_{r0-scaler} \times m_v \times g \end{aligned} \tag{5.5}$$

The stiction input to the estimator can now be found using (5.6):

$$F_{stic} = \begin{cases} -min(F_w, F_{stic-max-stationary}) \times sgn(F_w), & \text{if } |v_v| = 0 \\ -F_{stic-max-moving} \times sgn(v_v), & \text{if } |v_v| > 0 \end{cases} \tag{5.6}$$

### 5.3.2 Two-mass Model

The most simplified model that still includes the drivetrain dynamics is the two mass model, shown in Fig. 5.21. For most vehicles the tyre rotational stiffness is very high when compared to that of the drive shaft and gearbox. For the test vehicle there is a 4.5 ratio between the two. If the tyre flexibility is combined with that of the axle, it simplifies the model, but will no longer describe the higher frequency components during transients.

The axle inertia and friction losses are no longer part of the model; as they have significant values they can be added to the vehicle mass and frictional losses. The axle inertia is converted to an equivalent mass called $m_{axle}$ and the losses are added to the $b_v$ term.

As in the above three-mass model gradient force, cornering losses and mechanical friction brake losses are not included in the two-mass model. They can still be accounted for by including them as an input added to the $F_{stic}$ input.

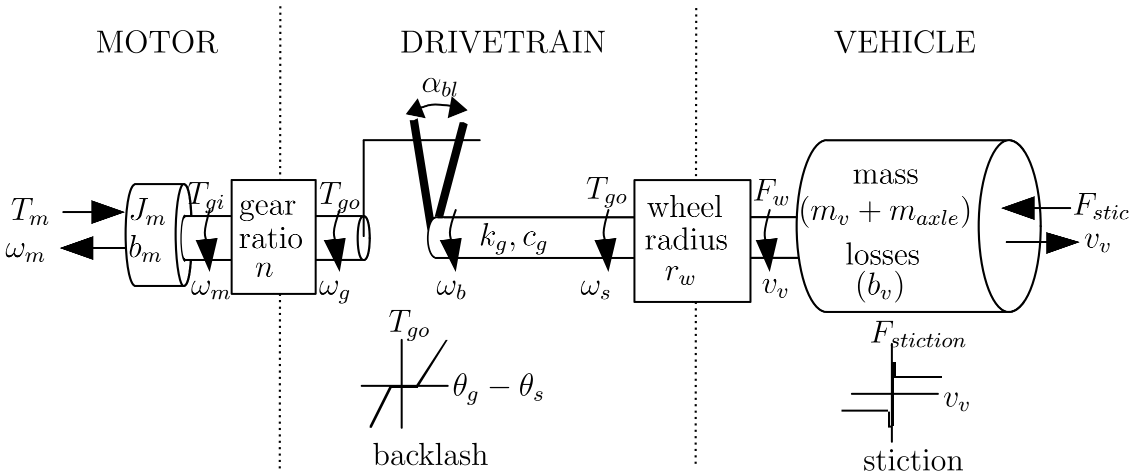

Figure 5.21: Rotating two mass model

The forces dependant on the square of vehicle speed, mainly aerodynamic drag, have to be linearised to be included within the linear vehicle model: they are then represented within the term $b_v$. This is a valid approximation due to the fact that the vehicles mainly operate at lower speeds, before the speed squared term dominates the losses. At the chosen operating point of $1500rpm$, the speed square losses only make up 4% of the total mechanical losses, shown in Fig. 5.22 and Fig. 5.23.

Rolling resistance includes a non-speed dependant term $C_{r0}$, often referred to as tyre stiction. This is a significant force acting on the vehicle at slow speeds and cannot be ignored or easily linearised: it makes up the majority of the losses acting upon the vehicle, 56% of losses at $1500rpm$ ($2.75ms^{-1}$). This is represented as a system input to the vehicle side of the model $F_{stic}$. The stiction is 40% higher when the vehicle is stationary than when it is moving, due to the extra force required to start the vehicle moving.

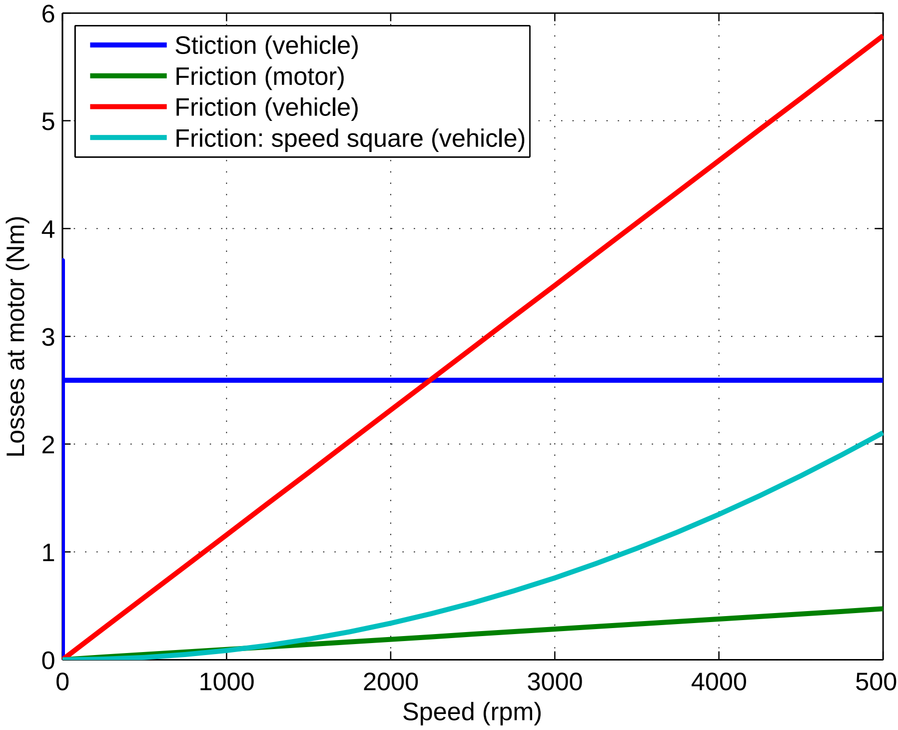

Figure 5.22: Graph showing the magnitude of torque losses referenced back to the motor

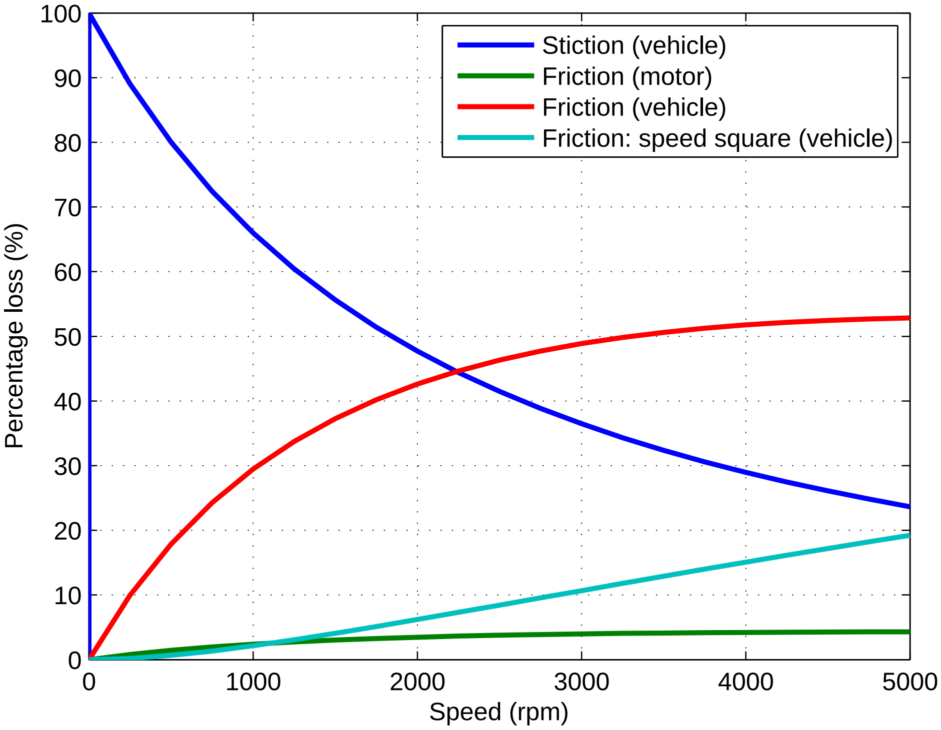

Figure 5.23: Graph showing the percentage of the losses

The differential equations ((5.7) to (5.12)) describe the vehicle model shown in Fig. 5.21, which operates in two linear modes: contact and backlash. The states are motor position ($\theta_m$) and speed ($\omega_m$), vehicle distance ($d_v$) and speed ($v_v$) and the position within the backlash ($\theta_{bl}$), which is equal to ($\theta_g - \theta_b$), shown in Fig. 5.21.

Contact mode:

$$\frac{d}{dt}\omega_m = \left(\frac{d_v}{n.r} - \frac{\theta_m}{n^2} + \frac{\theta_{bl}}{n}\right)\left(\frac{k_g}{J_m}\right) + \left(\frac{v_v}{n.r} - \frac{\omega_m}{n^2}\right)\left(\frac{c_g}{J_m}\right) - \left(\frac{\omega_m.b_m}{J_m}\right) + \left(\frac{T_m}{J_m}\right) \tag{5.7}$$

$$\frac{d}{dt}v_v = \left(\frac{\theta_m}{n.r} - \frac{d_v}{r^2} - \frac{\theta_{bl}}{r}\right)\left(\frac{k_g}{m_v}\right) + \left(\frac{\omega_m}{n.r} - \frac{v_v}{r^2}\right)\left(\frac{c_g}{m_v}\right) - \left(\frac{v_v.b_v}{m_v}\right) - \left(\frac{F_{stiction}}{m_v}\right) \tag{5.8}$$

$$\frac{d}{dt}\theta_{bl} = 0 \tag{5.9}$$

Backlash mode:

$$\frac{d}{dt}\omega_m = -\left(\frac{\omega_m.b_m}{J_m}\right) + \left(\frac{T_m}{J_m}\right) \tag{5.10}$$

$$\frac{d}{dt}v_v = -\left(\frac{v_v.b_v}{m_v}\right) - \left(\frac{F_{stiction}}{m_v}\right) \tag{5.11}$$

$$\frac{d}{dt}\theta_{bl} = \left(\frac{\theta_m}{n} - \frac{d_v}{r} - \theta_{bl}\right)\left(\frac{k_g}{c_g}\right) + \left(\frac{\omega_m}{n}\right) - \left(\frac{v_v}{r}\right) \tag{5.12}$$

This model (or other derivations of the two-mass model) are used for a lot of the estimation work within this thesis as it was found to be a good compromise between complexity and calculation time required.

### 5.3.3 Single-mass Model

This is the simplest model of a vehicle, as it ignores all drivetrain dynamics and reduces the vehicle into a single first order system, as shown in Fig. 5.24. This model is useful when you are only concerned with the inertia of the vehicle and the driving losses, such as when estimating mass. It has the disadvantage that the motor speed will not be equal to the measured motor speed during transients, so any estimator has to not respond to motor oscillations.

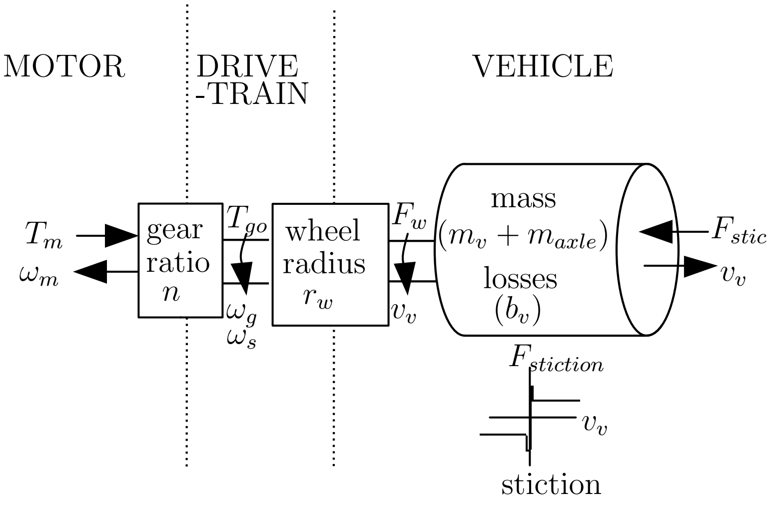

Figure 5.24: Rotating one mass model

## 5.4 Summary

For the experimental work a test vehicle was used to carry out tests indoors. This vehicle and the test setup is described in section 5.1, some additional sensors were added to the vehicle to help with the analysis of the results. There is a significant assumption used that the torque demand will be equal to the actual motor output torque, but this has been tested to confirm that it is, although it could be improved if the current demands are compensated for when the torque output is no longer linear with quadrature current ($I_q$).

The vehicle parameters are measured in section 5.2. Some of these parameters are found quite accurately by measuring directly such as tyre radius $r$ and axle stiffness $k_g$ and others fairly accurately using the Genetic Algorithm ($J_m$, $b_m$, $T_{stic-m}$, $c_g$, $\theta_{bl}$). Other parameters were determined less accurately by comparing the simulation to experimental results, although most of these parameters are less important to the accuracy of the model.

Finally in this chapter in section 5.3 the simplifications used to create the real-time implemented vehicle model are given. These reduced complexity models are then used in the following 2 chapters for estimating vehicle states and improving its control.

# Chapter 6

# Vehicle Speed and Mass Estimation

## 6.1 Introduction

In the literature review (chapter 3) several papers showed how improvements can be made to speed control systems where load speed feedback is available, in this case the load speed will be the vehicle speed. When comparing to an industrial drive system, such as a rolling mill, which is typically modelled similarly to the vehicle two-mass model 5.3.2. There is the potential for there to be a greater number of unmodelled parameter changes and changes in load torque/force for the vehicle application, whereas the industrial application will typically only have a changing load torque or inertia.

Vehicle speed is difficult to directly measure, although in steady state conditions the motor speed can be assumed to be the vehicle speed, taking into account the gear ratio and tyre radius. In transient conditions, where the motor torque or vehicle load is changing, they will not be equal. If the wheels lose traction with the road, the vehicle speed will also be different to the motor speed. In some instances the non-driving wheels can be used to give vehicle speed, but most industrial electric vehicles do not have wheel speed sensors. Using knowledge of the drivetrain parameters allows estimation techniques to be applied to estimate the vehicle speed. In the next chapter the state estimator is then used to help improve the vehicle's driving response and also prevent loss of traction with the tyres.

This chapter will start by considering vehicle speed estimation using Kalman Filtering (KF), where all vehicle parameters and external forces remain fixed. It will then consider mass changes and how estimating this can improve the speed estimation. The importance of the Kalman estimator noise matrices are then looked at, and a number of tuning methods considered. Finally alternative mass estimation methods are developed in section.

## 6.2 The Kalman Filter State Estimator

### 6.2.1 State Space Estimators

Most system models, such as the vehicle model in 5.3.2, contain more than one differential equation and more than one input and output. This makes such systems difficult to analyse and solve directly and a simpler approach is to use state space representation [86]. This allows for improved readability and a more compact notation; it also makes it easier to calculate in real-time and also to find a discrete form.

The standard State Space (SS) notation [87] is given by (6.1) and (6.2). The subscript $_\mathbf{c}$ is used to indicate that the matrices are in the continuous time domain:

$$\dot{\mathbf{x}}(t) = \mathbf{A_c}(t)\mathbf{x}(t) + \mathbf{B_c}(t)\mathbf{u}(t) \tag{6.1}$$

$$\mathbf{y}(t) = \mathbf{C_c}(t)\mathbf{x}(t) + \mathbf{D_c}(t)\mathbf{u}(t) \tag{6.2}$$

If assuming that it is Linear Time Invariant (LTI), this can be simplified to (6.3) and (6.4):

$$\dot{\mathbf{x}}(t) = \mathbf{A_c}\mathbf{x}(t) + \mathbf{B_c}\mathbf{u}(t) \tag{6.3}$$

$$\mathbf{y}(t) = \mathbf{C_c}\mathbf{x}(t) + \mathbf{D_c}\mathbf{u}(t) \tag{6.4}$$

For a system with $n$ states, $q$ inputs and $p$ outputs:

- $\mathbf{x}$ is the $n \times 1$ state vector
- $\mathbf{u}$ is the $q \times 1$ input vector
- $\mathbf{y}$ is the $p \times 1$ output vector
- $\mathbf{A_c}$ is the $n \times n$ state matrix
- $\mathbf{B_c}$ is the $n \times q$ input matrix
- $\mathbf{C_c}$ is the $p \times n$ output matrix

- $\mathbf{D_c}$ is the $p \times q$ feed-forward matrix

After removing the matrix **D**, as it is rarely applicable in a real system, Fig. 6.1 results:

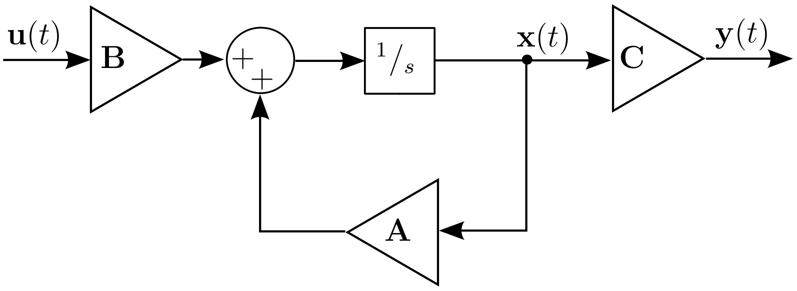

Figure 6.1: Continuous state space diagram

The above continuous case needs to be converted to its discrete equivalent in order to implement in real time. If assuming that it is Linear Time Invariant (LTI), the discrete state space form is given by (6.5) and (6.6):

$$\mathbf{x}(k+1) = \mathbf{A_d}\mathbf{x}(k) + \mathbf{B_d}\mathbf{u}(k) \tag{6.5}$$

$$\mathbf{y}(k) = \mathbf{C_d}\mathbf{x}(k) \tag{6.6}$$

Similar to the continuous case above, for a system with $n$ states, $q$ inputs, $p$ outputs and $T_s$ sample rate [88], the subscript $_\mathbf{d}$ is used to indicate the matrices are in the discrete domain, but from now on it has been omitted in some equations for clarity:

- $\mathbf{x}$ is the $n \times 1$ state vector
- $\mathbf{u}$ is the $q \times 1$ input vector
- $\mathbf{y}$ is the $p \times 1$ output vector
- $\mathbf{A_d}$ is the $n \times n$ state transition matrix and is found by (6.7) using the Taylor series expansion
- $\mathbf{B_d}$ is the $n \times q$ input matrix and is found by (6.8)
- $\mathbf{C_d}$ is the $p \times n$ output matrix and is found by (6.9)

$$\mathbf{A_d} = e^{\mathbf{A}T_s} = \mathbf{I} + \mathbf{A}T_s + \frac{\mathbf{A}^2T_s^2}{2!} + \frac{\mathbf{A}^3T_s^3}{3!} + ... \tag{6.7}$$

$$\mathbf{B_d} = \int_0^{T_s} e^{\mathbf{A}n} dn.\mathbf{B} = \left(\mathbf{I}T_s + \frac{\mathbf{A}T_s^2}{1! \times 2} + \frac{\mathbf{A}^2T_s^3}{2! \times 3} + \frac{\mathbf{A}^3T_s^4}{3! \times 4} + ...\right).\mathbf{B} \tag{6.8}$$

$$\mathbf{C_d} = \mathbf{C} \tag{6.9}$$

The discrete equations above (6.5) and (6.6) can be drawn as Fig. 6.2:

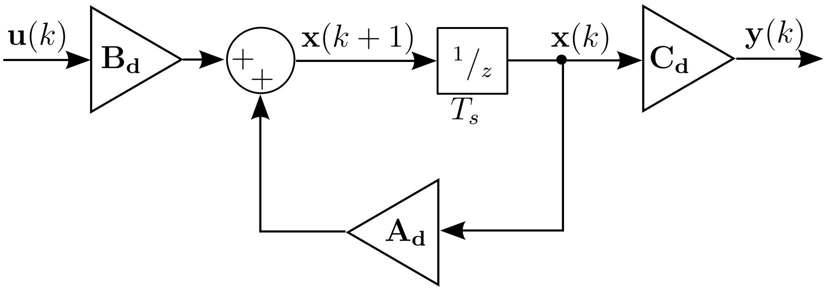

Figure 6.2: Discrete state space diagram

### 6.2.2 Kalman Filter Theory

The KF was first published in 1960 [89] as a modification of earlier filtering techniques such as the Wienner filter which is derived in the frequency domain based upon power spectral densities, whereas the Kalman filter is derived from the time domain, based upon the covariances of the noise, assuming it is Gaussian white noise [90].

The filter is used to estimate unmeasured states or reduce noise on measured signals, but the original algorithm only works for linear systems, requires a large number of calculations for real time implementation and the noise statistics need to be known for it to work well.

For an LTI system given by (6.10) and (6.11), with process noise $w$ given in vector $\mathbf{w_{k-1}}$ and measurement noise $v$. They are assumed to be random white noise and have covariances of $\mathbf{Q}$ for the process noise and $\mathbf{R}$ for the measurement noise given in vector $\mathbf{v_k}$.

$$\mathbf{x_k} = \mathbf{A_d}\,\mathbf{x_{k-1}} + \mathbf{B_d}\,\mathbf{u_{k-1}} + \mathbf{w_{k-1}} \tag{6.10}$$

$$\mathbf{y_k} = \mathbf{C_d}\,\mathbf{x_k} + \mathbf{v_k} \tag{6.11}$$

The full Kalman Filter (KF) algorithm is given by the following equations (6.12) to (6.16) [91], maintaining the earlier notation in section 6.2.1 of $\mathbf{A_d}$, $\mathbf{B_d}$, and $\mathbf{C_d}$, the superscript $^-$ refers to priori and $^+$ poseriori, the covariance is given in $\mathbf{P}$, the state estimation in $\mathbf{x}$ and the Kalman gain in $\mathbf{K}$:

$$\mathbf{P_k^-} = \mathbf{A_d}\,\mathbf{P_{k-1}^+}\,\mathbf{A_d^T} + \mathbf{Q} \tag{6.12}$$

$$\mathbf{K_k} = \mathbf{P_k^-}\,\mathbf{C_d^T}\left(\mathbf{C_d}\,\mathbf{P_k^-}\,\mathbf{C_d^T} + \mathbf{R}\right)^{-1} \tag{6.13}$$

$$\mathbf{\hat{x}_k^-} = \mathbf{A_d}\,\mathbf{\hat{x}_{k-1}^+} + \mathbf{B_d}\,\mathbf{u_{k-1}} \tag{6.14}$$

$$\mathbf{\hat{x}_k^+} = \mathbf{\hat{x}_k^-} + \mathbf{K_k}\left(\mathbf{y_k} - \mathbf{C_d}\,\mathbf{\hat{x}_k^-}\right) \tag{6.15}$$

The equation in 6.16 has two forms [91]: the first is the ‘Joseph stabilised’ form and has been shown to be more stable, the second form requires less calculations.

$$\begin{aligned} \mathbf{P_k^+} &= \left(\mathbf{I} - \mathbf{K_k}\,\mathbf{C_d}\right)\mathbf{P_k^-}\left(\mathbf{I} - \mathbf{K_k}\,\mathbf{C_d}\right)^T + \mathbf{K_k}\,\mathbf{R}\,\mathbf{K_k^T} \\ \mathbf{P_k^+} &= \left(\mathbf{I} - \mathbf{K_k}\,\mathbf{C_d}\right)\mathbf{P_k^-} \end{aligned} \tag{6.16}$$

The equations (6.14) and (6.15) can be combined to give the diagram in Fig. 6.3, both outputs of the lower left sum block are the same, $\mathbf{\hat{x}_k^-}$:

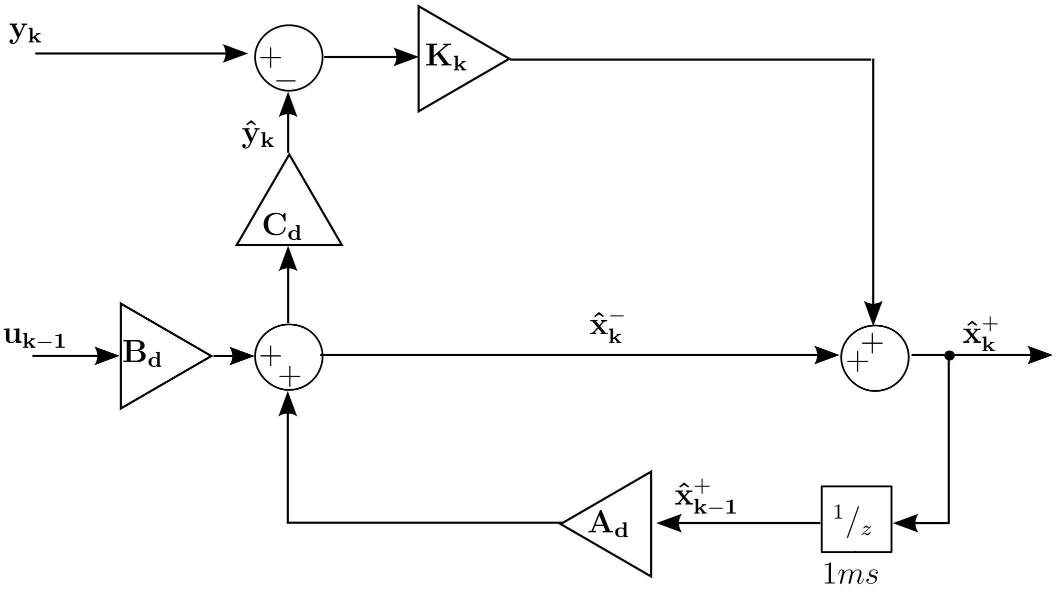

Figure 6.3: Kalman Filter diagram

As can be seen by the equations (6.12) to (6.16), the KF algorithm contains a great number of matrix and vector operations. For a simple 1 state, 1 input and output system there will be about 20 add and multiply operations with 1 divide. For the 5 state system in section 6.2.3, with 2 inputs and outputs there are nearly 1200 multiply and add operations, not including calculating the matrix inverse required in equation (6.13) as this depends upon the method used. This makes real time implementation difficult for systems with fast dynamics requiring short sample times.

As the calculation of the Kalman gain does not depend upon the inputs **u** or the measurements **y** it can be calculated off-line, provided the system is LTI or approximated to be [91]. In the equations (6.12), (6.13) and (6.16) above, the only inputs to them are $\mathbf{A_d}$, $\mathbf{Q}$, $\mathbf{C_d}$ and $\mathbf{R}$. If these matrices are assumed to be fixed then $\mathbf{K_k}$ will converge to a fixed value. It should be noted that when the $\mathbf{K_k}$ value would have been converging to a fixed value, replacing it with the fixed (steady state) version now gives a non-optimal filter during this time, however this happens quickly and it would normally have converged before the vehicle has started to drive. If using a fixed gain in the system in 6.2.3, commonly referred to as Steady State KF, there are only 75 maths operations required, instead of the 1200 for running the full KF algorithm.

### 6.2.3 Vehicle Dynamics State Space Model - 5 states

The two mass vehicle model defined in 5.3.2 can be converted into an equivalent state space model, as the differential equations are already arranged into 5 states: $\theta_m$, motor position; $\omega_m$, motor speed; $d_v$, vehicle distance; $v_v$, vehicle velocity and $\theta_{bl}$, gear backlash position. Even though the model is a simplified representation of a vehicle, it still contains two non-smooth non-linearities: the gear backlash and the tyre stiction.

The backlash is modelled based on the physical backlash model [9] and [78]; which allows for the backlash to be entered whilst the axle is still unwinding from previously transferring a force along it. As the backlash can only have one of two states: contact or backlash, it is preferable to make the 'in backlash' condition a mode and switch between two state matrices **A** depending upon the mode. Having these modes is necessary as when traversing the backlash there is a disconnect between the model input and output. It can be clearly seen in matrix $\mathbf{A_{bl}}$, (6.23) that during the backlash mode that the motor and vehicle are only affected by their own inputs and friction terms.

The stiction force part of the rolling resistance is calculated outside of the state space model and fed in as an input ($F_{stic}$); the stiction is either zero, equal to the force being applied at the wheels ($F_w$) or one of two levels applied in either direction. A higher stiction level is applied whilst stationary ($F_{stic-max-stationary}$), but it is clamped to ensure that it does not exceed the force at the wheels; it is equal and opposite to the force at the wheels until it reaches the max level ($F_{stic-max-stationary}$). When the vehicle then starts to move it reduces to the vehicle moving level ($F_{stic-max-moving}$) and is applied to oppose the vehicle velocity direction.

The maximum stiction levels are calculated in (6.17) based upon the equations developed in 4.3.4:

$$\begin{aligned} F_{stic-max-stationary} &= C_{r0} \times m_v \times g \\ F_{stic-max-moving} &= C_{r0} \times C_{r0-scaler} \times m_v \times g \end{aligned} \tag{6.17}$$

The stiction input to the estimator can now be found using (6.18):

$$F_{stic} = \begin{cases} -min(F_w, F_{stic-max-stationary}) \times sgn(F_w), & \text{if } |v_v| = 0 \\ -F_{stic-max-moving} \times sgn(v_v), & \text{if } |v_v| > 0 \end{cases} \tag{6.18}$$

In order to compromise between accurately modelling fast transients of the gearbox such as when the cogs collide and limiting the processor requirements for real time implementation a sample rate of $1ms$ was chosen. It was found in simulation that any slower than this was found to cause large errors in the backlash estimation which made predicting entries into backlash mode and exits difficult, as a typical backlash traversal time is in the tens of milliseconds and the other considered option of $5ms$ would be too slow.

The continuous state space model of the vehicle is shown below [9], where the states are motor angle $\theta_m$, motor angular velocity $\omega_m$, vehicle distance $d_v$, vehicle velocity $v_v$ and backlash angle $\theta_{bl}$ (6.19), the inputs are motor torque $T_m$ and stiction force $F_{stic}$ (6.20) and the measured states are $\theta_m$ and $\omega_m$ (6.21).

$$\mathbf{x} = \begin{bmatrix} \theta_m \\ \omega_m \\ d_v \\ v_v \\ \theta_{bl} \end{bmatrix} \tag{6.19}$$

$$\mathbf{y} = \begin{bmatrix} \theta_m \\ \omega_m \end{bmatrix} \tag{6.20}$$

$$\mathbf{u} = \begin{bmatrix} T_m \\ F_{stic} \end{bmatrix} \tag{6.21}$$

The state transition matrices for the two modes are contact mode (6.22) and backlash mode (6.23). In the contact mode, the backlash position state $\theta_{bl}$ cannot change as the system remains in contact, the other two position states are given by integrating their equivalent speed states. In the backlash mode, all of the axle stiffness $k_g$ and damping $c_g$ terms are removed, leaving just the friction terms. Although the backlash can be entered before the shaft has fully unwound, as it is assumed to have no inertia, it cannot generate any force. The backlash position state is now active and calculates the size of the backlash gap based upon speeds and position of each side of the backlash.

$$\mathbf{A_{co}} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ -\frac{k_g}{J_m.n^2} & -\frac{b_m + {}^{c_g}/_{n^2}}{J_m} & \frac{k_g}{J_m.n.r} & \frac{c_g}{J_m.n.r} & \frac{k_g}{J_m.n} \\ 0 & 0 & 0 & 1 & 0 \\ \frac{k_g}{m_v.n.r} & \frac{c_g}{m_v.n.r} & -\frac{k_g}{m_v.r^2} & -\frac{b_v + {}^{c_g}/_{r^2}}{m_v} & -\frac{k_g}{m_v.r} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.22}$$

$$\mathbf{A_{bl}} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -\frac{b_m}{J_m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -\frac{b_v}{m_v} & 0 \\ \frac{k_g}{c_g.n} & \frac{1}{n} & -\frac{k_g}{c_g.r} & -\frac{1}{r} & -\frac{k_g}{c_g} \end{pmatrix} \tag{6.23}$$

The input matrix (6.24) allows for motor torque and vehicle losses (specifically rolling resistance stiction) to be applied to the model.

$$\mathbf{B_{co}} = \mathbf{B_{bl}} = \begin{pmatrix} 0 & 0 \\ \frac{1}{J_m} & 0 \\ 0 & 0 \\ 0 & \frac{1}{m_v} \\ 0 & 0 \end{pmatrix} \tag{6.24}$$

The output matrix (6.25) simply separates the measured states from the unmeasured states; it remains the same for contact and backlash modes.

$$\mathbf{C_{co}} = \mathbf{C_{bl}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} \tag{6.25}$$

In order to switch between the modes the following condition is checked continually at every time step, $\omega_{bl}$ is calculated from the rate of change or $\theta_{bl}$ and $\alpha_{bl}$ is a fixed value (backlash size) (6.26).

$$\text{in backlash} = IF \left| \left( \frac{\omega_m}{n} - \frac{v_v}{r} - \omega_{bl} \right) \left( \frac{c_g}{k_g} \right) + \left( \frac{\theta_m}{n} - \frac{d_v}{r} \right) \right| < \left( \frac{\alpha_{bl}}{2} \right) \tag{6.26}$$

Although it is common to only take the first approximation to the Taylor series for the discrete form of the model, this can lead to inaccuracies in the discrete model. As this was being calculated off-line, up to the 5th order was used of the Taylor expansion given in section 6.2.1: (6.7) and (6.8). If a faster sample rate than $1kHz$ was used a more approximate discrete conversion could be used as it is of less importance.

The microprocessor used for this project can only perform 32bit fixed point maths, see section 5.1.2. This can cause accuracy issues at the extremes of the allowed number range and stability issues, for example if a multiply or add operation saturates the available range and is not compensated for in the software. By default 16.16 scaling is used where the greatest signed number is $32,767$ and the smallest number that can be represented is $15.259 \times 10^{-6}$. The model defined above uses SI or SI derived units for its states and parameters, so the motor speeds are in $rads^{-1}$ and the vehicle speed in $ms^{-1}$. For the test vehicle this equates to a difference in scaling of around 60 due to the gear ratio and tyre radius; this is the same as the difference in scaling between motor position and vehicle distance and motor torques and vehicle wheel force. The continual scaling from one unit to another leads to a loss of accuracy, equivalent to around 6 bits of precision ($2^6 = 64$). In order to remove this inaccuracy, a single point of reference can be chosen, converting all speed states to motor speed with units in $rpm$, and torques and forces in $100ths$ of a $Nm$, referenced to the motor. This does now mean that all the position and distance states are in units *rpm seconds* as they are the integral of the speed states. To increase the accuracy further the fixed point formatting was changed to use 14.18 scaling giving a maximum number of 8192 and smallest number of $3.815 \times 10^{-6}$ (four times smaller than the default).

To convert all of the matrices to the new scaling the following convention is used shown being applied to the $\mathbf{A_{co}}$ in (6.27) and $\mathbf{B_{co}}$ in (6.28). The scalers are multiplied by every row or column to change the units for each state. The rescaling is also applied to $\mathbf{A_{bl}}$ the same as to $\mathbf{A_{co}}$. The $\mathbf{C}$ matrix does not need rescaled as the outputs are now all in *rpm*.

$$\begin{array}{cc} & \begin{array}{ccccc} \begin{array}{c} m_{rpm} \\ \downarrow \\ m_{rads^{-1}} \end{array} & \begin{array}{c} m_{rpm} \\ \downarrow \\ m_{rads^{-1}} \end{array} & \begin{array}{c} m_{rpm} \\ \downarrow \\ v_{ms^{-1}} \end{array} & \begin{array}{c} m_{rpm} \\ \downarrow \\ v_{ms^{-1}} \end{array} & \begin{array}{c} m_{rpm} \\ \downarrow \\ s_{rads^{-1}} \end{array} \end{array} \\ \begin{array}{r} m_{rads^{-1}} \rightarrow m_{rpm} \\ m_{rads^{-1}} \rightarrow m_{rpm} \\ v_{ms^{-1}} \rightarrow m_{rpm} \\ v_{ms^{-1}} \rightarrow m_{rpm} \\ s_{rads^{-1}} \rightarrow m_{rpm} \end{array} & \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ -\frac{k_g}{J_m.n^2} & -\frac{b_m + {}^{c_g}/_{n^2}}{J_m} & \frac{k_g}{J_m.n.r} & \frac{c_g}{J_m.n.r} & \frac{k_g}{J_m.n} \\ 0 & 0 & 0 & 1 & 0 \\ \frac{k_g}{m_v.n.r} & \frac{c_g}{m_v.n.r} & -\frac{k_g}{m_v.r^2} & -\frac{b_v + {}^{c_g}/_{r^2}}{m_v} & -\frac{k_g}{m_v.r} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \tag{6.27}$$

$$\begin{array}{cc} & \begin{array}{cc} \begin{array}{c} m_{Nm} \\ \downarrow \\ m_{Nm-100} \end{array} & \begin{array}{c} v_{N} \\ \downarrow \\ m_{Nm-100} \end{array} \end{array} \\ \begin{array}{r} m_{rads^{-1}} \rightarrow m_{rpm} \\ m_{rads^{-1}} \rightarrow m_{rpm} \\ v_{ms^{-1}} \rightarrow m_{rpm} \\ v_{ms^{-1}} \rightarrow m_{rpm} \\ s_{rads^{-1}} \rightarrow m_{rpm} \end{array} & \begin{pmatrix} 0 & 0 \\ \frac{1}{J_m} & 0 \\ 0 & 0 \\ 0 & \frac{1}{m_v} \\ 0 & 0 \end{pmatrix} \end{array} \tag{6.28}$$

The following scaling factors are used in the conversions above:

$$m_{rpm} \rightarrow m_{rads^{-1}} = \frac{pi}{30} \tag{6.29}$$

$$m_{rpm} \rightarrow v_{ms^{-1}} = \frac{pi}{30}\frac{r_w}{n} \tag{6.30}$$

$$m_{rpm} \rightarrow s_{rads^{-1}} = \frac{pi}{30}\frac{1}{n} \tag{6.31}$$

$$m_{rads^{-1}} \rightarrow m_{rpm} = \frac{30}{pi} \tag{6.32}$$

$$v_{ms^{-1}} \rightarrow m_{rpm} = \frac{30}{pi}\frac{n}{r_w} \tag{6.33}$$

$$s_{rads^{-1}} \rightarrow m_{rpm} = \frac{30}{pi}n \tag{6.34}$$

$$m_{Nm} \rightarrow m_{Nm-100} = \frac{1}{100} \tag{6.35}$$

$$v_N \rightarrow m_{Nm-100} = \frac{1}{100}\frac{n}{r_w} \tag{6.36}$$

As the position / distance increments continually, the software has to be able to deal with this when they wrap around as the they have exceeded 8192; but this is easy to identify and account for.

After rescaling the model is converted to its discrete form based upon the method in section 6.2.1.

The matrices after going through this rescaling process can be found in the appendix A.1.

Ideally the Kalman Filter would be implemented on a floating point processor as this would remove the need for all the extra scaling above. Using fixed point potentially adds more process noise into the model which will cause the matrix ($Q$) to be larger and therefore the Kalman filter will rely less on the model and more on the measurements. As the fixed point scaling is chosen appropriately, the effect of using fixed point was found to be small, when comparing the estimator accuracy to simulation work.

### 6.2.4 Switched Fixed Kalman Gain Estimator

The Kalman filter algorithm includes a large number of matrix multiplications, which increases greatly with the number of states estimated and also the number of measured states. It is required to be implemented on a microcontroller (Texas Instruments TMS320C2811), which also performs numerous other tasks such as the motor control. In order to reduce the computational requirement to a minimum, a fixed Kalman gain is used. The gain can be calculated off-line through solving the Riccati equation [19], provided the matrices $\mathbf{A_d}$, $\mathbf{C_d}$, $\mathbf{Q}$ and $\mathbf{R}$ remain constant and are known.

The model described in section 5.3.2 and the previous section 6.2.3 suggests that gear backlash can be modelled with two linear modes [9]: switching between these modes on-line leads to a non-smooth system. This would normally be an issue for the KF and lead to instabilities, but as the Kalman gain is fixed this is less of an issue. During backlash mode the system is not observable or controllable due to the disconnect between the motor and vehicle; there is only feedback and controllable inputs on the motor side. As this only occurs for very short intervals the Kalman gain is set to zero and the estimator operates on an open loop basis. See (6.37) and (6.38) for the KF equations used during both conditions [91].

$$\mathbf{\hat{x}_k^+} = \left(\mathbf{I} - \mathbf{K_{d\_co}}\,\mathbf{C_d}\right)\left(\mathbf{A_{d\_co}}\,\mathbf{\hat{x}_{k-1}^+} + \mathbf{B_d}\,\mathbf{u_{k-1}}\right) + \mathbf{K_{d\_co}}\,\mathbf{y_k} \tag{6.37}$$

$$\mathbf{\hat{x}_k^+} = \mathbf{A_{d\_bl}}\,\mathbf{\hat{x}_{k-1}^+} + \mathbf{B_d}\,\mathbf{u_{k-1}} \tag{6.38}$$

### 6.2.5 Different Sample Rates and Delays

The encoder is used to generate the speed feedback every $5ms$. The test vehicle is fitted with a Hall sensor encoder which has a state transition 6 times every electrical rotation or 24 (as the motor has 4 pole pairs) every mechanical rotation. Therefore it is not possible to calculate a new speed at a faster rate than $5ms$ at all vehicle speeds. The speed is then integrated to give the estimator the position feedback, as absolute position is not required. Although the test vehicle also has a higher accuracy incremental encoder fitted for test purposes, it is not standard to have this feedback available and only the lower resolution hall effect sensor.

The drivetrain system has fast changing backlash dynamics that requires the estimator to run at least $1ms$ to correctly estimate the transition into and out of backlash mode, as the backlash duration is typically around $20ms$ for the test vehicle used. This means that the correction part of the estimator runs five times slower ($5ms$) than the state propagation part, see figure 6.3. The fixed Kalman gain is therefore calculated assuming a $5ms$ version of $\mathbf{A_d}$ and $\mathbf{B_d}$, and the correction part is divided by 5 so it can be applied over the next 5 iterations of the estimator, every $1ms$. This also allows the PI speed controller to run at a faster rate than the feedback, therefore increasing its bandwidth.

In Fig. 6.4 it is shown how the corrector part of the estimator is running a different rate to the rest.

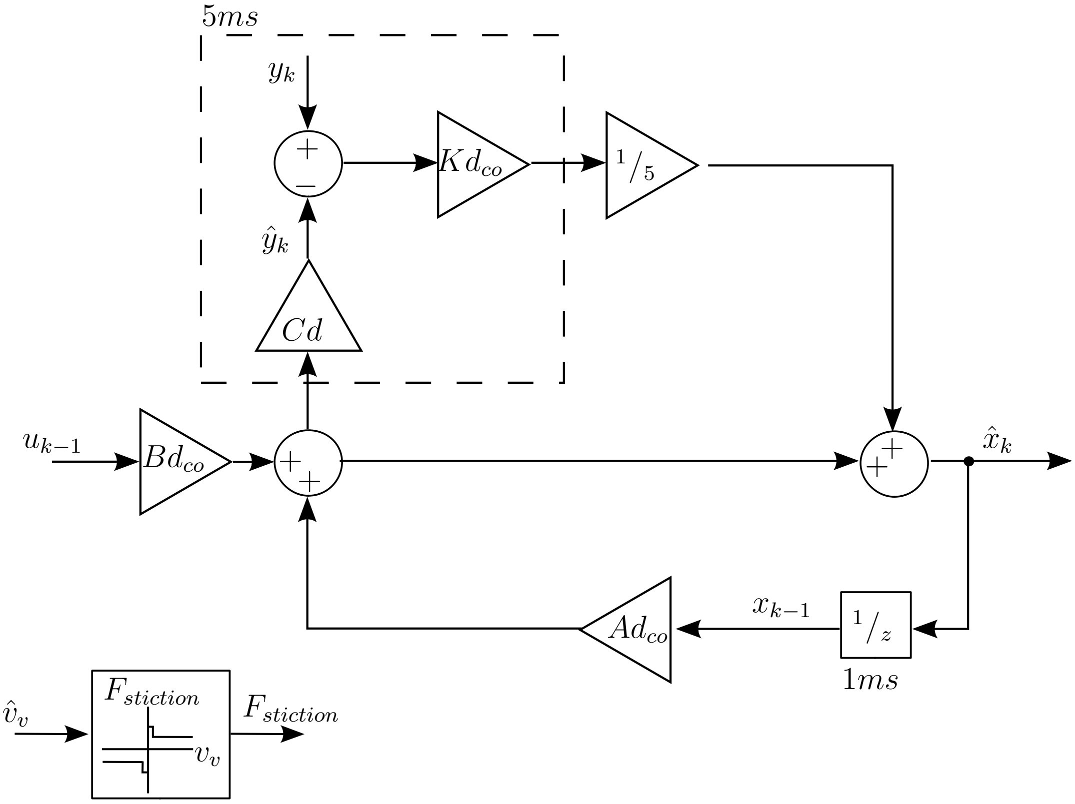

Figure 6.4: Kalman Filter diagram with more than one sample rate

In Fig. 6.5 the estimator now has the switched mode shown from the previous section, these are shown as switches on the corrector and the outputs of the state transition matrix.

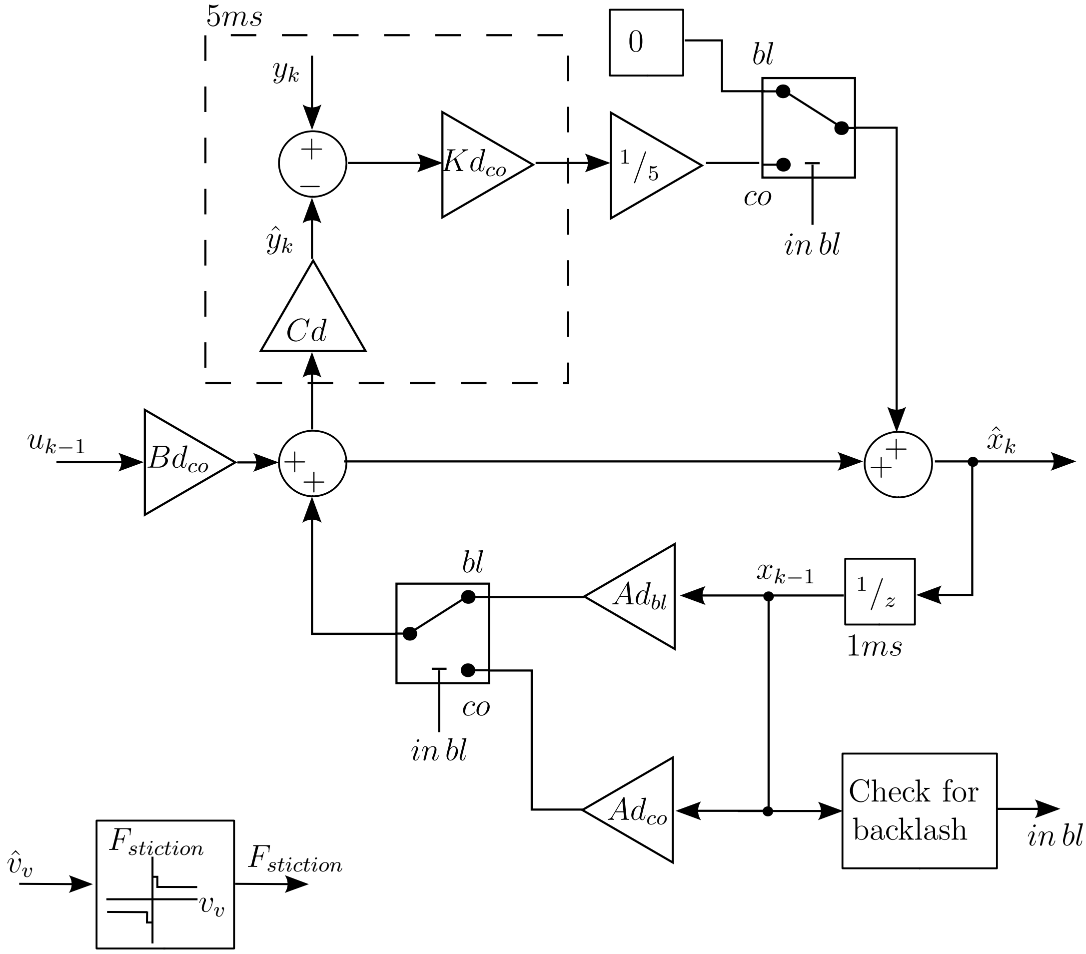

Figure 6.5: Kalman Filter diagram with more than one sample rate and backlash mode

### 6.2.6 Determining the Noise Matrices

The Kalman Filter requires information on the process and measurement noises of the system. The measurement noise covariance $\mathbf{R_{co}}$ (6.39) can be measured from recorded data of the motor encoder; by separating the noise from the signal and measuring the noise variance. However, obtaining the process noise $\mathbf{Q_{co}}$ (6.40) is less well defined and cannot be directly calculated or measured. A trial and error method has been used to generate suitable values for $\mathbf{Q}$ initially. In both $\mathbf{Q}$ and $\mathbf{R}$ only diagonal entries are used as they have the most significant effect on the operation of the filter.

$$\mathbf{R_{co}} = \begin{pmatrix} R_1 & 0 \\ 0 & R_2 \end{pmatrix} \tag{6.39}$$

$$\mathbf{Q_{co}} = \begin{pmatrix} Q_1 & 0 & 0 & 0 & 0 \\ 0 & Q_2 & 0 & 0 & 0 \\ 0 & 0 & Q_3 & 0 & 0 \\ 0 & 0 & 0 & Q_4 & 0 \\ 0 & 0 & 0 & 0 & Q_5 \end{pmatrix} \tag{6.40}$$

To separate the noise (for matrix $\mathbf{R_{co}}$) from the time varying signal a number of steps are applied to the data, shown in Fig. 6.6. In this application the signals are all quite slow moving vehicle speeds, which are quite low frequency, they are significantly different from that of the noise making it easier to separate the two signals. For the first stage the signal is passed through a 2nd order low pass filter with a cut-off of around $6.5Hz$, this filtered signal is then subtracted from the shifted unfiltered signal. It is shifted by $\sim 35ms$ as this is approximately the same lag as that of the filter for the measured waveform. Also $6.5Hz$ was chosen as it gives an approximate delay that is a multiple of $5ms$ (the sample rate of the measured signal) so that the delay can accurately match the phase shift of the filtered signal. The second stage involves a 2nd order high pass filter with a cut-off of $10Hz$, therefore leaving just the noise signal. The variance of the noise signal can be measured directly in MATLAB and used for determining the **R** matrix.

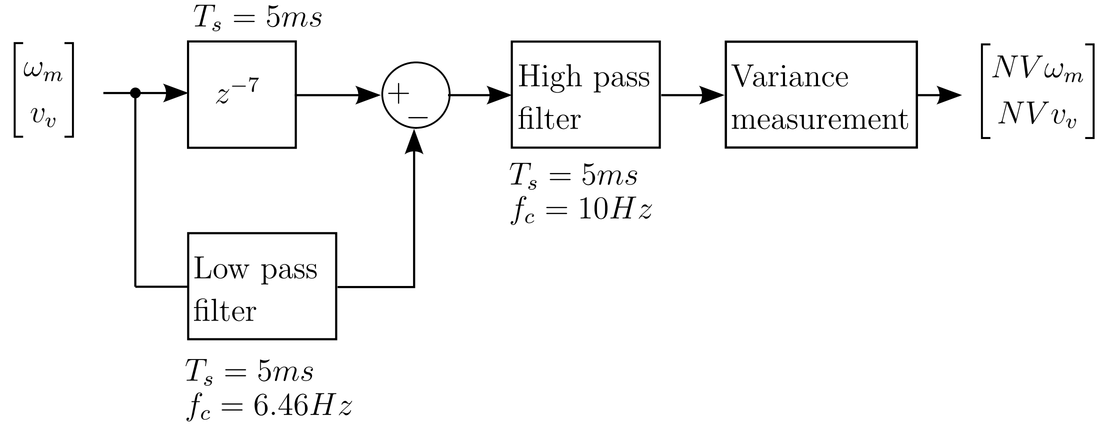

Figure 6.6: Diagram showing how the noise is extracted from the signal

The magnitude of the **Q** matrix is mainly determined by how accurate the model parameters are as incorrect modelling is a form of process noise and also how well the linear model fits the real non-linear system. The **R** matrix is based upon the noise from the encoder speed measurement. As this uses the hall-effect encoder, the speed is based on timing between edges received.

### 6.2.7 Simulation Results

The speed estimator performance has been analysed over a number of simulation runs with five operating speeds in closed loop speed mode, and then repeated with different mass and gradient changes. For the tests done in speed control a fixed rate acceleration, constant speed then deceleration is used. Due to the limited space available that only allowed for driving a fixed distance these style of tests were used for experimental work and so the same profile was used for experimental work for comparison. The main issues seen by vehicle model (estimator) errors are either seen when accelerating due to the mass being incorrect or during steady state due to the losses being incorrect. This simple profile covers these two main scenarios where the estimation errors will be seen.

If the vehicle is driven with no changes in dynamics, such as no gradient or mass change the estimate is seen to be very accurate, the speed error never exceeds $70rpm$ and is typically 1% to 2% of the vehicle speed, shown in Fig. 6.7. Note the scaling of the speed error axis changes between figures.

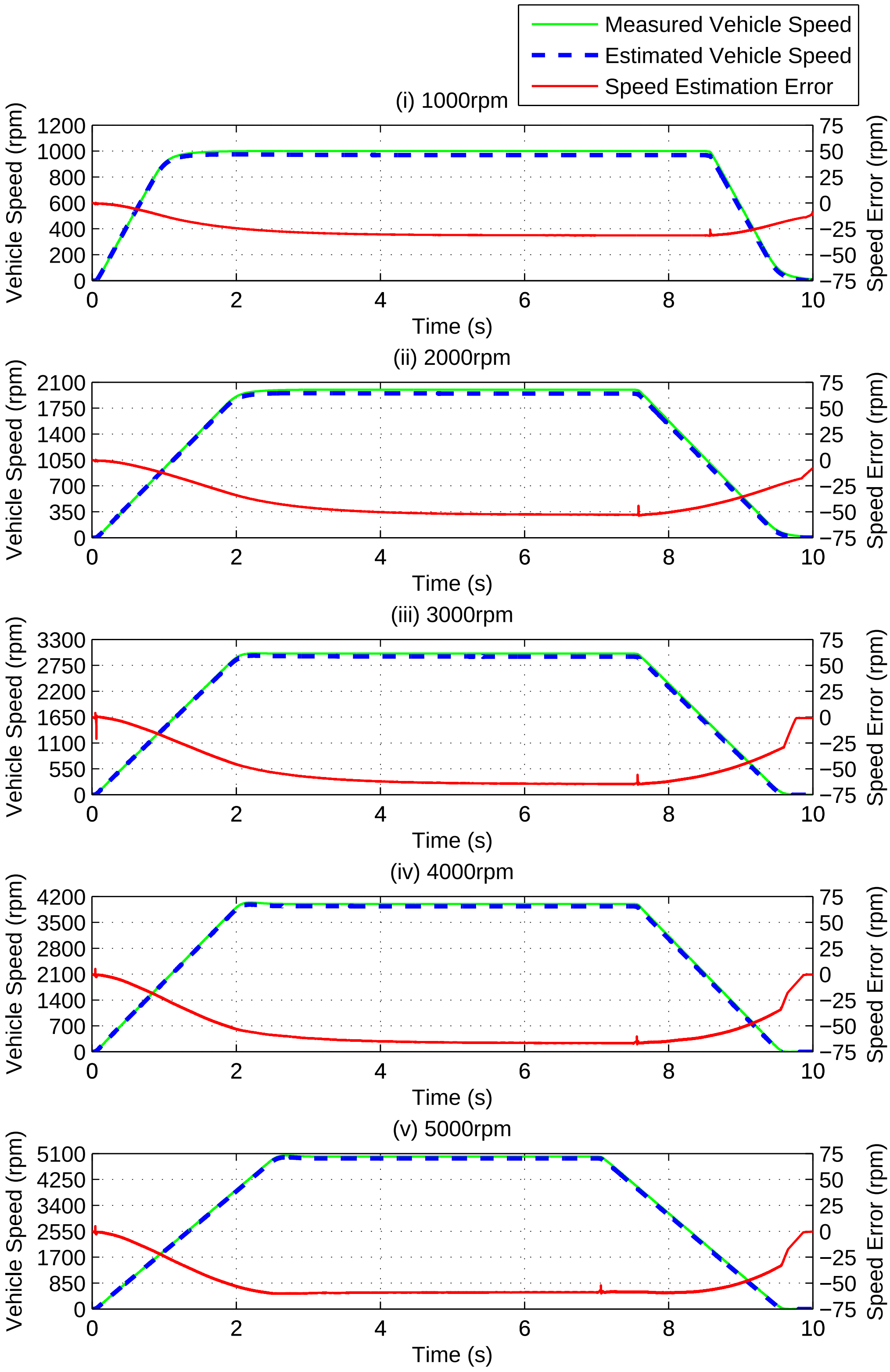

Figure 6.7: Simulation results of speed estimation with no load mass and no gradient

If a load increase of $200kg$ is added to the vehicle, the estimation accuracy is poor especially during acceleration. A maximum speed estimation error of $600rpm$ is seen and in percentage error the worst case is the $1000rpm$ test where the error is over 20%, shown in Fig. 6.8. Although it is seen to correct eventually as the runtime of the test is quite long.

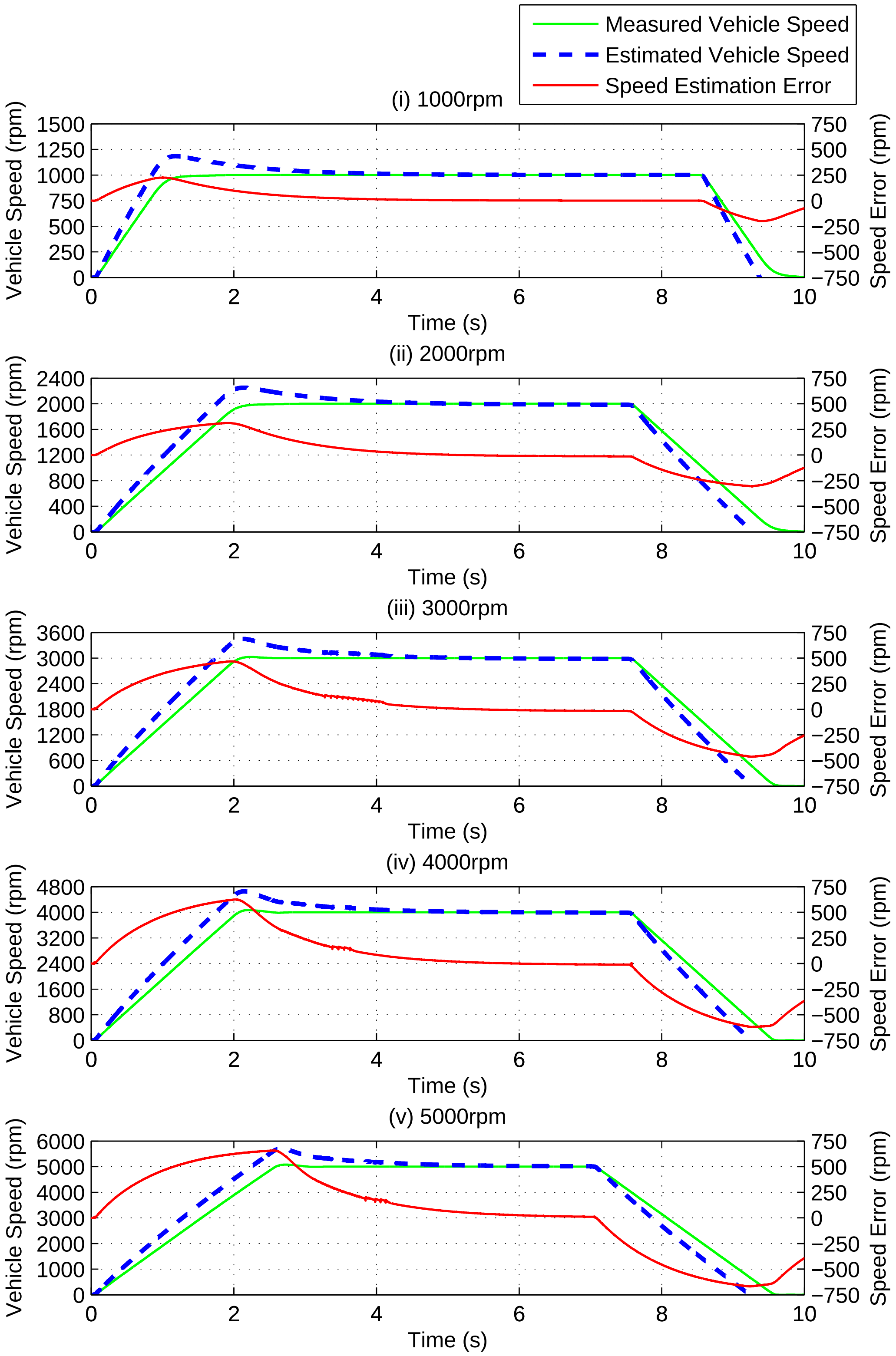

Figure 6.8: imation with $200kg$ load and no gradient

If this load is then increased to $500kg$ the estimation accuracy becomes even worse during acceleration, shown in Fig. 6.9. In some cases over 1000*rpm* error and only just being corrected before the deceleration stage. In the 1000*rpm* test the speed estimation error is over 50%.

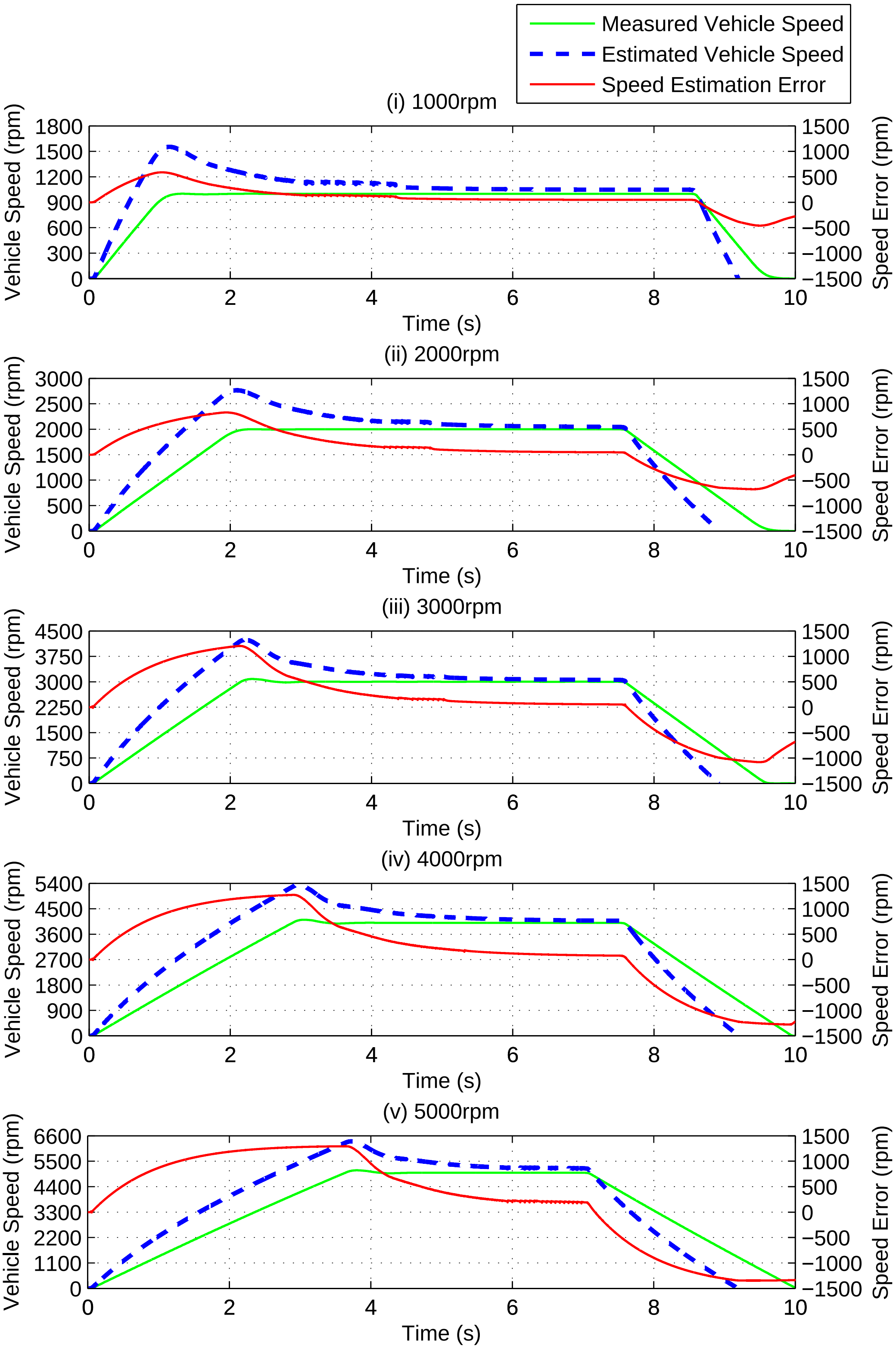

Figure 6.9: Simulation results of speed estimation with 500$kg$ load and no gradient

In the next test a gradient of 5 degrees is encountered after driving 5 metres

(1000*rpm* trace), shown in Fig. 6.10. Longer distances of 10, 15 and 20 meters are used for higher speed traces, so that the gradient is not encountered whilst still accelerating and the impact of each seen separately. This causes there to be a fixed error in the speed estimation which never decreases. The speed estimator is more strongly affected by changes in load torque such as stiction and gradients, with mass only being a major issue during acceleration.

In the 1000*rpm* trace (i) the gradient starts after 3.38 seconds, in the 2000*rpm* trace (ii) it is 3.87 seconds, in the 3000*rpm* trace (iii) it is 2.91 seconds, in the 4000*rpm* trace (iv) 3.14 seconds and in the 5000*rpm* trace (v) the gradient starts after 3.53 seconds.

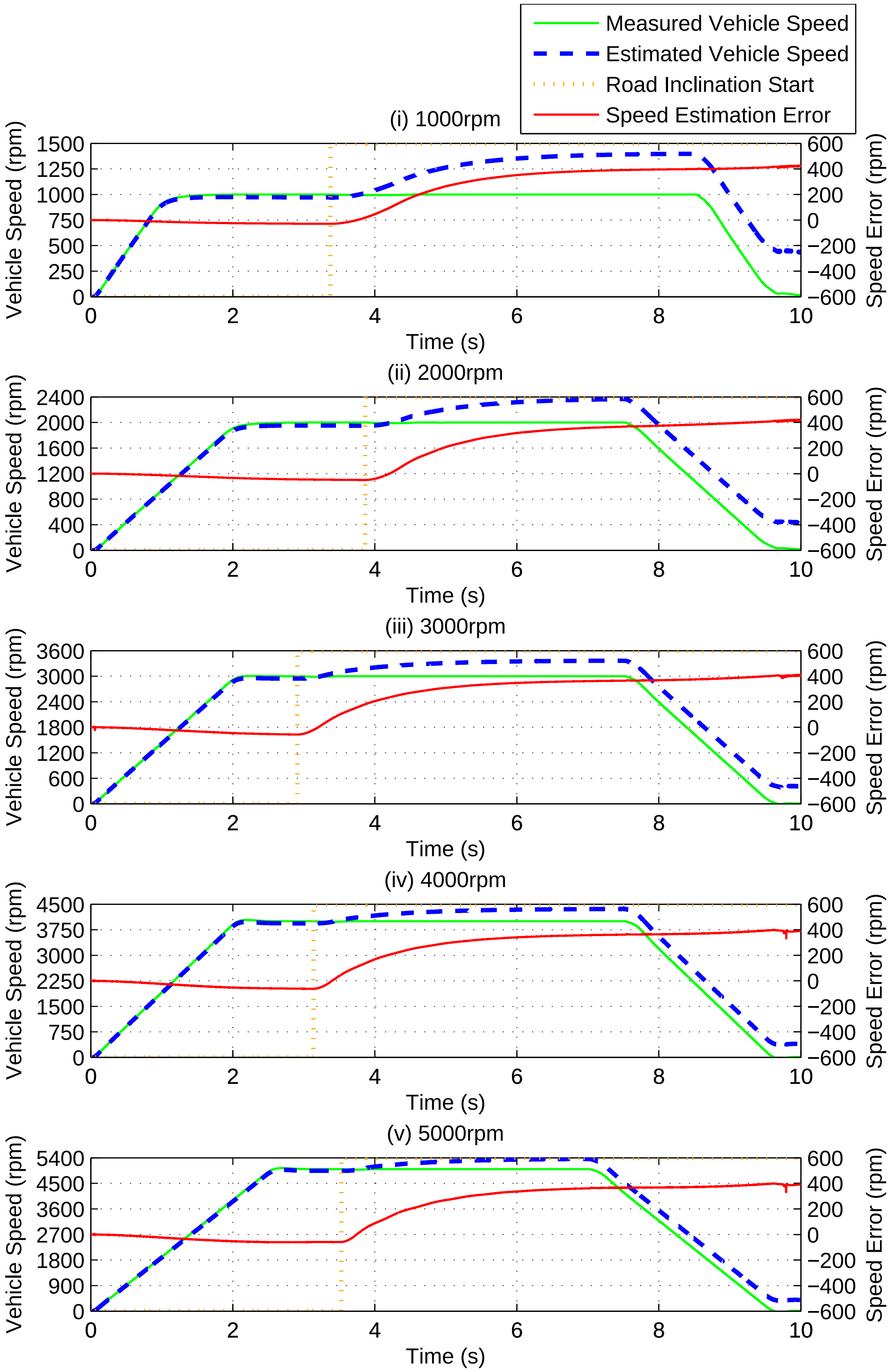

Figure 6.10: Simulation results of speed estimation with no load mass and 5 degree gradient after $5m$ (i), $10m$ (ii & iii), $15m$ (iv) and $20m$ (v)

### 6.2.8 Experimental Results

The speed estimator has been implemented on the test vehicle and the results of this are shown in the following figures in this section. When all the vehicle parameters are known it is possible to accurately estimate the vehicle speed. A number of tests where driving forwards and then carrying out a direction change were performed at increasing speeds: starting at $200rpm$ shown in Fig. 6.11, $500rpm$ shown in Fig. 6.12 and $1000rpm$ shown in Fig. 6.13. A $1500rpm$ test was done without a direction change shown in Fig. 6.14 and one in torque control mode shown in Fig. 6.15 up to around $2000rpm$.

All of the results (without mass changes or gradients) show the vehicle speed estimate matching the measured value quite accurately with considerable reduction in noise, mainly due to the quality of wheel speed sensor used causing a large amount of measurement noise. The estimated speed is seen to typically have only a few percent error from the actual speed.

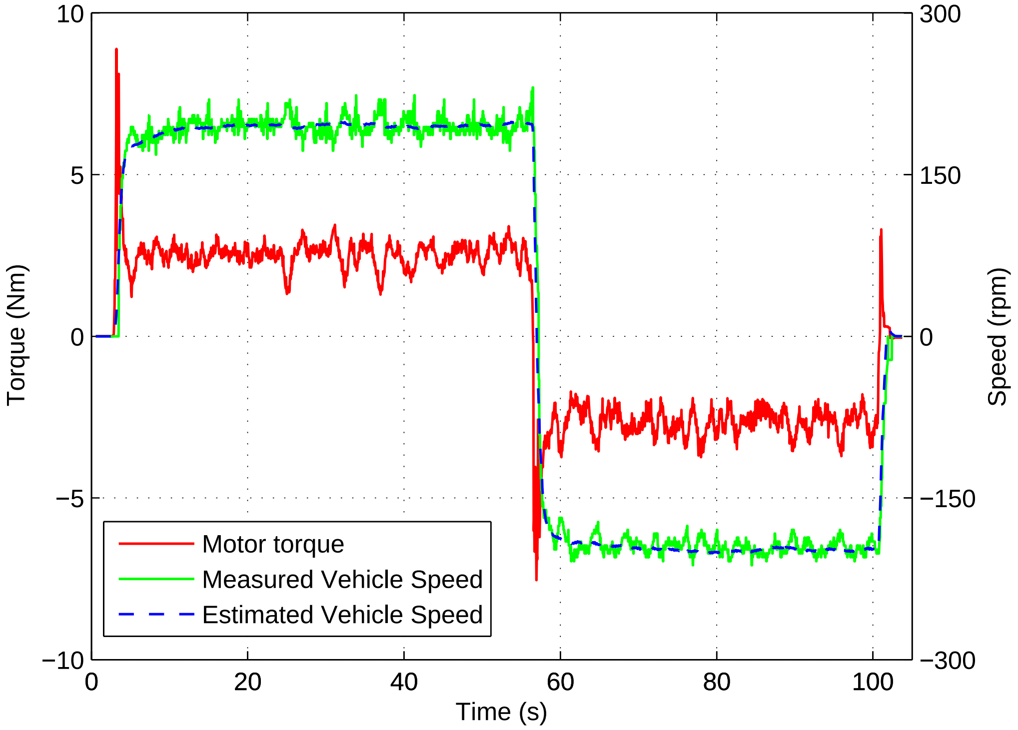

Figure 6.11: Experimental results of speed estimator at $200rpm$ in speed mode

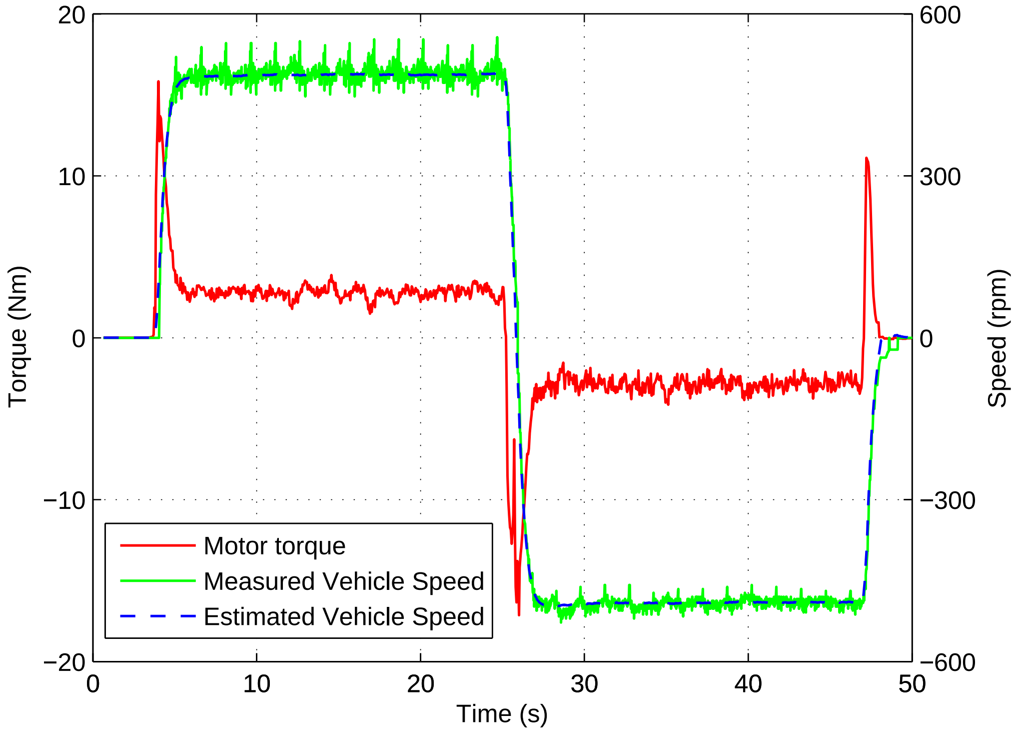

Figure 6.12: Experimental results of speed estimator at 500$rpm$ in speed mode

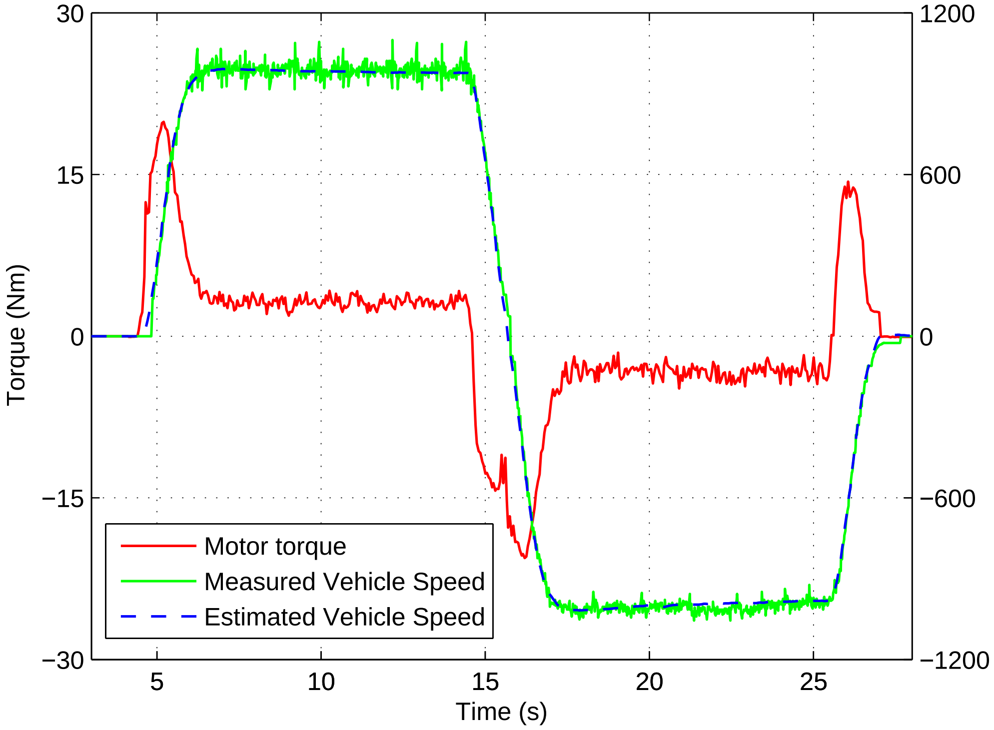

Figure 6.13: Experimental results of speed estimator at 1000$rpm$ in speed mode

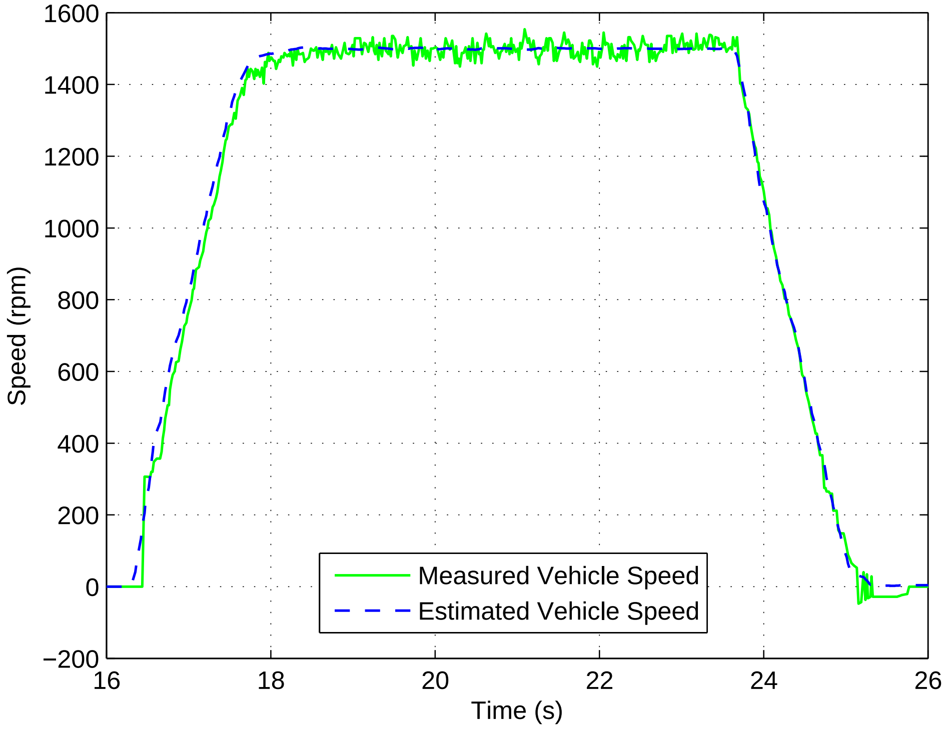

Figure 6.14: Experimental results of speed estimator at 1500$rpm$ in speed mode

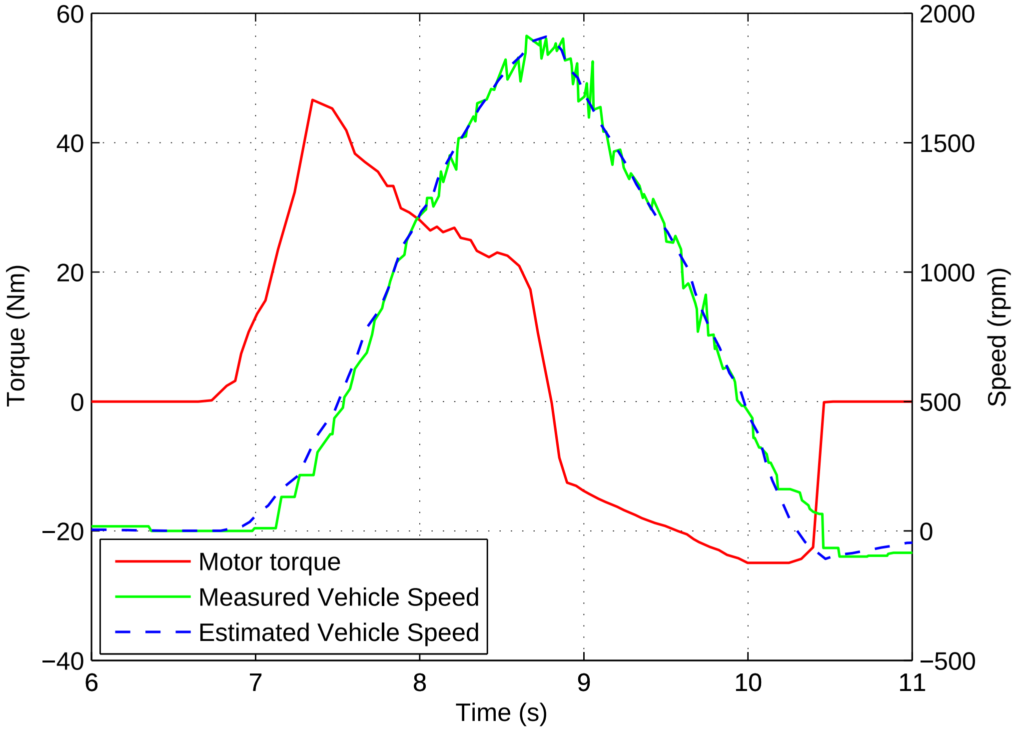

Figure 6.15: Experimental results of speed estimator in torque mode

The results above are all done under ideal conditions where a lot of assumptions are made for parameters and driving conditions remaining constant. It is common for industrial vehicles to carry or tow large heavy loads: these can be greater than the unloaded mass of the vehicle and therefore this change has a significant impact on the vehicle's response. As there is no feedback from the vehicle side of the system, significant errors in vehicle mass will cause the estimated vehicle speed to drift away from the actual. Feedback is unable to correct this during acceleration, shown in Fig. 6.16 and Fig. 6.17, for experimental results at 1000$rpm$ and 1500$rpm$.

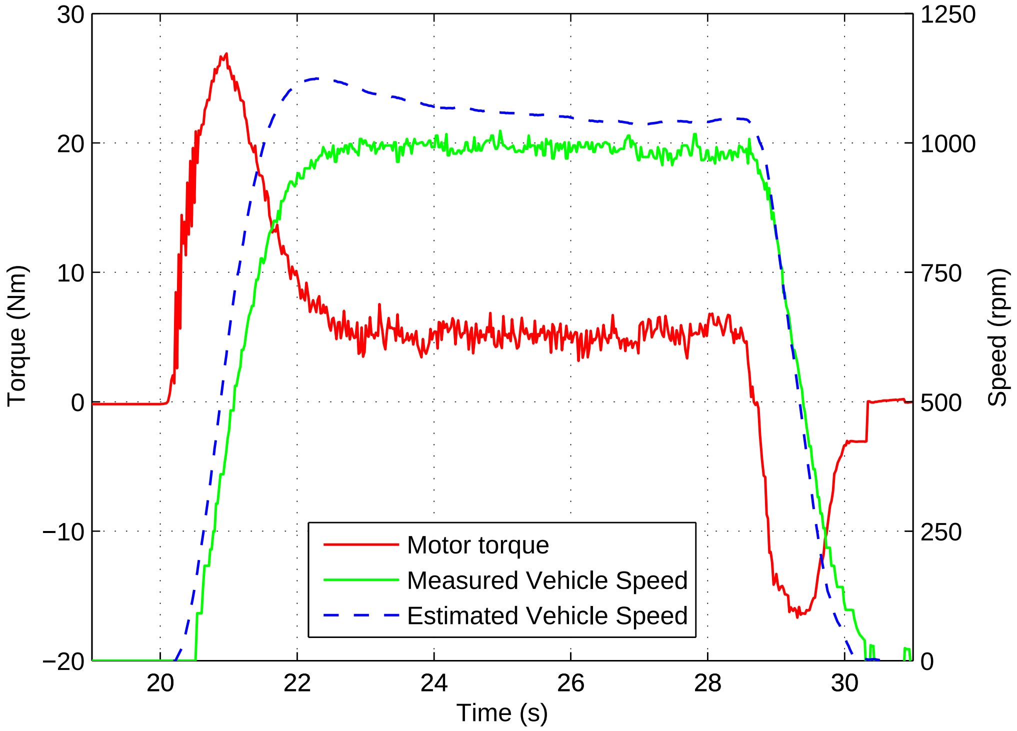

Figure 6.16: Experimental results of speed estimation at 1000$rpm$ with 200$kg$ mass increase in speed mode

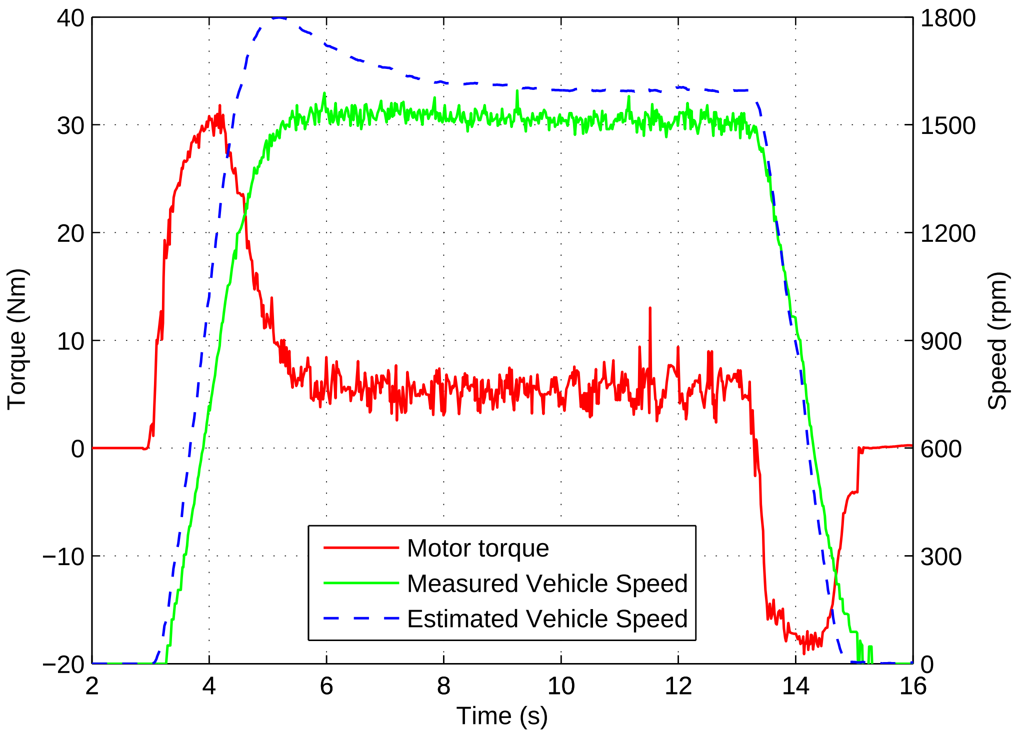

Figure 6.17: Experimental results of speed estimation at 1500$rpm$ with 200$kg$ mass increase in speed mode

### 6.2.9 Discussion

It has been shown that it is possible to estimate vehicle speed, but it has also been shown to not give very accurate results under changing vehicle driving conditions. Both mass changes and gradients lead to significant errors that would make the estimator unusable in its current form.

The results assume that the vehicle parameters are not time varying whereas variables such as tyre rolling resistance and the vehicle mass are likely to change when the vehicle is used, with changing loads and tyre temperature. The variable with the most significant effect is vehicle mass and it is considered in the following sections 6.3, 6.4 and 6.5 of this chapter. The mass changes also cause the stiction part of rolling resistance to increase with mass increase, a steady state error is shown in Fig. 6.16 and Fig. 6.17.

For the estimated vehicle speed to be useful as a replacement for the measured motor speed, it needs to have a very small error. Ideally the estimate should only be a couple of percent difference from the actual vehicle speed. Any significant error with the estimate (greater than 5%) would defeat the purpose of estimating vehicle speed.

## 6.3 Considering Mass Changes

### 6.3.1 When the Vehicle Mass Changes

The results in sections 6.2.7 and 6.2.8; Figs. 6.16 and 6.17 show that change in vehicle mass can deteriorate the accuracy of the vehicle speed estimation. For the test vehicle used and other passenger vehicles, the mass changes are likely to be quite small when compared to the overall mass of the vehicle. For industrial vehicles though it is quite common for vehicles such as tow tractors or mining haulers to carry loads up to and greater than the vehicle unloaded mass. This large unknown change to the vehicle model needs to be accounted for.

The Extended Kalman Filter is often used for estimation for time varying non-linear systems, it does however require a large amount of computational time and in this work high computational efficiency is required. A simpler method has been proposed using an Recursive Least Squares (RLS) parameter estimator to track the errors in mass caused by the vehicle being loaded. As mass can only change when the vehicle is stationary, only a single estimate of mass is required as soon as possible after the vehicle has started to move.

### 6.3.2 Mass Estimation Model

The entire vehicle dynamics have now been simplified in (6.41), $J_{total}$ is the entire system inertia, including the vehicle mass and drivetrain inertias, all referenced to the motor, refer to single mass model in section 5.3.3. The inertia is then converted to equivalent vehicle mass using (6.42), after subtraction the known axle and drivetrain inertia. As stiction is dependent on the vehicle mass, on every iteration of the Recursive Least Squares (RLS) algorithm the stiction term is updated, depending on the previous mass estimate (6.43). This allows both stiction and mass changes to be found whilst only having one parameter being estimated, meaning all vectors and matrices in the algorithm are $1 \times 1$. It does however rely on the stiction remaining proportional to mass across the full vehicle operating load and speed range.

$$T_m - T_{stiction} - \omega_m . b_{total} = \left( {}^{1}/_{J_{total}} \right) \frac{d\omega_m}{dt} \tag{6.41}$$

$$m_{total} = m_v + m_{load} = (J_{total} - J_{axle}) \left( {}^{n^2}/_{r^2} \right) \tag{6.42}$$

$$T_{stiction} = (m_v + m_{load}) . g . C_0 . ( {}^{r}/_{n} ) \tag{6.43}$$

The mass estimation model uses a greater number of assumptions than the previous drivetrain model, such as using motor speed feedback to give vehicle acceleration. This is an issue due to the noise content of the signal and the fact that motor speed is not equal to vehicle speed during transients, especially the case when the vehicle starts to move. This has been solved through using a pre-filter on both the input signals (motor torque and acceleration) to the RLS, a $10Hz$ second order Butterworth low pass filter which removes most of the noise [20]. The delay of the pre-filter is tolerable as the mass estimate is not used continually in real time to update the Kalman Filter (KF). Only a single value of mass is selected and used based on the following criteria: the rate of change of the mass estimate is less than $10kg$ for each iteration of the RLS for 20 consecutive samples. This selects the mass value when the estimate has levelled off and reached the correct value.

Due to the relative size of the RLS parameters, the inputs and the outputs of the estimator, 9.23 fixed point scaling is used for the RLS as this will maximise the accuracy. Care needs to be taken to ensure that the maximum possible number size of 256 is not exceeded.

In order to ensure that there is sufficient acceleration and torque applied to compare, at low torques and accelerations the algorithm is disabled, small values cause the estimate to to go out of range causing the fixed point number to wrap. It is tuned to to not respond too strongly to noise as is required to run during transients and as the mass is not changing it does not need to track a changing value.

### 6.3.3 Recursive Least Squares Theory

The Recursive Least Squares (RLS) algorithm used is shown in equations (6.44) to (6.46) [91] it operates at a sample rate of $5ms$. It is obvious that the algorithm shares a lot with the KF in section 6.2.2. The first step is calculate the correction vector **K** (6.44), following this the parameter estimate **Θ** can be updated (6.45) and finally the error covariance **P** (6.46). For this system all vectors are single values due to only one parameter to be estimated.

A fixed forgetting factor **R** of 0.98 is used (set quite high as the algorithm runs during transients when the measurements are noisy), **F** is the input and in this case motor torque, **Θ** is the fixed parameter to be estimated and is equal to $1/J_{total}$ and **Y** is the measured feedback of motor acceleration.

$$\mathbf{K_k} = \mathbf{P_{k-1}}\mathbf{F_k}^T(\mathbf{F_k}\mathbf{P_{k-1}}\mathbf{F_k}^T + \mathbf{R})^{-1} \tag{6.44}$$

$$\mathbf{\Theta_k} = \mathbf{\Theta_{k-1}} + \mathbf{K_k}(\mathbf{Y_k} - \mathbf{F_k}\mathbf{\Theta_{k-1}}) \tag{6.45}$$

$$\mathbf{P_k} = \mathbf{R}^{-1}(\mathbf{I} - \mathbf{K_k}\mathbf{F_k})\mathbf{P_{k-1}} \tag{6.46}$$

As mass should not change when the vehicle is moving, a single estimate of mass is required as soon as possible after the vehicle has started to move. In order to ensure this can happen quickly the RLS is initialised just before the estimator is enabled with the torque input and a calculated acceleration based upon the unloaded mass. This was found to reduce the time for a mass estimate to be found.

The overall scheme is given in Fig. 6.18, note that the RLS inputs are pre-filtered and that the mass estimate is used to compensate the torque input.

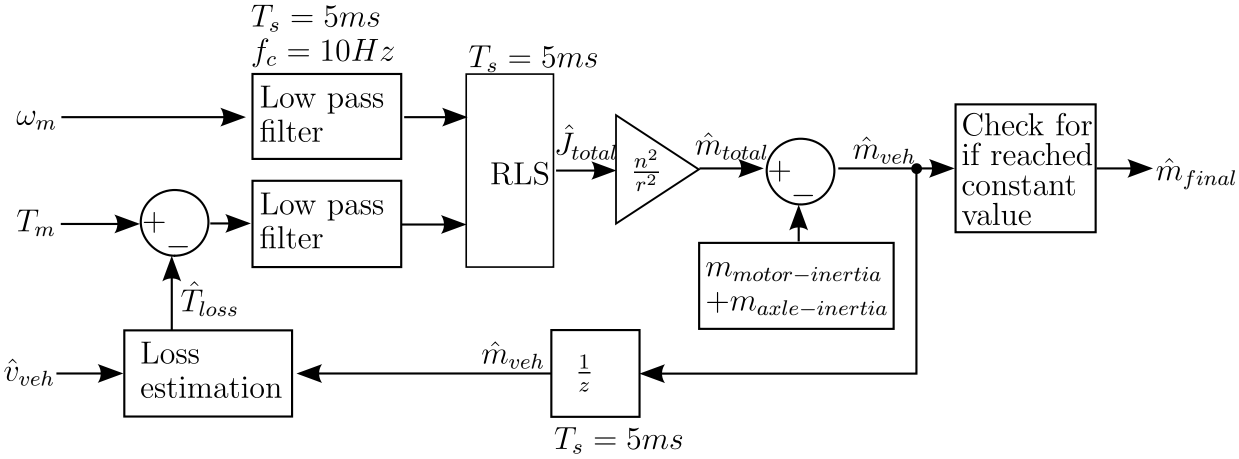

Figure 6.18: Recursive Least Squares estimation method

### 6.3.4 Recursive Least Squares Results

The RLS scheme was implemented on the test vehicle and the results are discussed below. The majority of the tests were done in speed mode, so there can be seen to be a large torque and acceleration initially followed by constant speed driving and then deceleration to a stop. In Fig. 6.19 the two inputs to the RLS scheme are shown in the first plot; the differential of the motor speed can be seen to be very noisy, this is after the pre-filter. The mass estimate though is found to be fairly flat after the initial transient due to the very high forgetting factor of 0.98.

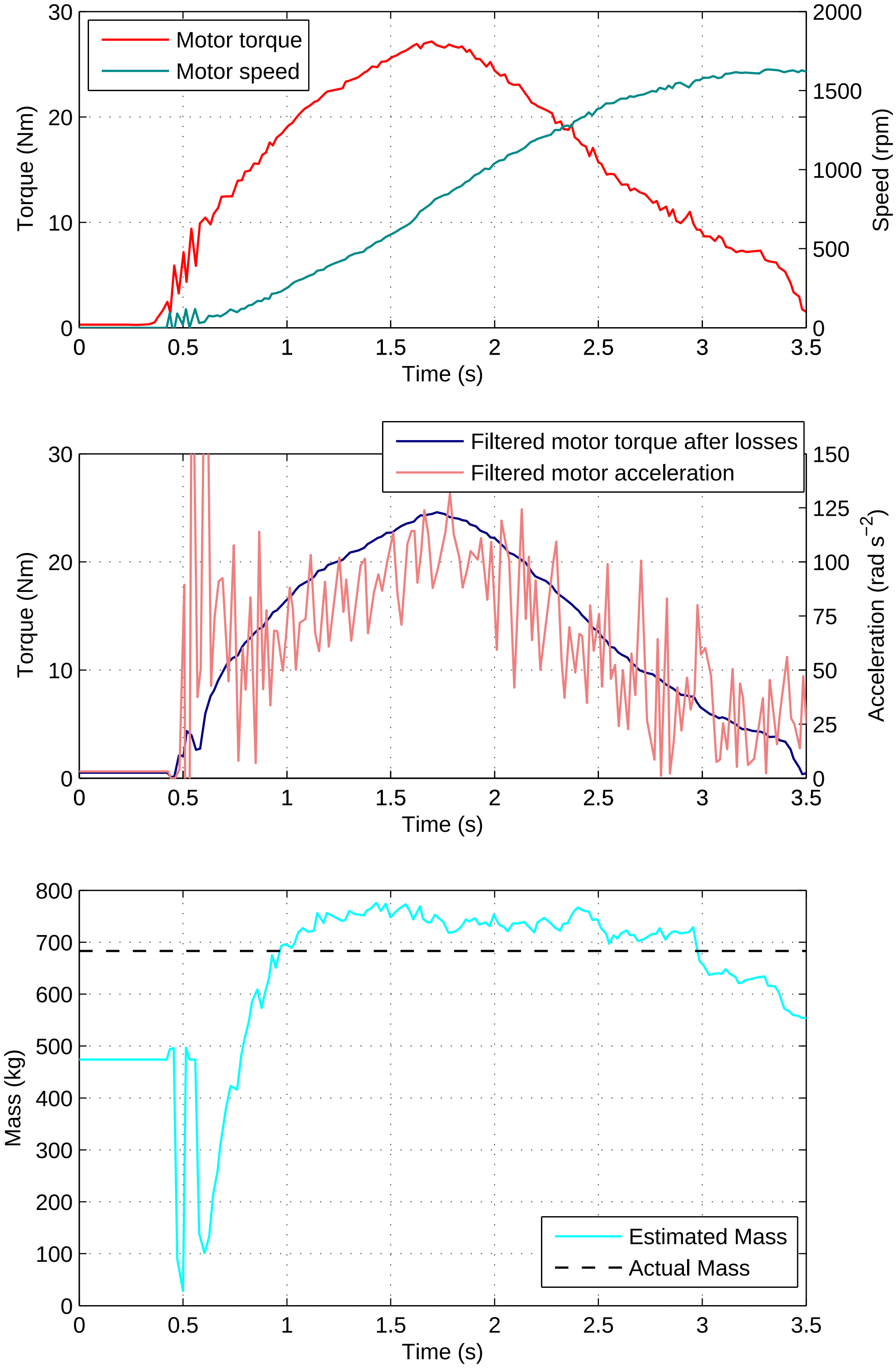

Figure 6.19: Experimental results of Recursive Least Squares mass estimate

It is shown in Fig. 6.20 that it takes around 1.5 seconds for the new vehicle mass to be determined. In this case a value of $730kg$ is obtained: vehicle $483kg$, load $200kg$ and estimation error $47kg$, the axle and motor inertia equivalent of $71kg$ has already been removed from this figure. After the vehicle has stopped and remained stationary for a specified time (2 seconds), the mass estimate can be reset to the unloaded value, as it can no longer be assumed that the vehicle is still loaded. The delay to find the mass was found to be too long and it can clearly be seen to flatten much earlier than the selected value is chosen. The conditions for selecting a mass value were relaxed to allow an earlier value to be selected.

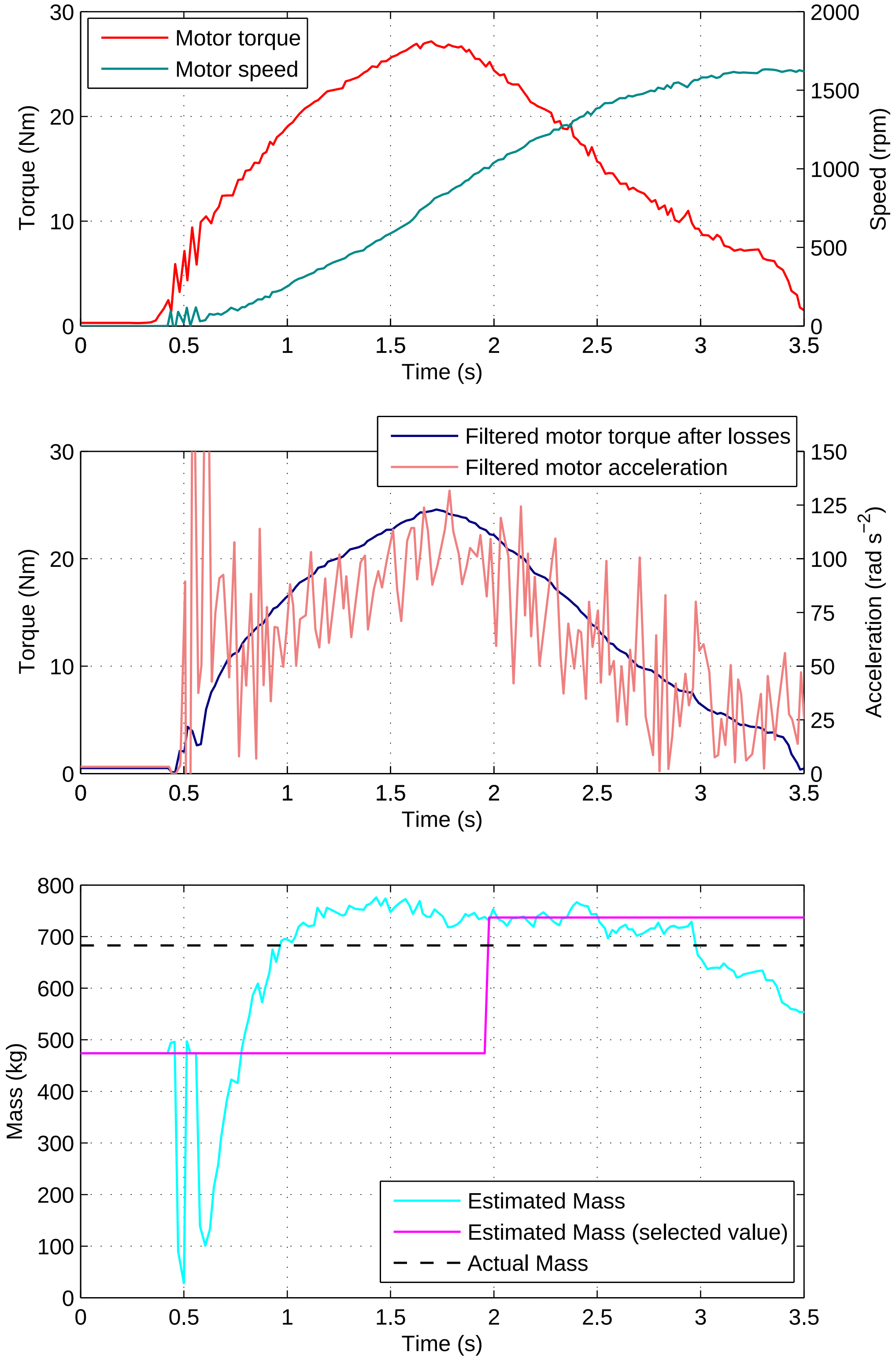

Figure 6.20: Experimental results of Recursive Least Squares with a mass value selected

It is important that the mass estimate is found quickly during the initial acceleration, during a speed mode application if the vehicle is accelerating to a fixed speed another opportunity will not occur for some time, although due to limited driving space all of the results here show very short start and stop responses. In Fig. 6.21 the mass is shown to be found fairly quickly but in a comparable driving trace shown in Fig. 6.22, the mass is not found until after the vehicle stops accelerating giving a much larger speed estimation error. This is due to oscillations in the torque demand being transferred into the mass estimate, so the selection criteria is not met. It is important then that the speed loop is tuned well, although in this example it is closed around the noisy and oscillatory measured motor speed.

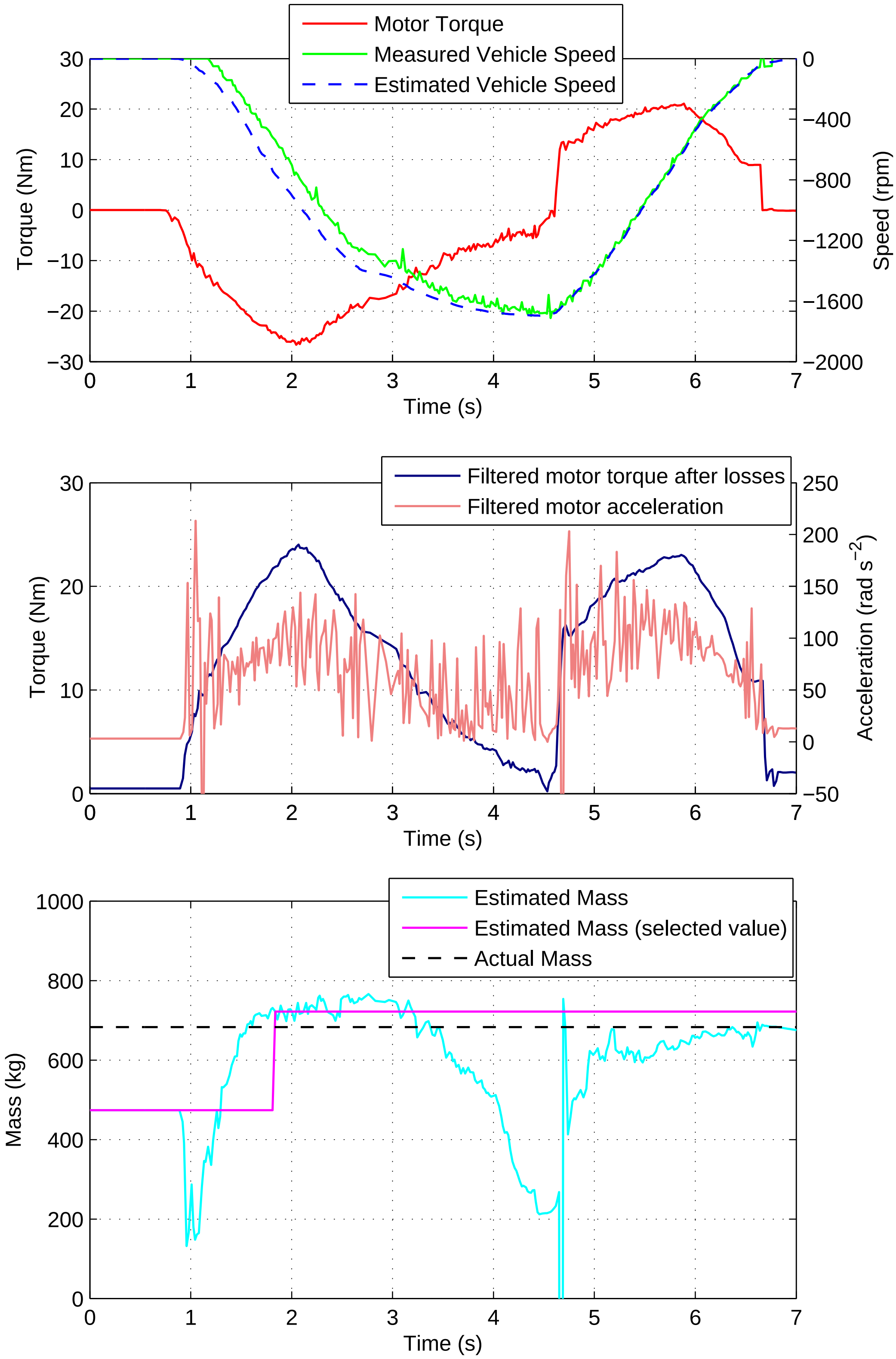

Figure 6.21: Experimental results of Recursive Least Squares with a selected mass value quickly determined

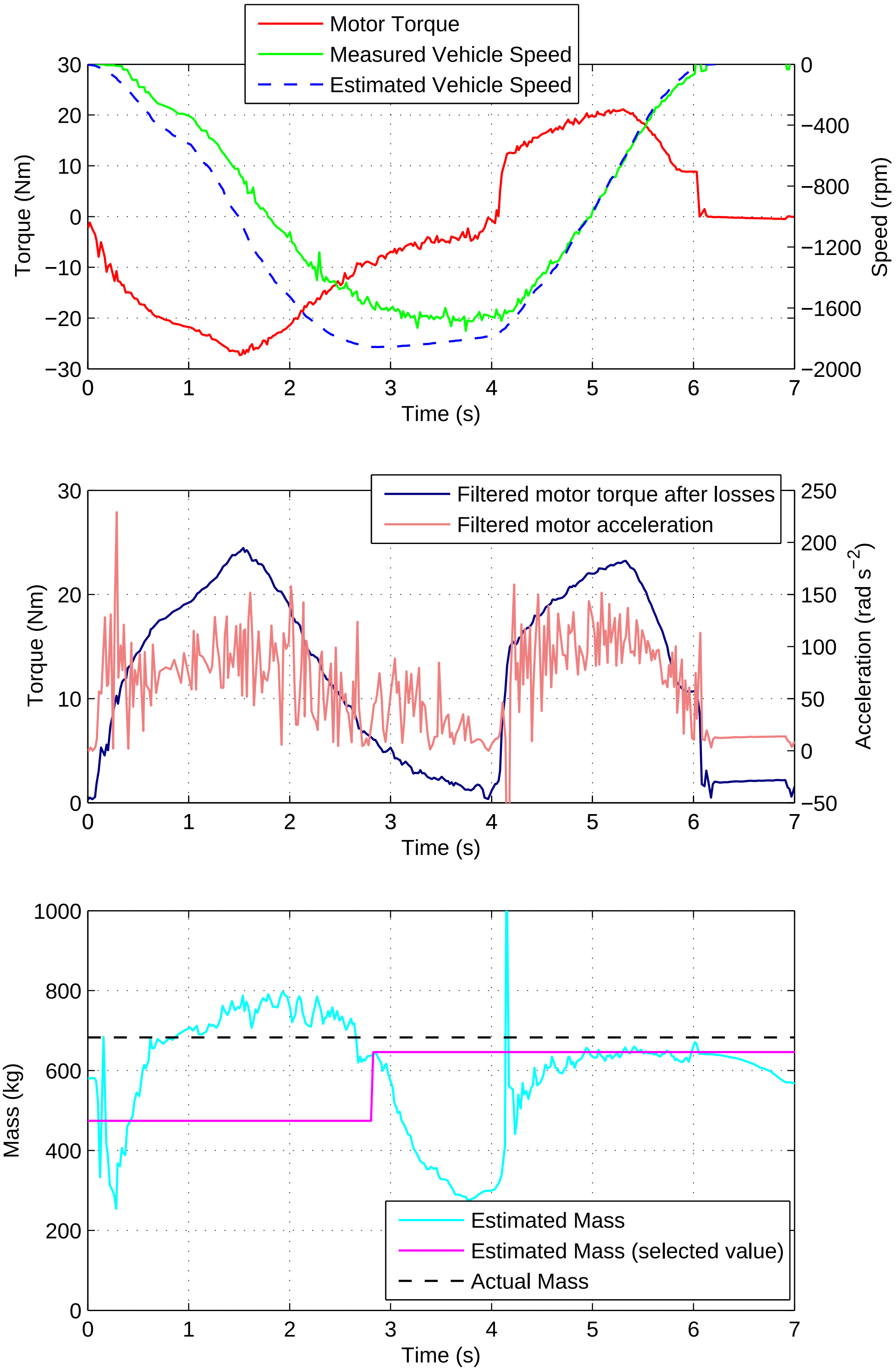

Figure 6.22: Experimental results of Recursive Least Squares with a selected mass value not determined quickly

In Fig. 6.23 the mass estimate is shown to go from $-10,000kg$ to $8000kg$ within $100ms$. Large noise spikes in the measured acceleration saturate the 256 limit of the 9.23 scaling, the estimator is seen to recover very quickly and a valid mass estimate is still found.

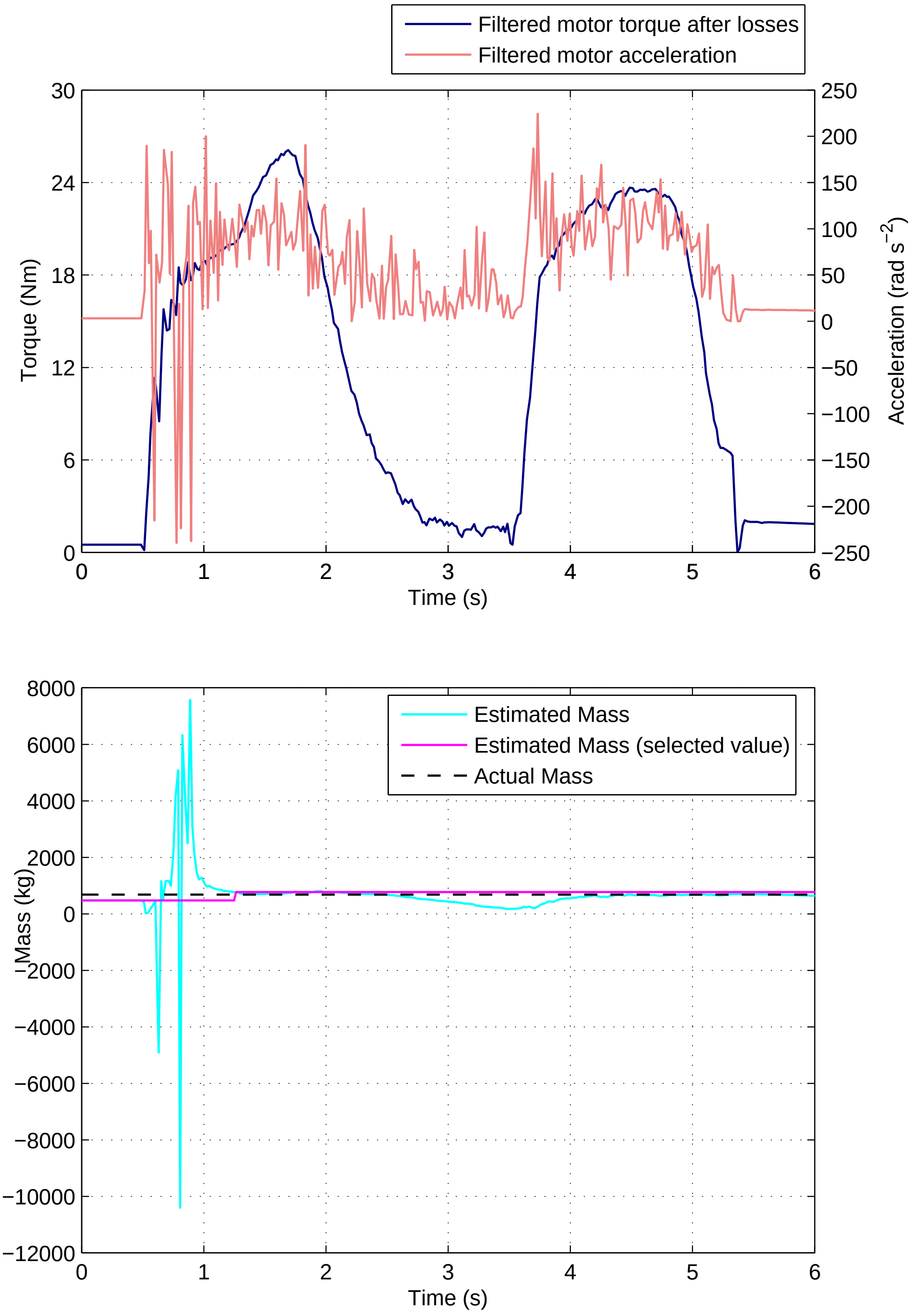

Figure 6.23: Experimental results of Recursive Least Squares with a large glitch in mass estimate

As the losses (both stiction and friction) use fixed coefficients any errors here will cause the acceleration and brake torque to acceleration ratio to be wrong. In Fig. 6.24 below there is a clear difference shown between the mass estimate during acceleration and braking; the first part of the trace is accelerating to 1500$rpm$ and the second half (3 seconds onwards) deceleration. If the losses were corrected here it would reduce this error.

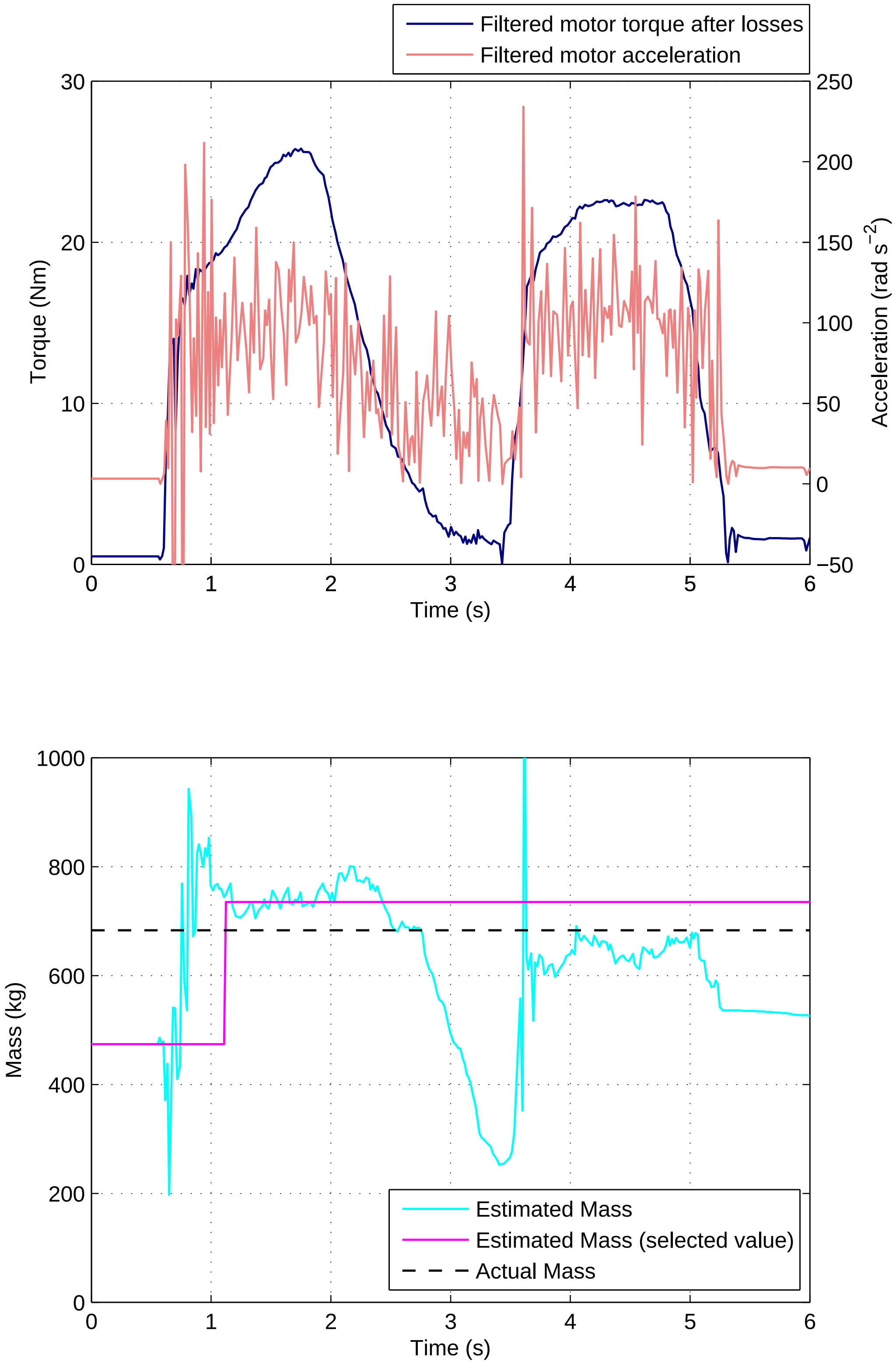

Figure 6.24: Experimental results of Recursive Least Squares with a different mass estimate when accelerating and braking

It can be seen in the above figures that at low acceleration and torque conditions the mass estimation performance is poor, due to the errors being larger than the useful information. The mass estimator is switched off at very low torque and acceleration levels to avoid an incorrect mass value being found during these conditions.

Due to the large amount of noise in the acceleration signal, improved mass estimation could possibly be achieved measuring the acceleration directly using the accelerometer. This would add an additional cost to the vehicle and the above results show that the estimated mass signal is almost free from noise.

### 6.3.5 Jacobian Matrices

With knowledge of the estimated vehicle mass, the speed estimator now needs to be corrected; although up until the mass estimate is found, the unloaded (incorrect) vehicle mass has to be used. For this system, a change in mass causes a change in the discrete matrices ($\mathbf{A_{d-co}}$, $\mathbf{A_{d-bl}}$, $\mathbf{B_{d-co}}$ and $\mathbf{K_{d-co}}$). The change in these matrices is inversely proportional (almost linearly) to the change in mass, allowing them to be quickly recalculated. The Jacobian matrices determining the change with mass: $\mathbf{J_{A_{d-co}}}$ (6.47), $\mathbf{J_{A_{d-bl}}}$ (6.48), $\mathbf{J_{B_d}}$ (6.49) and $\mathbf{J_{K_{d-co}}}$ (6.50), are calculated off-line, so that just one multiplication is required for each entry in the matrices when the new mass is obtained.

The Jacobian matrices define how the state matrices change with a given state or parameter, typically for all the states within the state space model. In this case though due to the linear nature of the model, only the differential of the state space matrices with vehicle mass is required. Normally the continuous model is partially differentiated to calculate the Jacobian, but as the discrete form is required for this work. It is determined by calculation.

$$\mathbf{J_{A_{d-co}}} = \frac{\delta \mathbf{A_{d-co}}}{\delta\left(1/m_{load}\right)} = \frac{\mathbf{A_{d-co-full-load}} - \mathbf{A_{d-co-no-load}}}{1/(m_{full-load} - m_{no-load})} \tag{6.47}$$

$$\mathbf{J_{A_{d-bl}}} = \frac{\delta \mathbf{A_{d-bl}}}{\delta\left(1/m_{load}\right)} = \frac{\mathbf{A_{d-bl-full-load}} - \mathbf{A_{d-bl-no-load}}}{1/(m_{full-load} - m_{no-load})} \tag{6.48}$$

$$\mathbf{J_{B_d}} = \frac{\delta \mathbf{B_d}}{\delta\left(1/m_{load}\right)} = \frac{\mathbf{B_{d-full-load}} - \mathbf{B_{d-no-load}}}{1/(m_{full-load} - m_{no-load})} \tag{6.49}$$

$$\mathbf{J_{K_{d-co}}} = \frac{\delta \mathbf{K_{d-co}}}{\delta\left(1/m_{load}\right)} = \frac{\mathbf{K_{d-co-full-load}} - \mathbf{K_{d-co-no-load}}}{1/(m_{full-load} - m_{no-load})} \tag{6.50}$$

The mass correction (Jacobian) matrices are found by simply calculating two versions of each matrix; one for the unloaded mass and one for maximum loaded vehicle mass (in this case an additional $250kg$, approximately 50% of the unloaded vehicle mass). The difference of the two (loaded - unloaded) divided by the inverse of the mass change is then calculated. The accuracy of using this method has been tested and graphed below as it assumes linearity to the inverse of mass, shown in Figs. 6.25, 6.26, 6.27 and 6.28.

For the speed estimator matrix $\mathbf{A_{d-co}}$, mass correction calculation error shown in Fig. 6.25, for most load mass values the mass correction had a negligible effect on their accuracy. The main effect was noticed for rows 3 and 4 and columns 3 and 4; the matrix entries mainly controlling the ratio of vehicle states to other vehicle states, as the mass has the greatest impact on these. Although across the range that was calculated, up to around $750kg$, the error was minimal and it was only mass changes above this that lead to a reduced accuracy of 5%. For the matrix $\mathbf{A_{d-bl}}$, shown in Fig. 6.26, none of the values had a considerable error after being corrected by the mass. For the matrix $\mathbf{B_{d-co}}$, shown in Fig. 6.27, all of the second column was affected by mass changes, to a similar amount as the $\mathbf{A_{d-co}}$ matrix above, rows 3 and 4 were also the same. For the Kalman gain matrix $\mathbf{K_{d-co}}$ shown in Fig. 6.28 the first 3 rows had negligible error across the mass range, but the 4th row and 2nd column has a significant error of 50% although outside of the calculated range ($250kg$ load). The 5th row and 2nd column has a very significant error of up to 1000%, but the entries in this row/column are so comparatively small the error is not important.

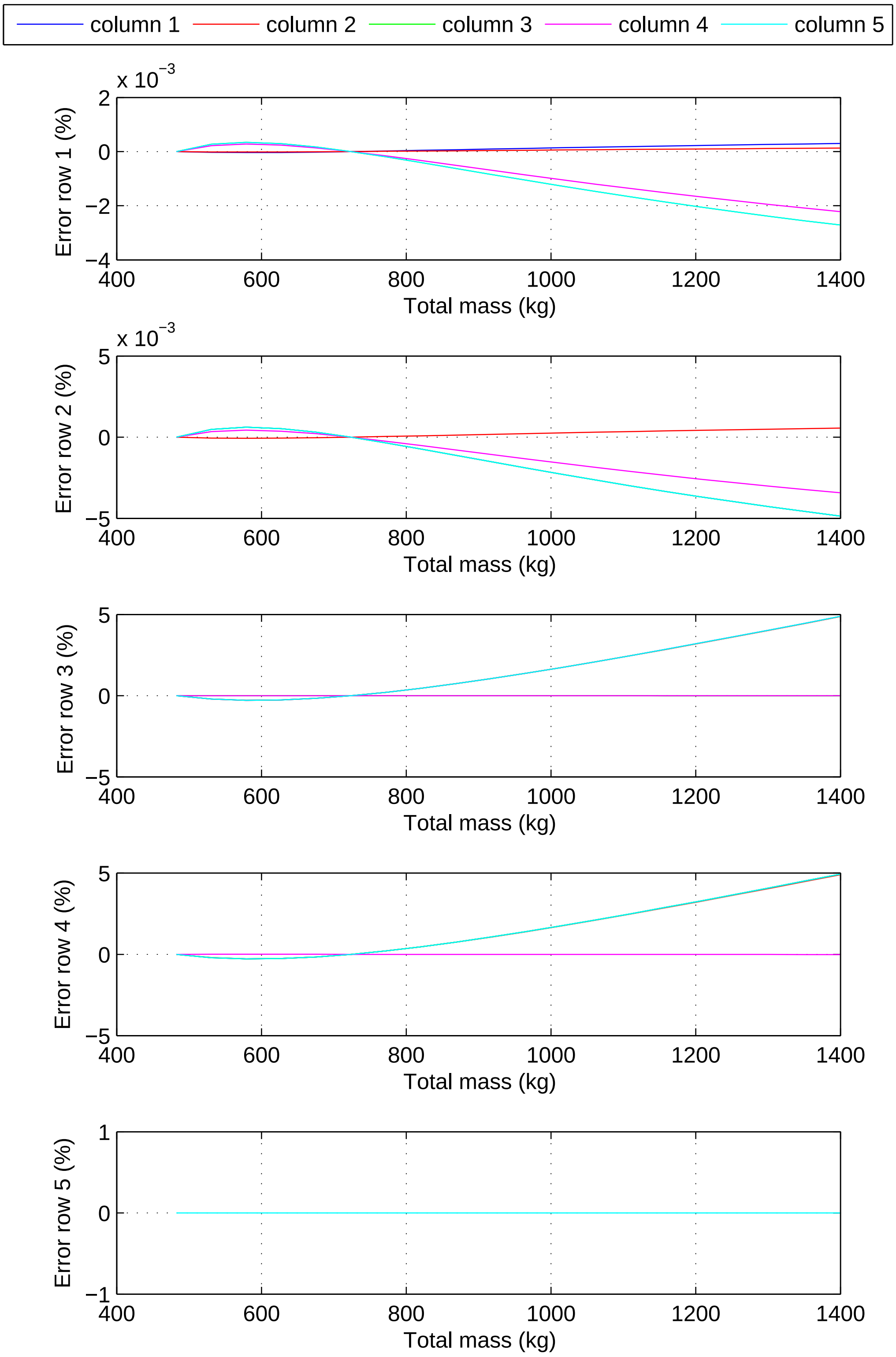

Figure 6.25: Mass correction error for the $\mathbf{A_{d-co}}$ (state transition) matrix

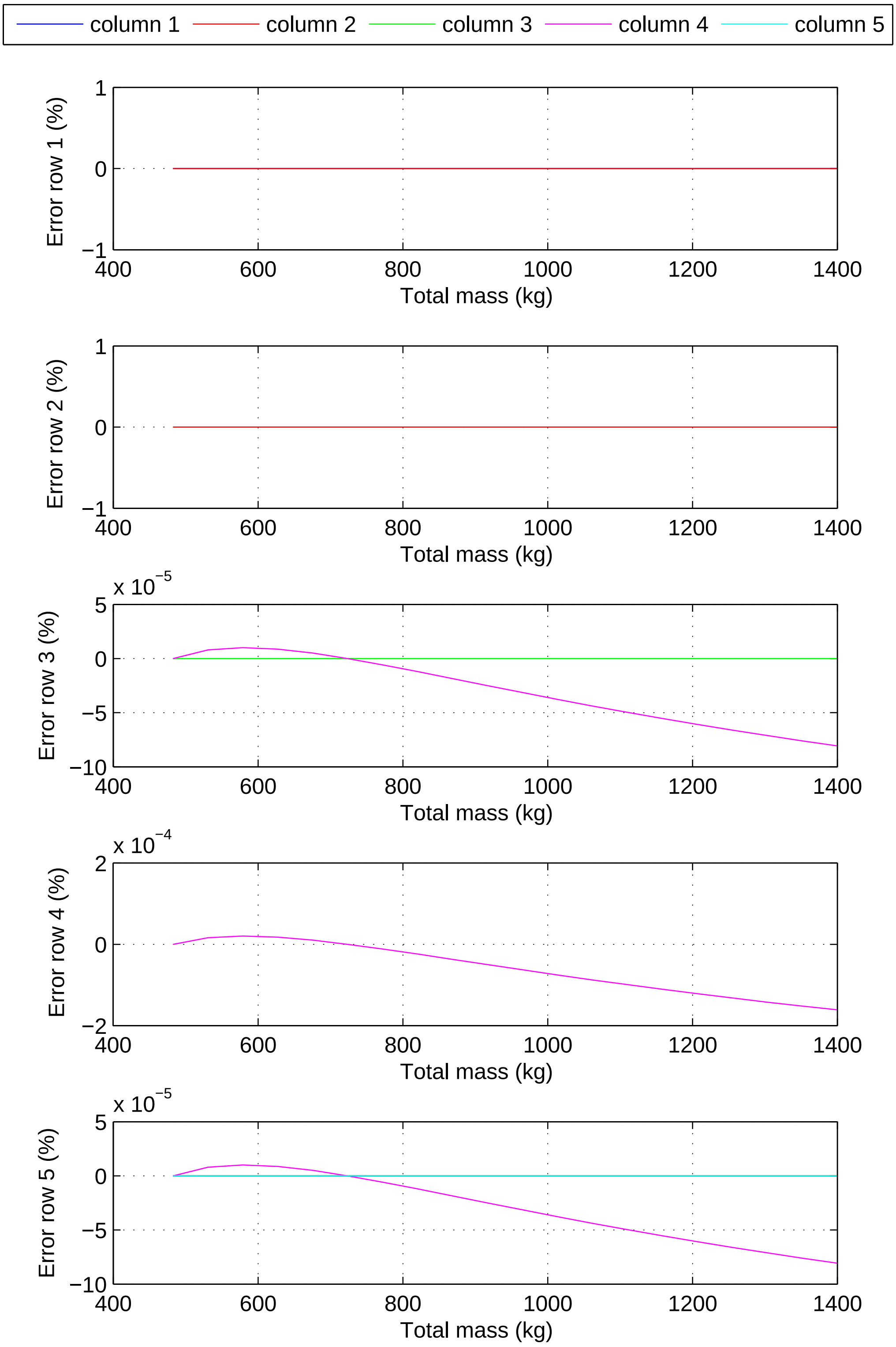

Figure 6.26: Mass correction error for the $\mathbf{A_{d-bl}}$ (state transition) matrix

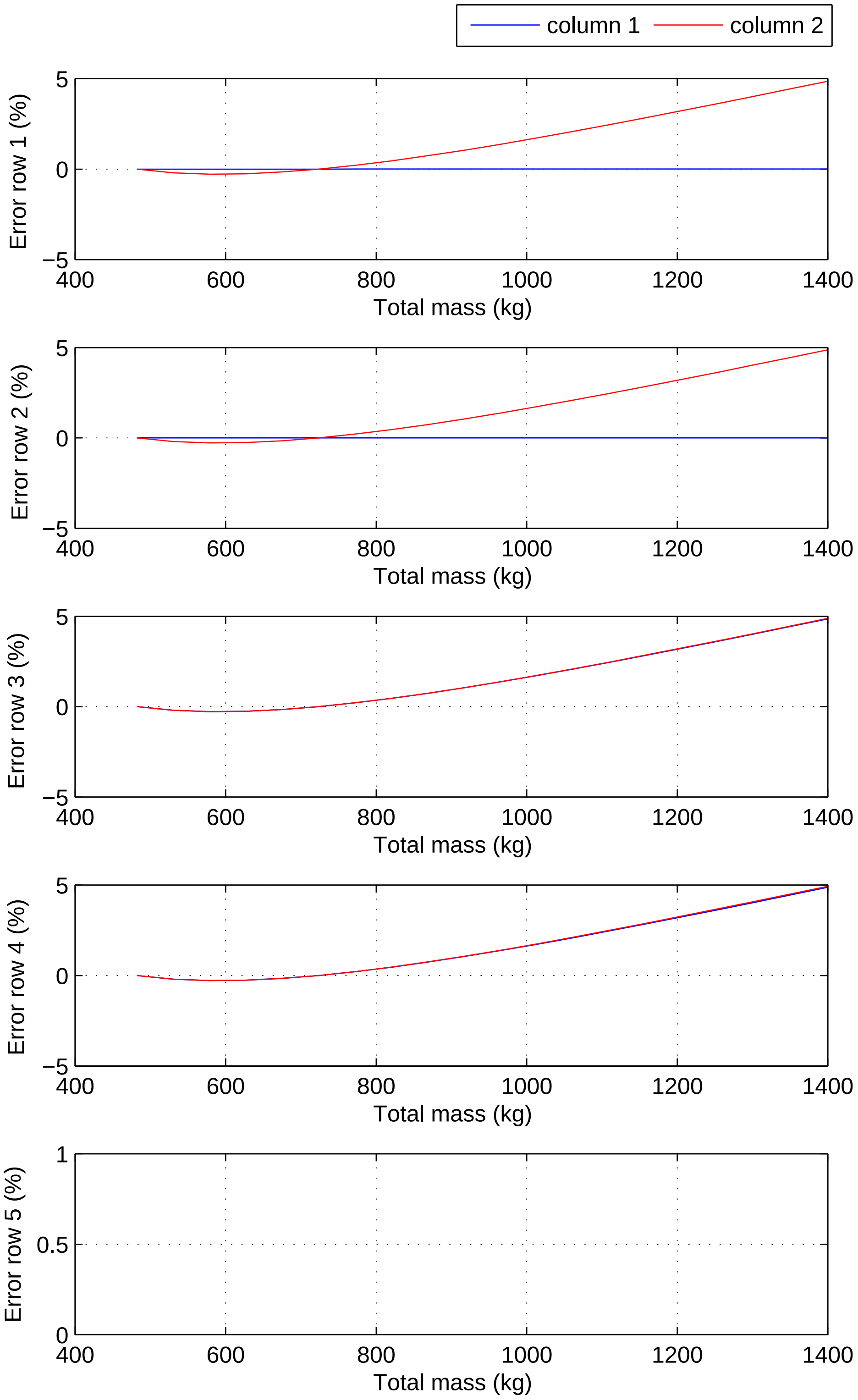

Figure 6.27: Mass correction error for the $\mathbf{B_{d-co}}$ (input) matrix

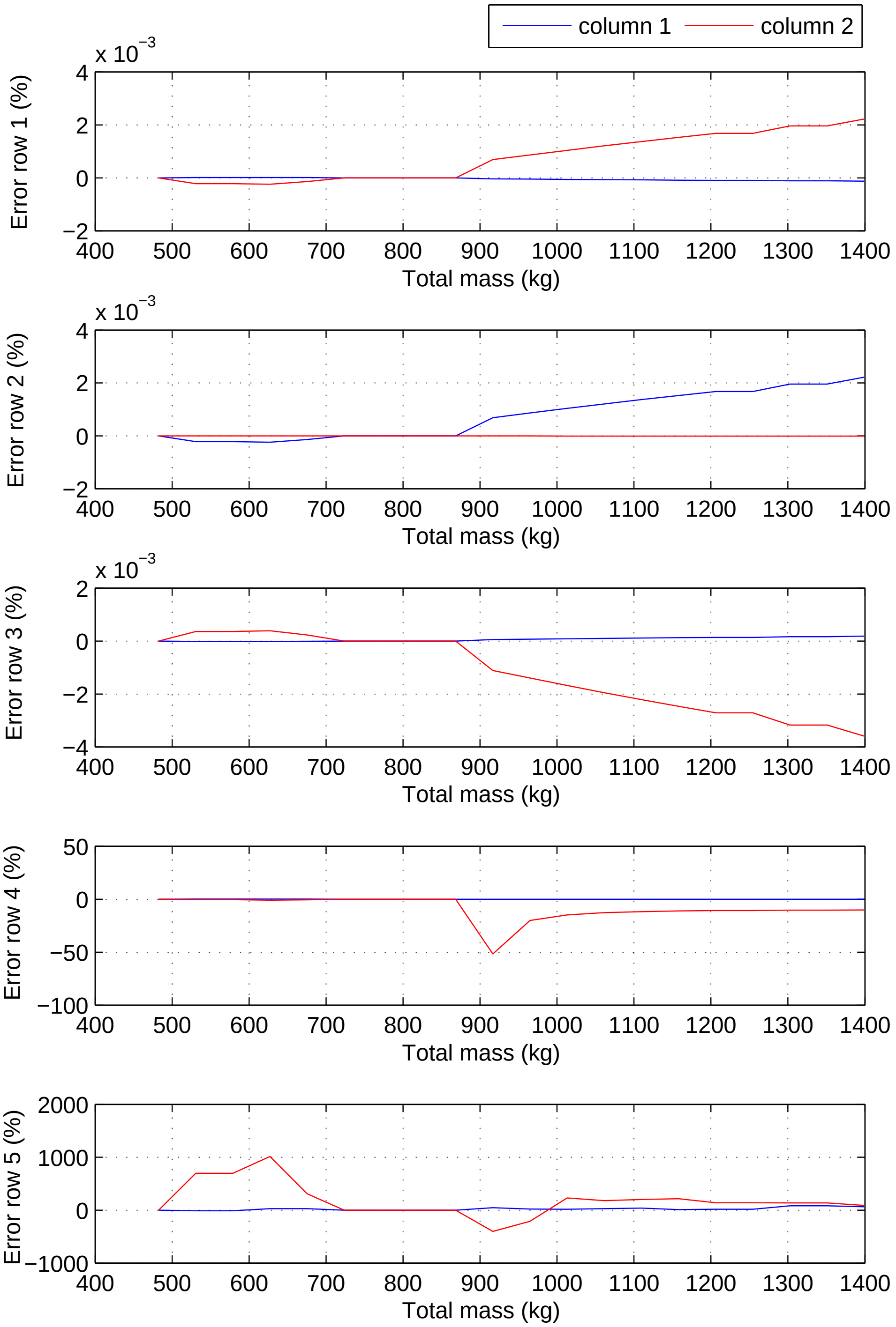

Figure 6.28: Mass correction error for the $\mathbf{K_{d-co}}$ (Kalman gain) matrix

### 6.3.6 Mass Corrected Kalman Filter

In order to correct the discrete state space model with mass changes, the load mass needs to be calculated from the RLS 'final' vehicle mass (6.51). This can then be used along with the Jaaobian matrices to give the mass corrected state space matrices using the following equations: (6.52), (6.53), (6.54) and (6.55).

$$m_{load} = m_{final} - m_{no-load} \tag{6.51}$$

$$\mathbf{A_{d-co}} = \mathbf{A_{d-co-no-load}} + \mathbf{J_{A_{d-co}}} \times (1/m_{load}) \tag{6.52}$$

$$\mathbf{A_{d-bl}} = \mathbf{A_{d-bl-no-load}} + \mathbf{J_{A_{d-bl}}} \times (1/m_{load}) \tag{6.53}$$

$$\mathbf{B_{d}} = \mathbf{B_{d-no-load}} + \mathbf{J_{B_{d}}} \times (1/m_{load}) \tag{6.54}$$

$$\mathbf{K_{d-co}} = \mathbf{K_{d-co-no-load}} + \mathbf{J_{K_{d-co}}} \times (1/m_{load}) \tag{6.55}$$

Figure 6.29 shows the KF and highlights which matrices are corrected with mass, the switches are to indicate the transition between contact and backlash mode. The $\mathbf{B_{d-co}}$ matrix is not switched with backlash, and the contact mode version is always used. There will be a slight difference between its contact and backlash version after it has been converted to its discrete form, it is dependant on the $\mathbf{A_d}$ matrix. This was found to be insignificant and so has been ignored to reduce the required storage in the micro controller. The entire speed and mass estimation algorithm takes around $22\mu s$ to execute on average every $1ms$ iteration.

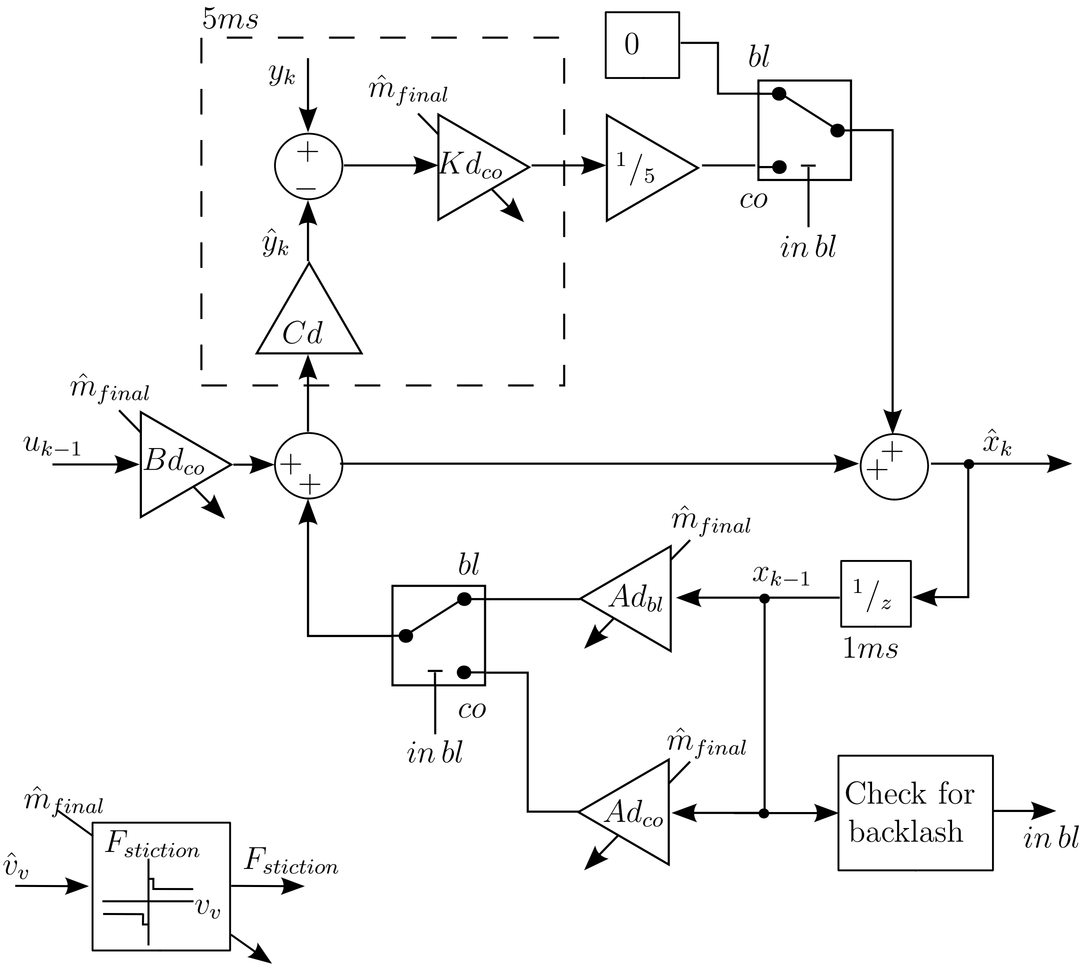

Figure 6.29: Complete Kalman Filter diagram with more than one sample rate, backlash and mass feedback

### 6.3.7 Simulation Results

The results in this section use the same simulated test data as the ones in section 6.2.7 to allow for direct comparison. The unloaded test is actually slightly worse than without the mass estimator, as the RLS mass estimate does not match the actual mass and so the speed estimate is slightly worse, shown in Fig. 6.30. But as typically the mass of a vehicle will change each time it is driven (especially for an industrial one) the benefit is greater than the reduced accuracy in Fig. 6.30.

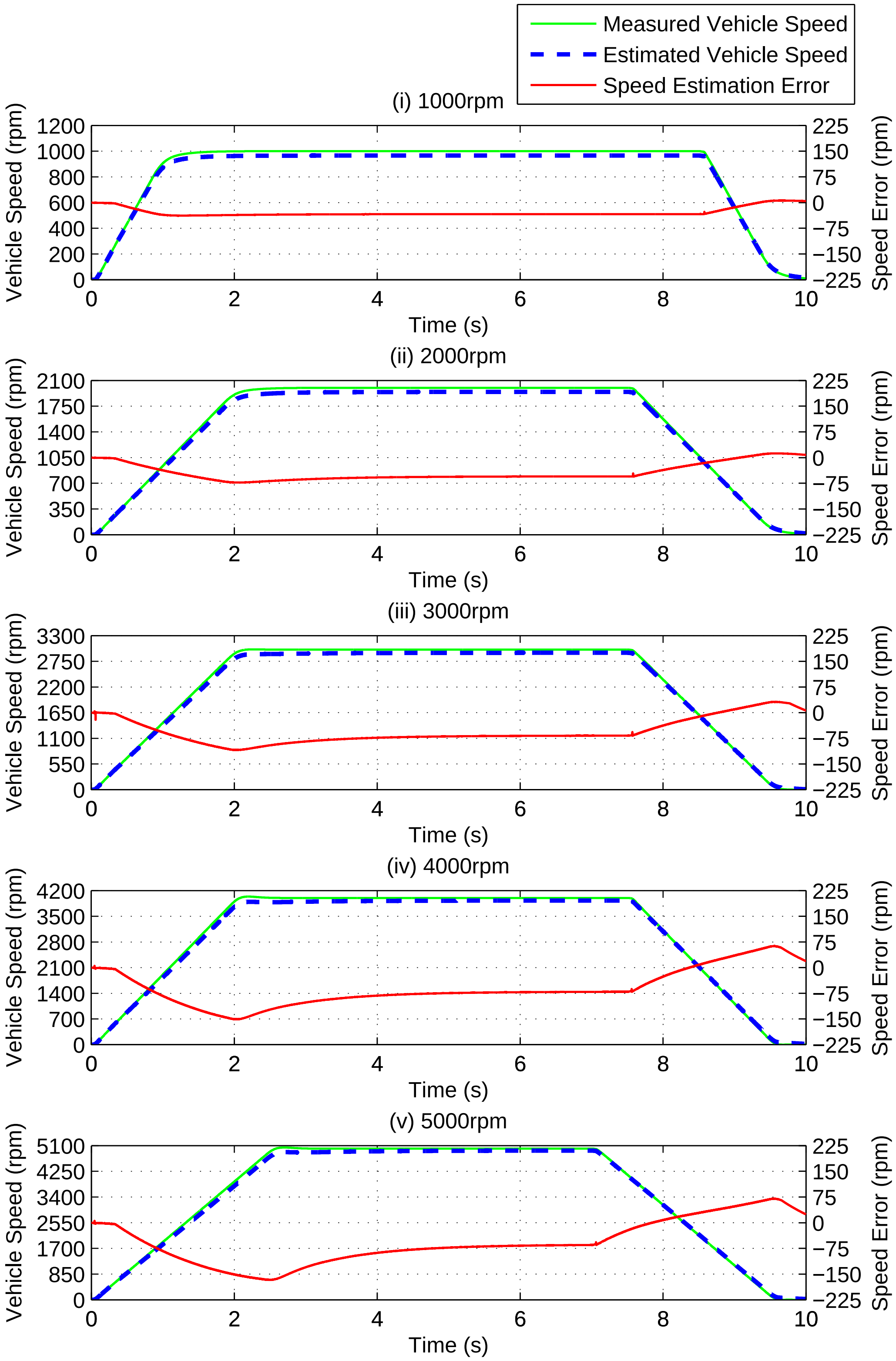

Figure 6.30: Simulation results of speed and mass estimation with no load mass and no gradient

After a typical mass change of $200kg$ the speed estimation initially starts to drift away but is quickly corrected, shown in Fig. 6.31.

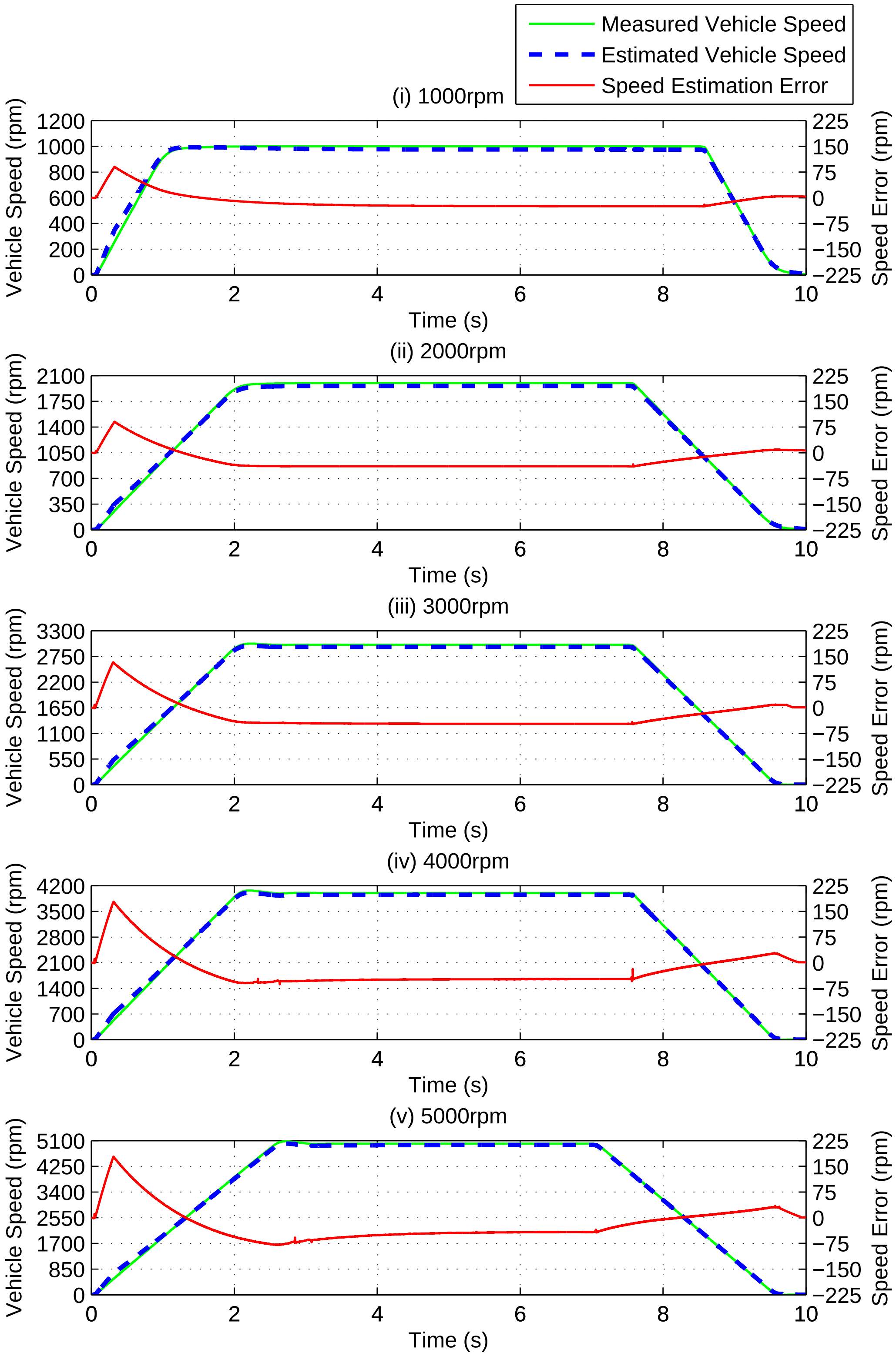

Figure 6.31: Simulation results of speed and mass estimation with 200$kg$ load and no gradient

A higher mass change of $500kg$ as expected gives a larger speed estimation error, but this is still easily corrected, shown in Fig. 6.32.

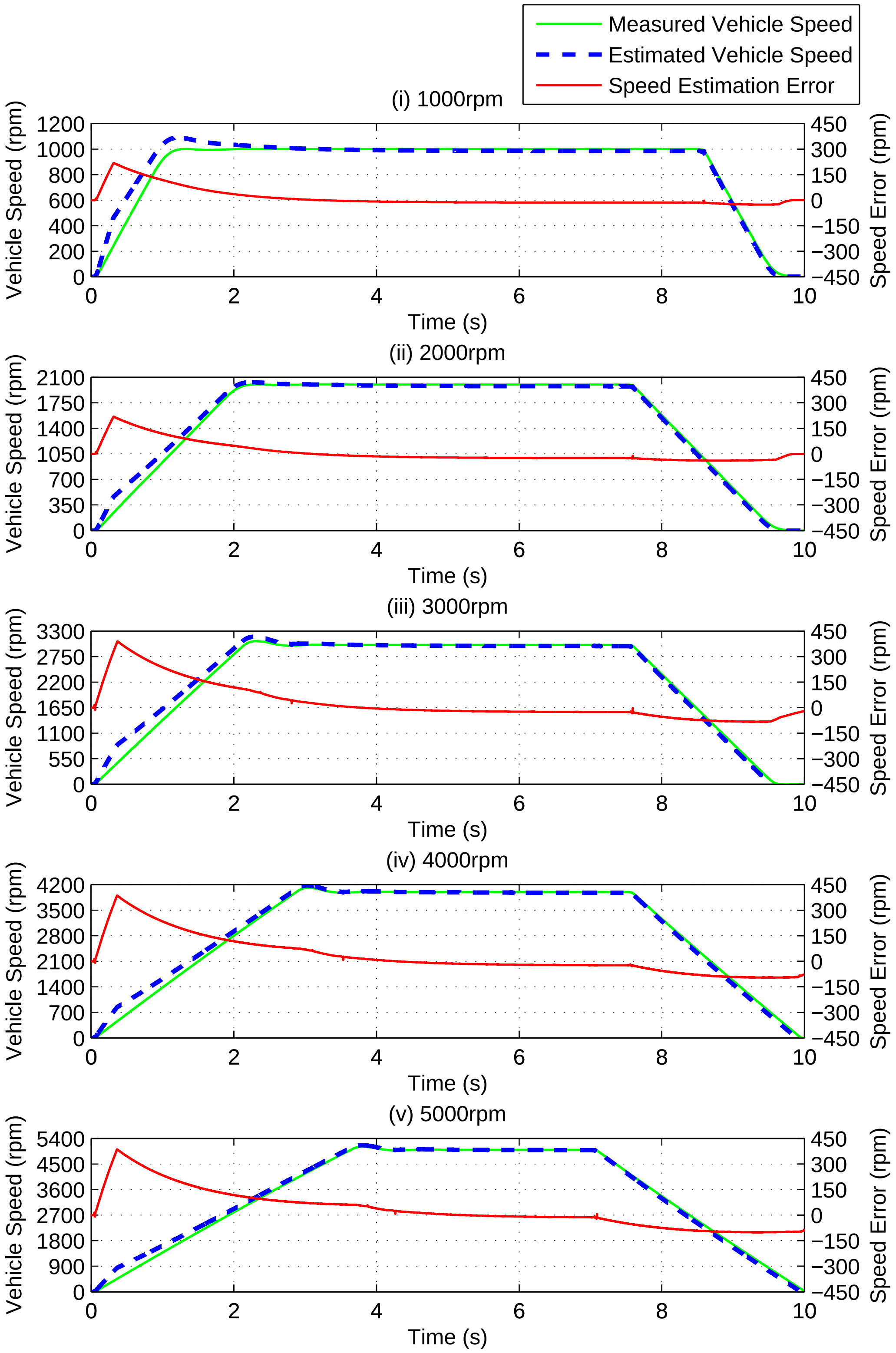

Figure 6.32: Simulation results of speed and mass estimation with $500kg$ load and no gradient

Changes in road gradient however are not compensated for and the results shown in Fig. 6.33 are not improved over Fig. 6.10. Other solutions need to be considered for compensating for non-modelled changes such as gradients.

In the 1000$rpm$ trace (i) the gradient starts after 3.38 seconds, in the 2000$rpm$ trace (ii) it is 3.87 seconds, in the 3000$rpm$ trace (iii) it is 2.91 seconds, in the 4000$rpm$ trace (iv) 3.14 seconds and in the 5000$rpm$ trace (v) the gradient starts after 3.53 seconds.

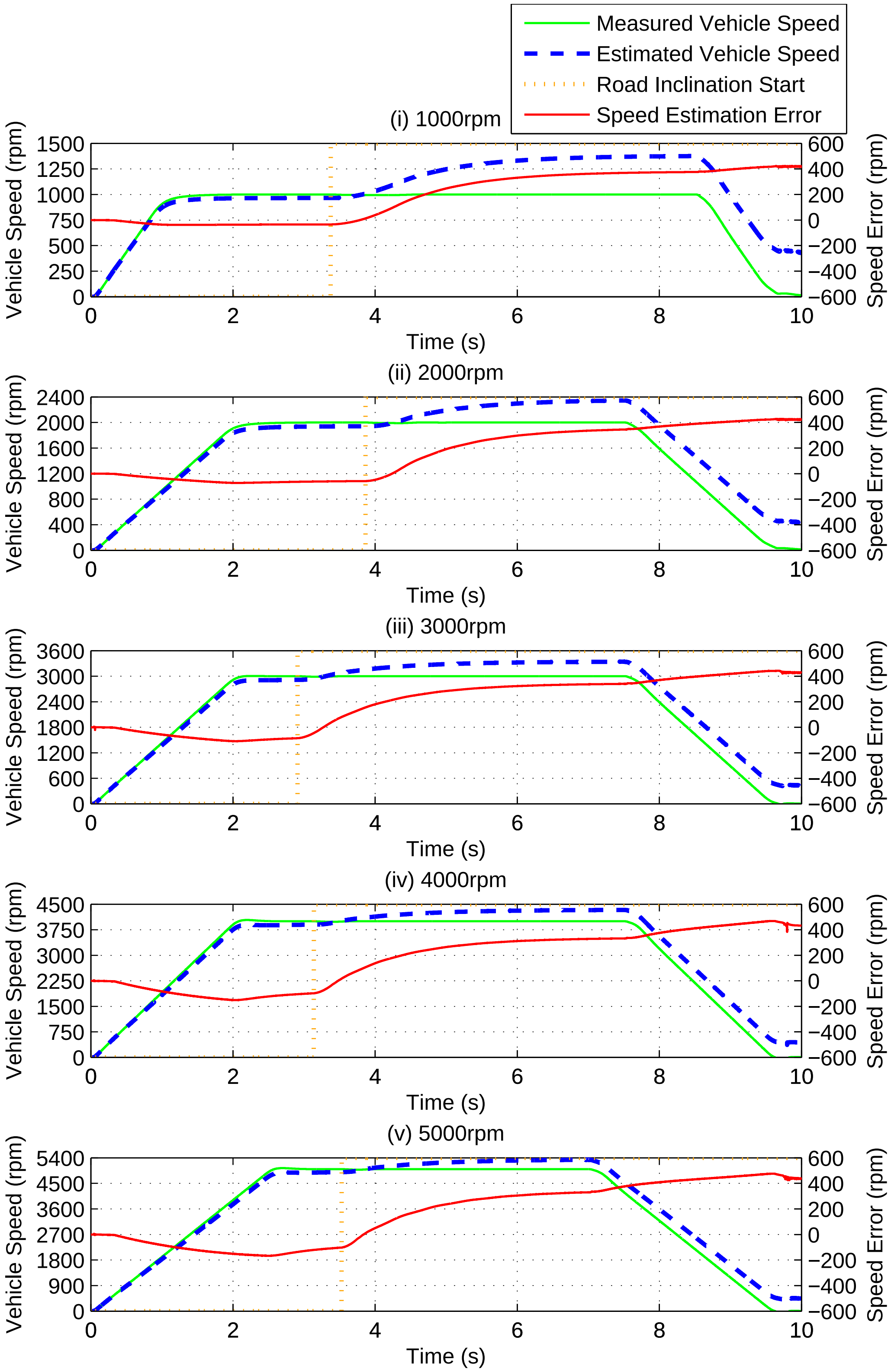

Figure 6.33: Simulation results of speed and mass estimation with no load mass and 5 degree gradient after 5$m$ (i), 10$m$ (ii & iii), 15$m$ (iv) and 20$m$ (v)

### 6.3.8 Experimental Results

In Figs. 6.34 and 6.35 it is shown that the vehicle speed estimation now tracks the measured speed more accurately after an extra load mass of 200$kg$ is added to the vehicle, when compared to the earlier results shown in Figs. 6.16 and 6.17 where there was no mass correction. It takes 0.5 to 1 seconds for the mass to be found which means the speed estimator is corrected before it drifts too far away from the actual. There is a notable change in the acceleration rate of the speed estimate after the mass values has been updated. This can be seen to then have an effect on the speed loop, when speed is closed around the vehicle speed estimate, there is a sudden increase in motor torque demand.

The mass estimate output value (not selected) can be seen to be reasonably noisy which gives a bit of variation to the selected value each time. After the mass estimate flattens off it would be better if the average of the estimated values were used.

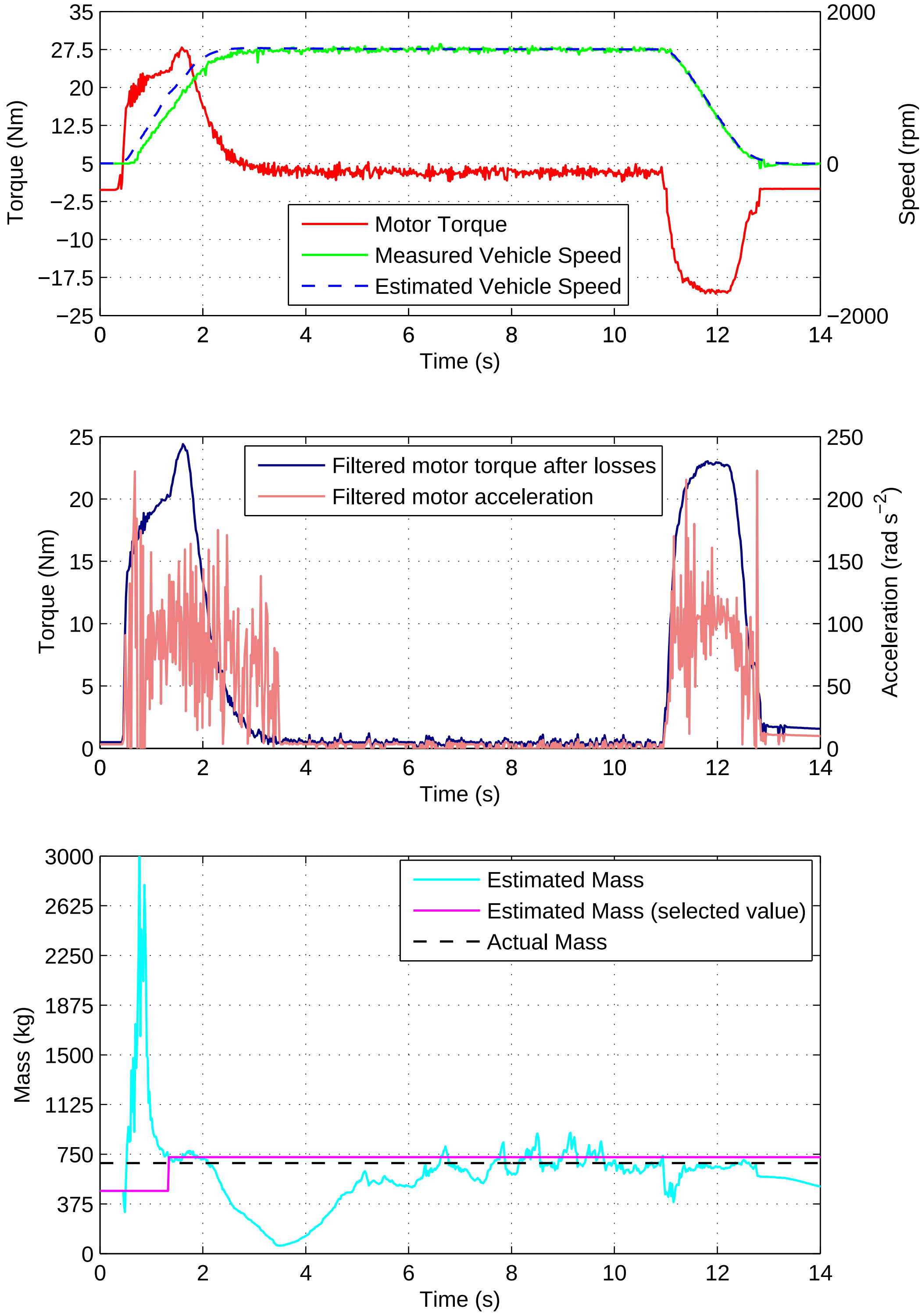

Figure 6.34: Experimental results of speed estimator response with $200kg$ load mass in speed mode

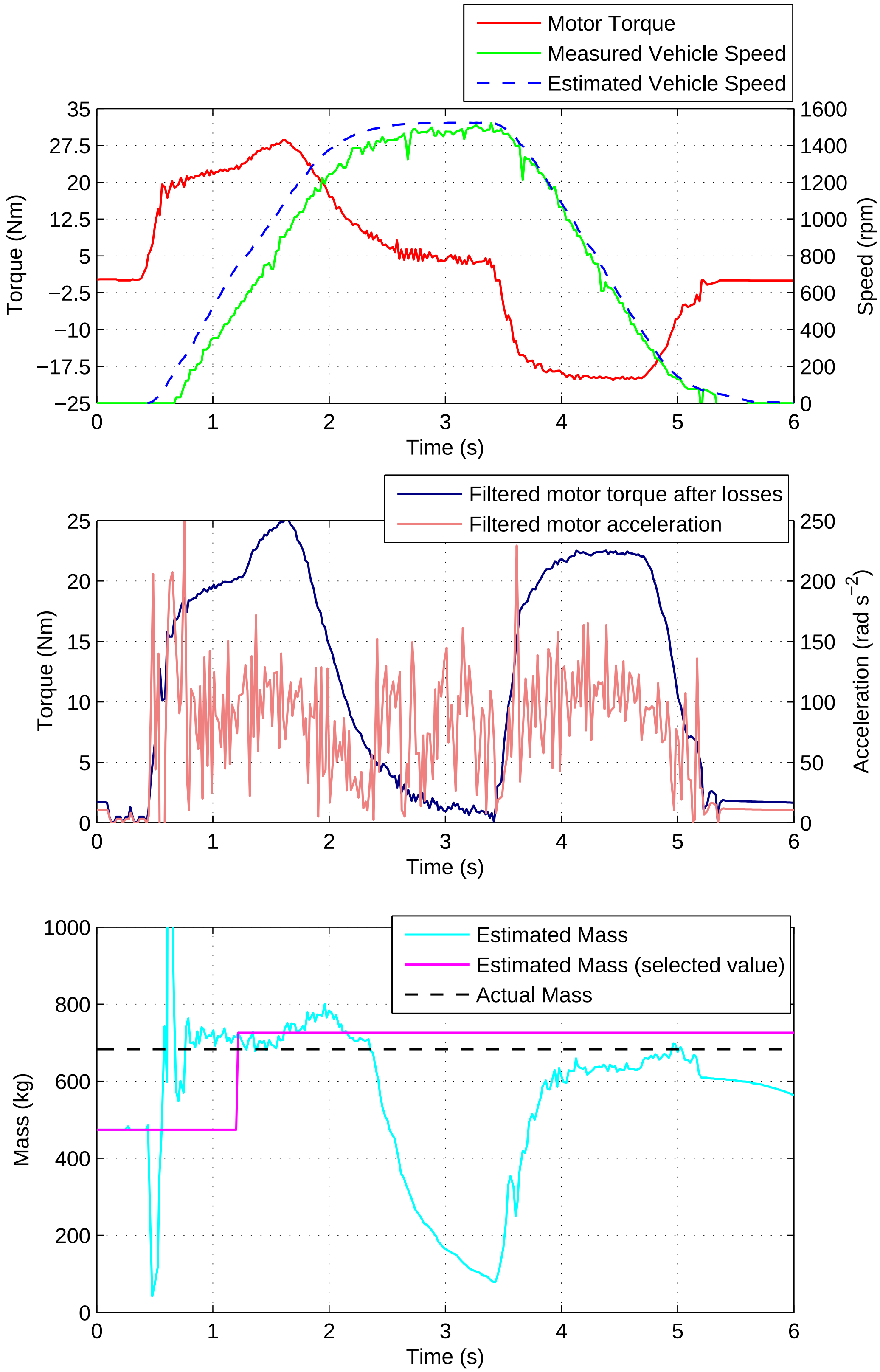

Figure 6.35: Experimental results of speed estimator response with $200kg$ load mass in speed mode

### 6.3.9 Discussion

The above simulation results in section 6.3.7 show that even large mass changes can give good estimation accuracy after the mass has been found, although when driving up a gradient there is no improvement over the non-mass compensated results. The experimental results in section 6.3.8 also confirm that with a typical $200kg$ mass change, once the mass has been found, that the speed estimate is fairly accurate.

But as the time taken to obtain the mass estimate is fixed, the simulation results show that higher accelerations causes there to be a larger error, as more distance has been driven before the mass is updated. The time to find the mass could be greatly improved just by changing the mass selection procedure so that it does not take as long. The mass estimate is activated during the initial acceleration transient, in other work it is common to avoid these sorts of conditions.

There has not been much published work for correcting a two-mass model speed estimator using a second load inertia estimator. In this case it was found to be the least computational requirement method. The few other examples of correcting a Kalman Filter with RLS [23, 24] only consider single mass systems.

If the vehicle losses are modelled with an incorrect stiction coefficient, the mass estimate was seen to be different during acceleration and braking. As these parameters only change very slowly over time, it would be useful to develop a scheme that compares the two mass estimates to correct these parameters and stores them. A higher mass estimate during acceleration than braking indicates that the modelled losses are not high enough.

## 6.4 Tuning the Kalman Filter Noise Matrices

### 6.4.1 Importance of Tuning

The process noise matrix **Q** has a strong influence on the performance of the Kalman Filter, as it describes the accuracy of the model and so controls the balance between noise reduction and estimation error. Larger values of **Q** suggest larger process noise so therefore more correction based on measurements. The previous trial and error tuned **Q** showed that the estimator had poor robustness to mass changes and gradients if these are not also estimated and compensated.

### 6.4.2 Cost Function

The performance of the estimator is judged depending upon the per unit (PU) error of the motor speed (6.56) and vehicle speed (6.57) estimation, and also the noise reduction of the simulated signals compared to the measured ones (6.58) and (6.59). It is quite common to try to minimise the estimation error only [25,37], but in this case the noise is also considered as the Kalman Filter should be tuned to compromise between using the model and measured information. Without considering the noise there is the tendency for the feedback to be favoured to much, leading to a noisy estimate. This has been carried out using the same method as used for determining matrix **R**; by extracting the noise from the signal (measured and estimated) using high pass filtering and then measuring the variance of this noise signal.

It is desirable to find values for **Q** that are robust to changes in the vehicle's mass; so that accurate vehicle speed can be estimated without having to correct for mass changes using Recursive Least Squares. As the **Q** optimisation is carried out off-line using measured vehicle data, it is possible to run the estimator under more than one condition simultaneously for each potential value for **Q**, the cost function is averaged across the different conditions to give $J_{avg}$ (6.61). Measured data is recorded experimentally for motor torque, motor speed and vehicle speed, for both unloaded and loaded conditions at one speed level. This will double the computational requirement to evaluate each chromosome but will produce the optimal **Q** for both unloaded and loaded driving conditions, shown in Fig. 6.36. This could also be expanded for different driving speeds, torque levels and inclines, rather than just the 1500$rpm$ speed used in these tests.

It is important that the estimator in the simulation uses the same fixed point formatting as used on the actual vehicle implementation. The reduced accuracy of fixed point can be regarded as a form of process noise [17] and so including its effect here will ensure that the found value of **Q** considers this.

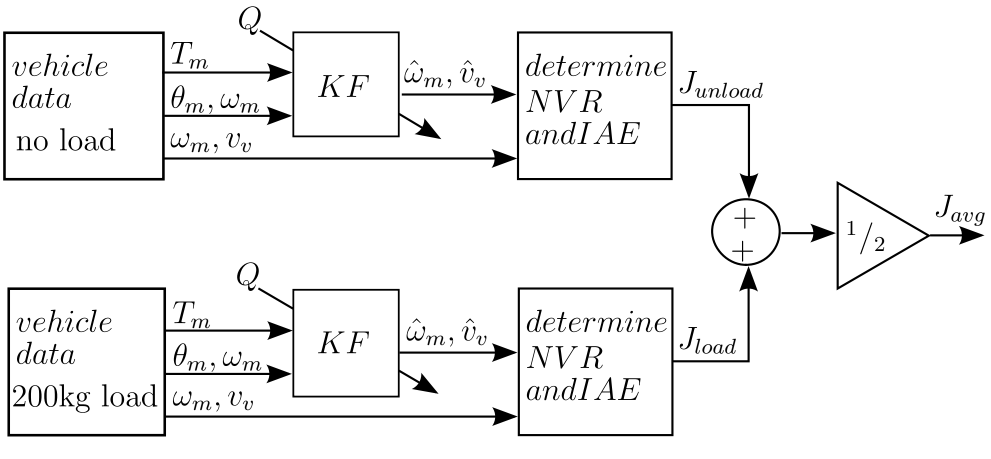

Figure 6.36: Diagram showing how the potential process noise values are simultaneously evaluated on two sets of measured vehicle data

Integrated absolute error (IAE) for motor speed (6.56) and vehicle speed (6.57) is calculated by working out the PU size of the absolute error, the integrated error is divided by the integrated measured speed, to give an IAE in the 0 to 1 range:

$$IAE\,\omega_m(PU) = \frac{\sum_{k=1}^{n} ||\omega_m(k)| - |\hat{\omega_m(k)}||}{\sum_{k=1}^{n} |\omega_m(k)|} \tag{6.56}$$

$$IAE\,v_v(PU) = \frac{\sum_{k=1}^{n} ||v_v(k)| - |\hat{v_v(k)}||}{\sum_{k=1}^{n} |v_v(k)|} \tag{6.57}$$

Ideally the measurement noise variance (NV) will be reduced by the KF (NR noise reduction) for motor speed (6.58) and vehicle speed (6.59), the noise variance reduction (NVR) as a ratio (0 to 1) is calculated:

$$NVR\,\omega_m(PU) = \frac{N\hat{V}\omega_m}{NV\omega_m} \tag{6.58}$$

$$NVR\,v_v(PU) = \frac{N\hat{V}v_v}{NV\,v_v} \tag{6.59}$$

Cost function $J$ for both unloaded or loaded conditions (6.60):

$$J_{unload/load} = \frac{IAE\,\omega_m + IAE\,v_v + NVR\,\omega_m + NVR\,v_v}{4} \tag{6.60}$$

Overall cost function $J_{avg}$ is the average cost function for both unloaded and loaded conditions (6.61):

$$J_{avg} = \frac{J_{unload} + J_{load}}{2} \tag{6.61}$$

### 6.4.3 Genetic Algorithm Optimisation

Genetic Algorithm (GA) is a search and optimisation algorithm that is used to find the best solution to a problem where the performance can be evaluated with a cost function. In this case finding the optimum value for the process noise matrix **Q** to minimise the integrated error of the estimated states and also reduce the noise content of these signals. The technique mimics natural evolution and is suitable for use on non-linear and noisy systems, as it can avoid local minima and is derivative free [36].

The basic operation of the algorithm [36] follows these steps:

- Initialise the population of 100 “chromosomes” (vectors) with random possible solutions for **Q**. These are initialised within a chosen predetermined range that are similar to the expected values. As matrix **Q** has 5 diagonal values (the rest of the values are zero), each “chromosome” (vector) contains 5 “genes” (parameters).

- Motor torque $T_m$, motor speed $\omega_m$ and vehicle speed $v_v$ recorded from vehicle experiments are used to run the estimator off-line with each of the possible set of values for **Q**. The Kalman gain **K** is calculated for the potential **Q** values, and then a fixed Kalman gain estimator is used. This then generates a performance indication of the estimator using a cost function - see section 6.4.2.

- Selection is then carried out based upon the fitness result in the step above. The highest performing (lowest cost function value) “chromosomes” (vectors of parameters) are selected to form a pool of optimal candidates suitable for creating the next generation.

- In order to create the next generation, crossover with a probability ratio of 0.6, is performed on the pool of suitable parents above. This takes two of the chromosomes and swaps over some of the genes randomly to try to create a better new pair of chromosomes.

- Non-linear systems can have local minima that are not the most optimal solution. Mutation is therefore performed, with a probability ratio of 0.35, to ensure that the optimisation finds the overall most suitable values. This mutation works by randomly changing a gene by a small amount in one of the new chromosomes. Normally the mutation ratio would be quite low, but in this case it is higher due to there being many non-optimal minima that otherwise would cause the process to stall. As the genes are only changed by a small amount the effect of this at each step is quite small.

- The process is then repeated with the newly formed set of possible **Q** values until either the maximum number of iterations is reached (100 iterations), or the desired performance is attained.

The GA algorithm in seen to mostly improve the average cost function performance ($J_{avg}$) at each generation step, see figure 6.37. In some cases, mutation leads to an increase in the average cost function ($J_{avg}$), for example at around 15 generations. But ultimately this leads to the minimum cost function decreasing at around 30 generations, meaning improved values have been found. Without mutation the algorithm could converge to a non-optimal minima and not reach the optimal point.

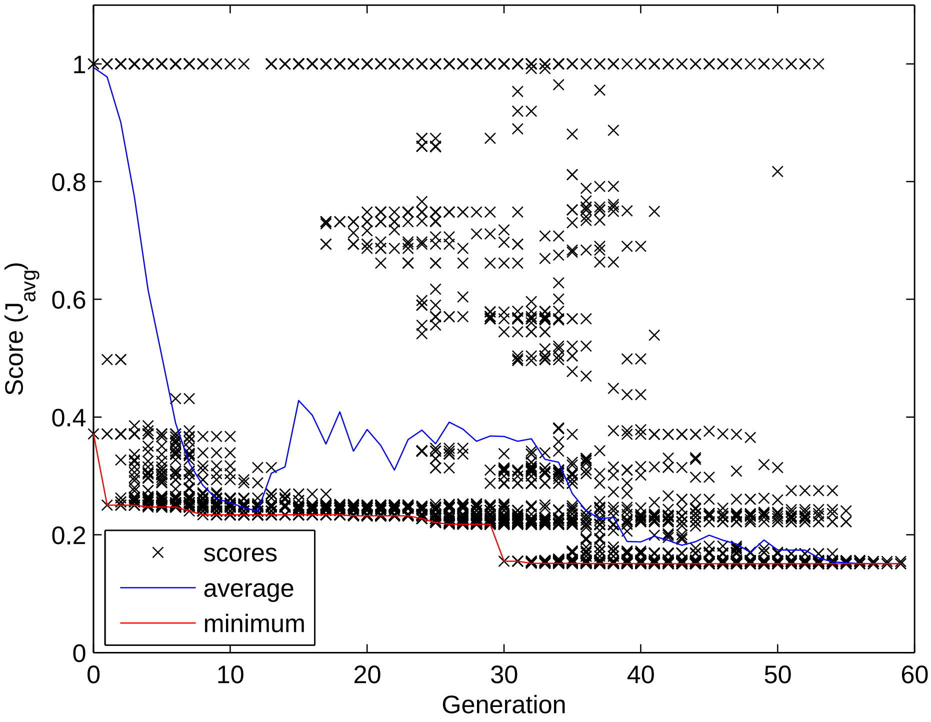

Figure 6.37: How the cost function score ($J_{avg}$) improves with each iteration of the Genetic Algorithm

### 6.4.4 Results

Now that the optimal value of **Q** has been found, the effect of mass changes on the estimator accuracy is tested. This is initially tested in simulation, then repeated with experimental data. The first test is with no mass change or gradient, shown in Fig. 6.38. It can be seen that the speed estimation is very accurate and even at $5000rpm$ the maximum error is $10rpm$. For the higher speed tests there is seen to be some high frequency oscillation or noise of around $2rpm$ to $3rpm$. The estimator tuning scheme only used data at $1500rpm$, if this was expanded to a range of vehicle speeds then this noise could be reduced.

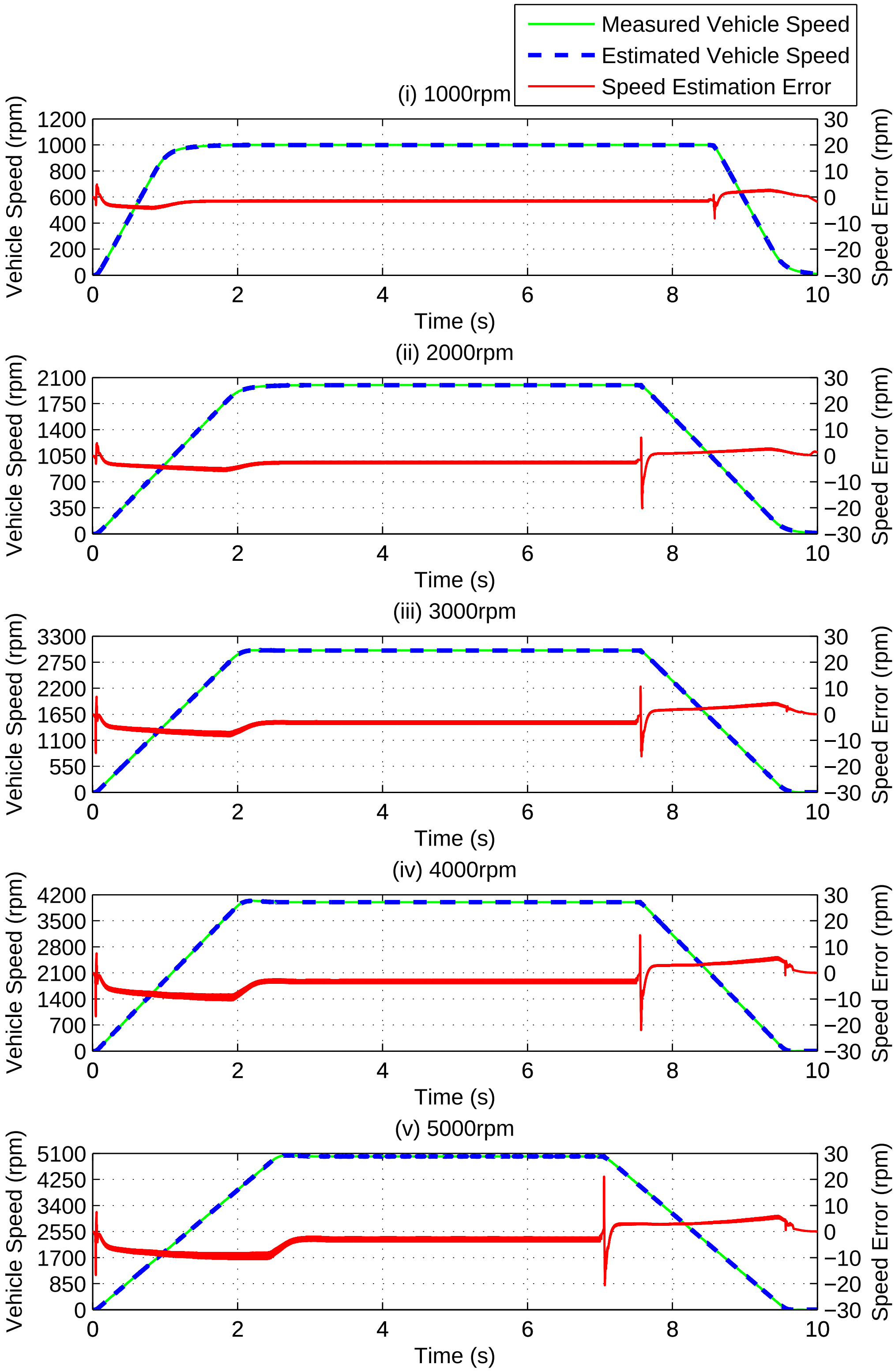

Figure 6.38: Simulation results of speed estimation (with Genetic Algorithm tuned **Q**) with no load mass and no gradient

When the load mass is increased to $200kg$ with no RLS mass estimation, the speed error reaches a worst case value of $30rpm$ during acceleration, sown in Fig. 6.39.

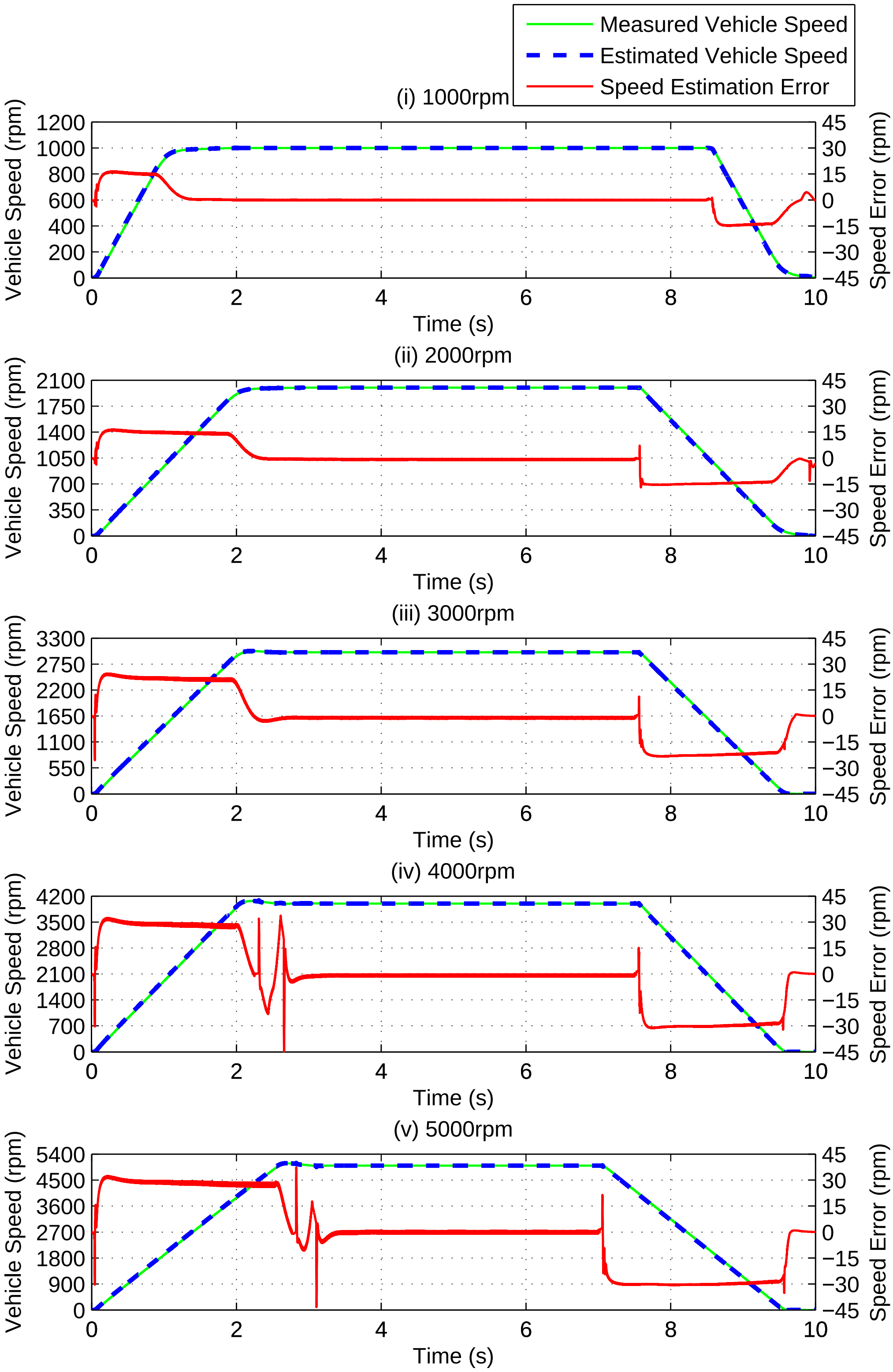

Figure 6.39: Simulation results of speed estimation (with Genetic Algorithm tuned $\mathbf{Q}$) with $200kg$ load mass and no gradient

A higher load mass of $500kg$ doubles the speed estimation error to $60rpm$ for the $5000rpm$ test, shown in Fig. 6.40. This error is quite small and only just over 1% of the vehicle speed.

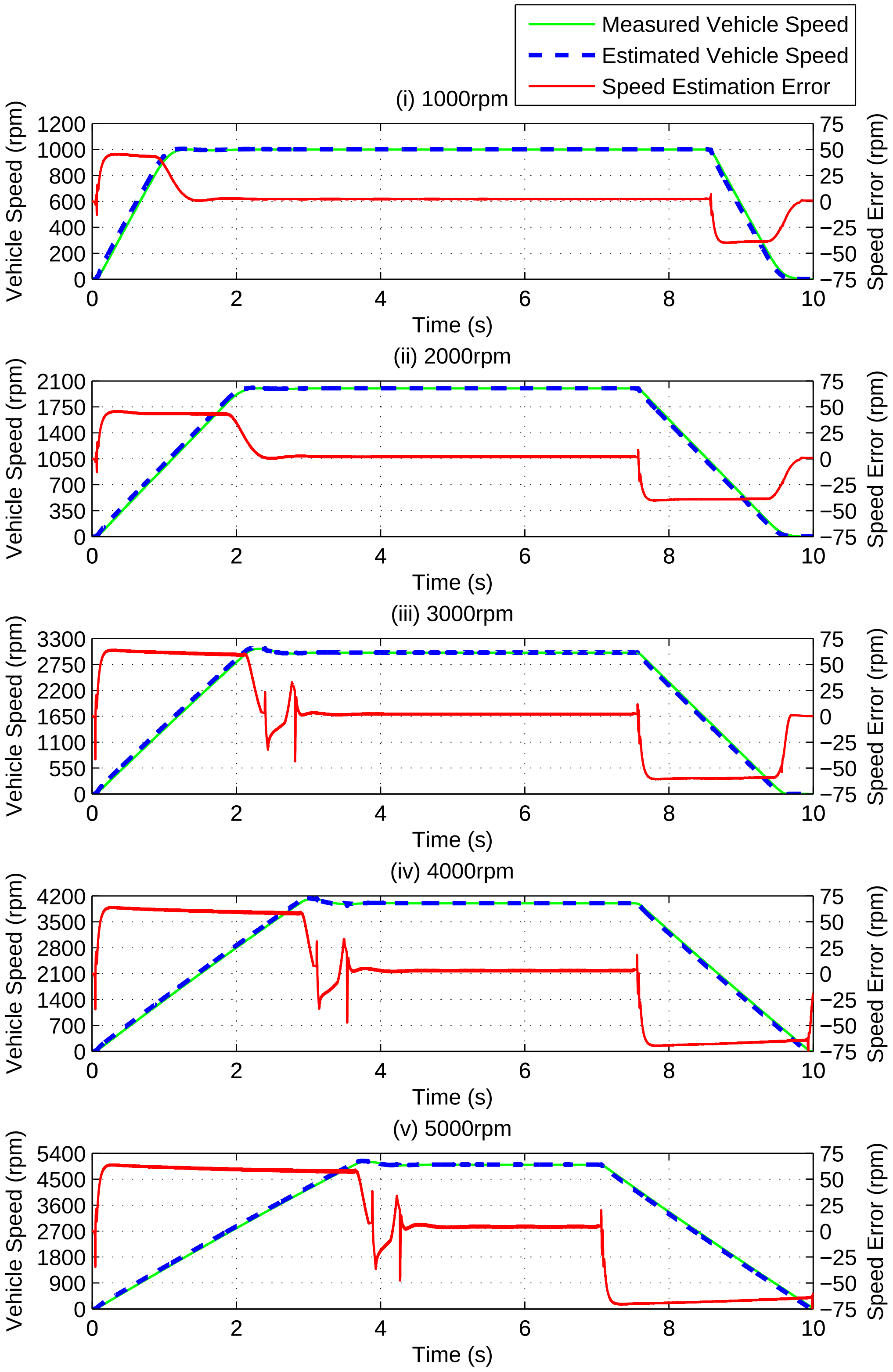

Figure 6.40: Simulation results of speed estimation (with Genetic Algorithm tuned $\mathbf{Q}$) with $500kg$ load mass and no gradient

The accuracy of the GA tuned estimator is also simulated when driving up a road gradient, shown in Fig. 6.41. When these results are compared to that of the trial and error tuned **Q** and **R** it can be seen that the speed error is greatly decreased. It is over 10 times smaller than the error shown in Fig. 6.10 and 6.33.

In the 1000$rpm$ trace (i) the gradient starts after 3.38 seconds, in the 2000$rpm$ trace (ii) it is 3.87 seconds, in the 3000$rpm$ trace (iii) it is 2.91 seconds, in the 4000$rpm$ trace (iv) 3.14 seconds and in the 5000$rpm$ trace (v) the gradient starts after 3.53 seconds.

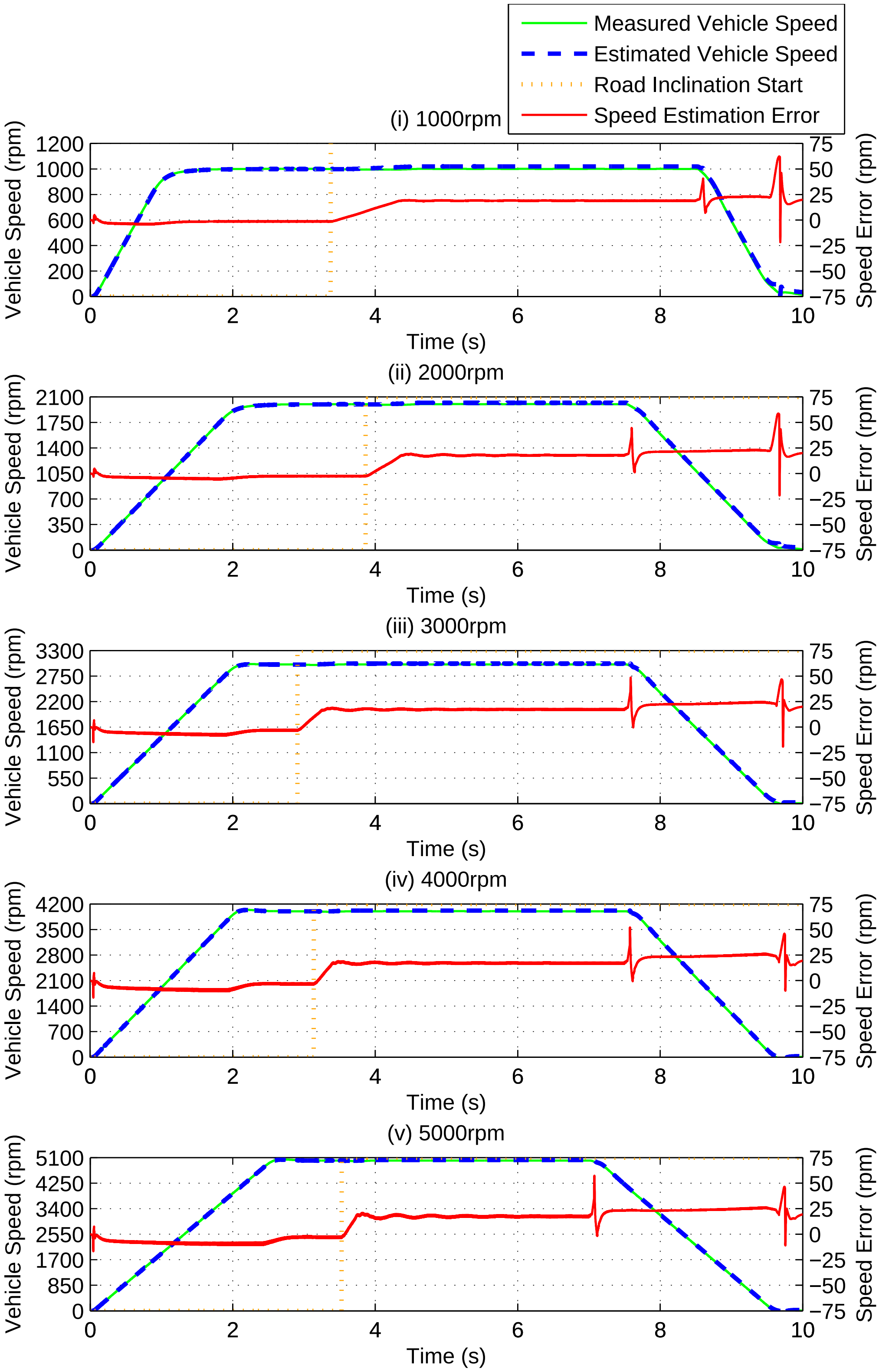

Figure 6.41: Simulation results of speed estimation (with Genetic Algorithm tuned **Q**) with no load mass and 5 degree gradient after $5m$ (i), $10m$ (ii & iii), $15m$ (iv) and $20m$ (v)

Experimentally the response has been tested without RLS mass estimation and compensation (the same as the simulation results), but with a $200kg$ load added, shown in Fig. 6.42.

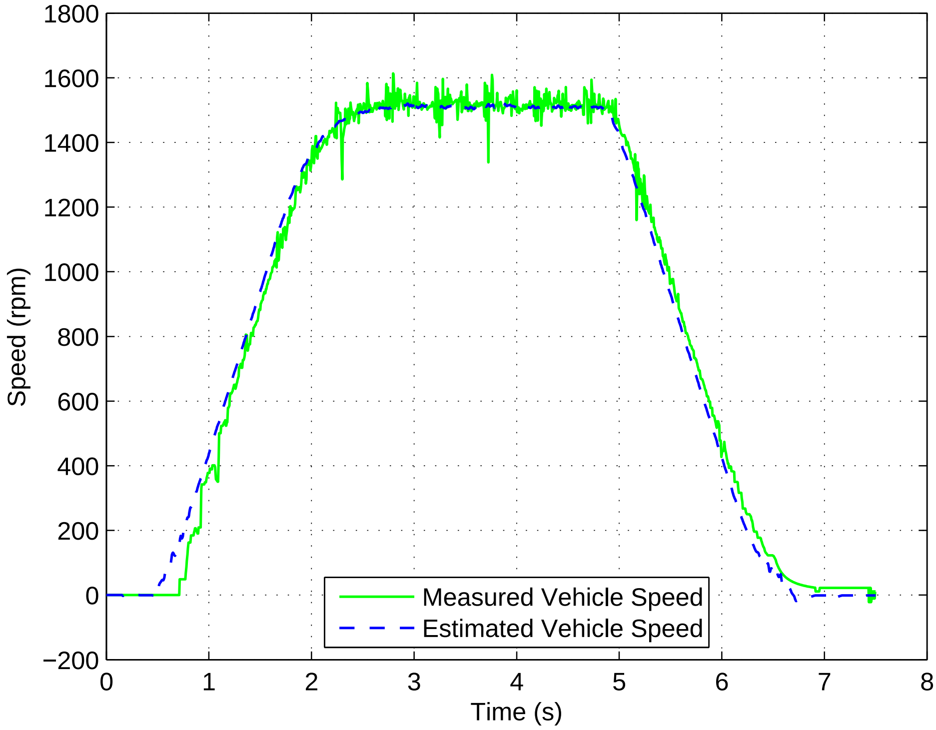

Figure 6.42: Experimental results of speed estimation (with Genetic Algorithm tuned $\mathbf{Q}$) with $200kg$ load mass and no mass estimation or correction

The mass robustness of the estimator is greatly improved compared to the trial and error tuned $\mathbf{Q}$ without RLS mass compensation, shown in Fig. 6.17. There is a small error during the acceleration but the performance is comparable to the trial and error tuned $\mathbf{Q}$ without a load mass change, shown in Fig. 6.14, or better than with a load mass change and using the RLS compensator, shown in Fig. 6.35.

### 6.4.5 Other Optimisation Techniques

It is important to compare the performance of the Genetic Algorithm (GA) method to other optimisation techniques to check that the GA optimised $\mathbf{Q}$ and $\mathbf{R}$ gives the best performance. Two alternative methods were compared: a non-linear least squares based scheme and a non-linear minimisation search algorithm based on the Nelder-Mead method.

For analysing the estimator error, the same method as for GA was used, see section 6.4.2 where the cost function is explained. The issue with this is that there is no direct way of obtaining the differential of the cost function as other methods often require this, an advantage of GA is that this is not required, but it is possible to approximate the differential of the cost function at each iteration to overcome this.

The first alternative scheme is an iterative least squares based optimiser. A single vector of **Q** values is started with, this is evaluated for the loaded and unloaded case to generate a single $J_{avg}$ output. The five covariances in the vector of **Q** values are then each changed by a small amount in both directions with the aim of reducing the $J_{avg}$ score. Partial derivatives are calculated at each step to create a Jacobian matrix. This is continued until either 100 iterations or the optimisation stalls. Due to the highly non-linear nature of the relationship between the vector of covariances in matrix **Q** there are many local minima which cause the non-optimal results to be found. The response when tuned with the Least Squares tuned **Q** are shown in Figs. 6.43 and 6.44, where measured vehicle speed is green, estimated vehicle speed is the blue dashed lines and the speed estimation error is in red.

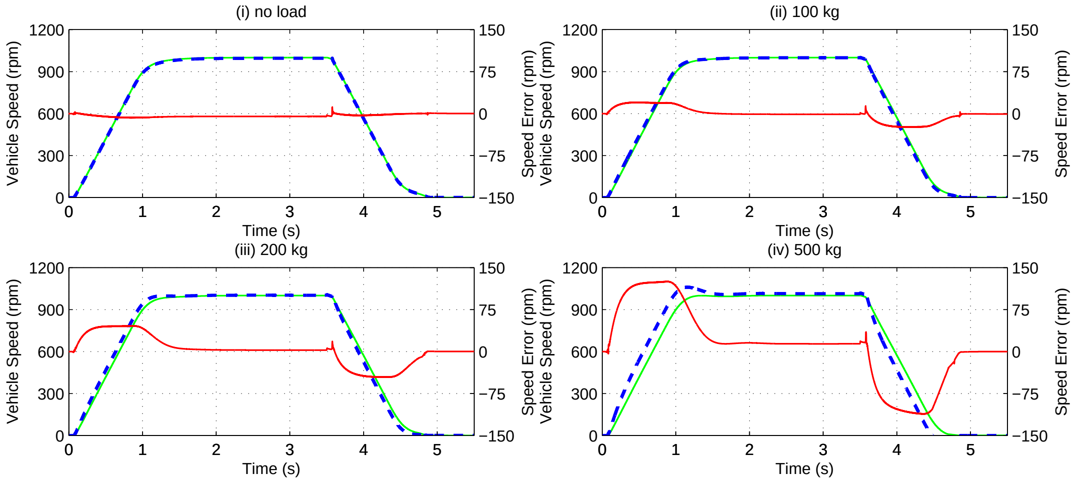

Figure 6.43: Simulation results of speed estimation (with Least Squares tuned **Q**) at 1000$rpm$ with increasing load mass ((i)0$kg$, (ii)100$kg$, (iii)200$kg$, (iv)500$kg$)

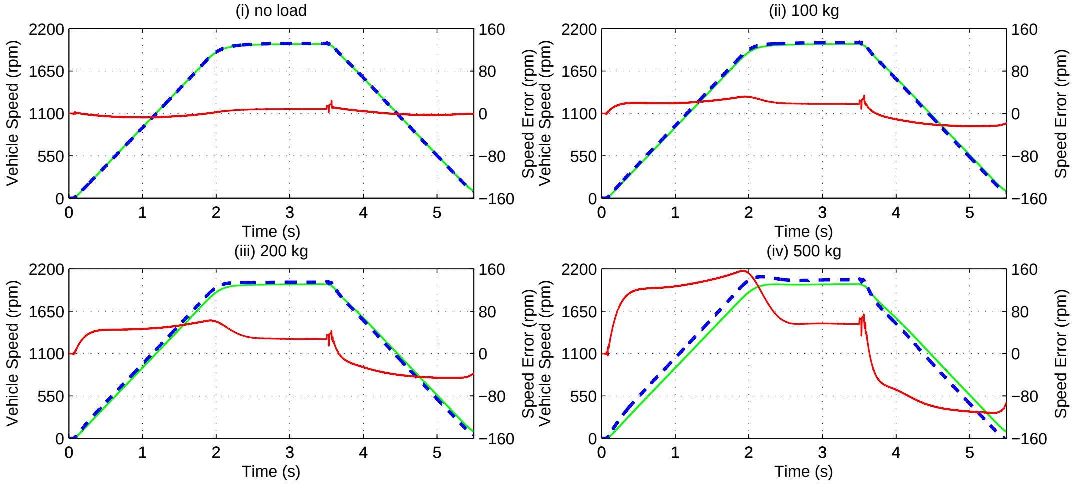

Figure 6.44: Simulation results of speed estimation (with Least Squares tuned **Q**) at 2000$rpm$ with increasing load mass ((i)0$kg$, (ii)100$kg$, (iii)200$kg$, (iv)500$kg$)

The second alternative method is a minimisation search based method which uses simplexes. It follows a similar process as the least squares in that there is only one current set of values for **Q**. This time the derivatives are not calculated at each step making this method easier and faster to calculate. Instead a simplex is created which in this case will have 6 vectors (due to there being 5 variables). New values are tested within this simplex to see if they improve the results. The response when tuned with the Minimisation Search tuned **Q** are shown in Figs. 6.45 and 6.46, where measured vehicle speed is green, estimated vehicle speed is the blue dashed lines and the speed estimation error is in red.

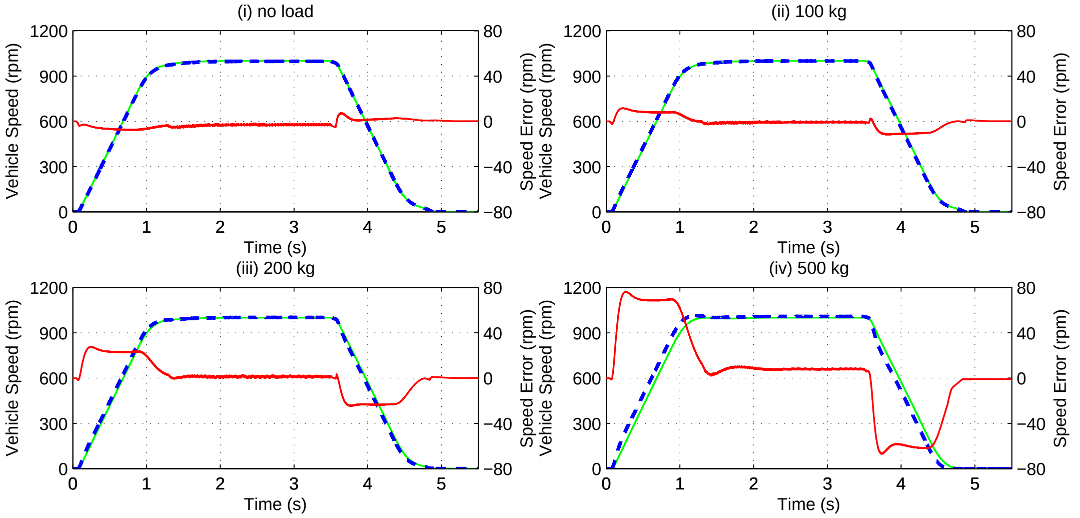

Figure 6.45: Simulation results of speed estimation (with Minimisation Search tuned **Q**) at 1000$rpm$ with increasing load mass ((i)0$kg$, (ii)100$kg$, (iii)200$kg$, (iv)500$kg$)

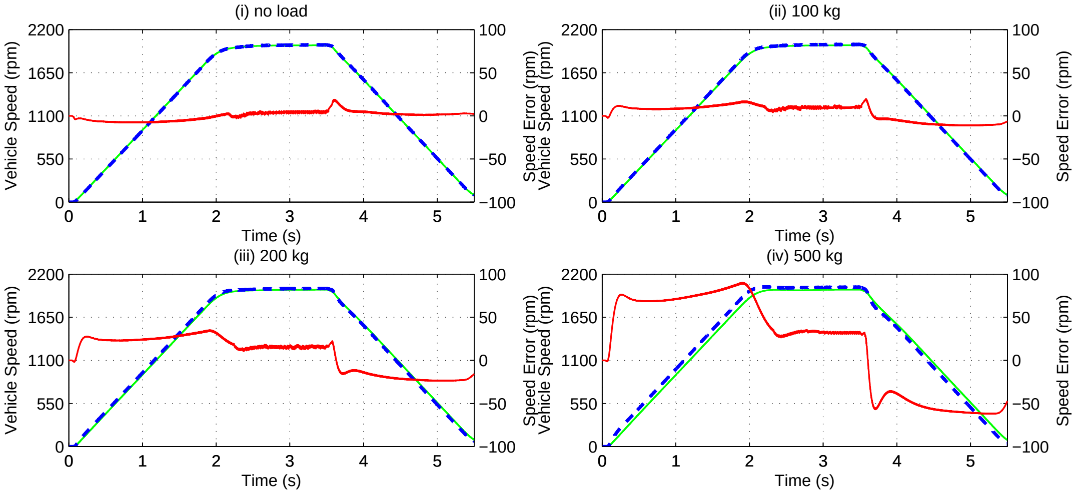

Figure 6.46: Simulation results of speed estimation (with Minimisation Search tuned **Q**) at 2000$rpm$ with increasing load mass ((i)0$kg$, (ii)100$kg$, (iii)200$kg$, (iv)500$kg$)

At no load and with small mass changes the estimation is fairly accurate with both schemes, but any large load added to the vehicle causes a large speed estimation error. The response shown in Figs. 6.43, 6.44, 6.45 and 6.46 was still better than the trial and error tuned **Q**, shown in Fig. 6.17.

### 6.4.6 Optimisation Method Comparison

It is possible for industrial electric vehicle applications to have more significant mass changes than the 200$kg$ experimentally tested, which is only equivalent to around 45% of the vehicles unloaded mass. The robustness of the estimator across larger mass changes has been tested in simulation as it is not possible to increase the load mass to much higher levels experimentally, shown in Fig. 6.47. In this graph T & E refers to trial and error, QA to Genetic Algorithm, NLLS is Non-linear Least Squares and MIN is Minimisation Search, the lowest error shows which estimator gains are the most optimal. This is to see the limits of the estimation with both the trial and error **Q** and the GA tuned **Q**, and also with and without the RLS mass estimator feedback for both cases.

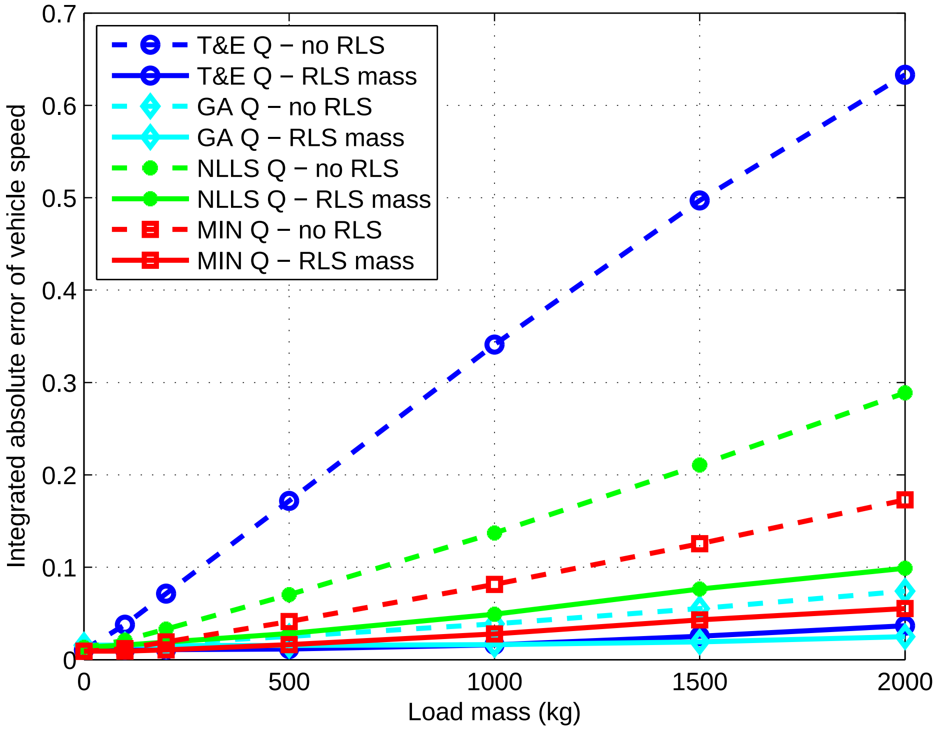

Figure 6.47: Graph showing how the per-unit integrated absolute error of vehicle speed varies with load mass (in simulation)

The best performance over the large mass range was achieved using the GA tuned **Q**, with and without the RLS mass compensation. The performance without mass estimation (for GA tuned **Q**) was almost as good as for the other **Q** values with mass estimation. The trial and error (T & E) tuned **Q** gave the worst response, for both with and without RLS mass compensation. The Least Squares (LS) and the Minimisation Search (MIN) algorithms gave acceptable response with the mass compensation, but poor without.

## 6.5 Considering Alternative Speed Estimation Models

In some of the above simulation results show that when driving on road surfaces with gradients, the effect still is not accounted for and can lead to errors although the better tuned Kalman Filter is more tolerant of non-modelled gradients. A solution is to include load force as a state as well as the pre-calculated input for stiction. This section will also look at how his load force estimation can be used to give a better mass estimate.

### 6.5.1 Additional Load Force State - 6 states

Adding this extra state increases the order of the system to 6, with the states now: motor position $\theta_m$, motor speed $\omega_m$, vehicle distance $d_v$, vehicle speed $v_v$, backlash position $\theta_{bl}$, and load force $F_{load}$, see (6.62). The inputs (6.64) and outputs (6.63) remain the same as the previous 5 state model.

$$\mathbf{x} = \begin{bmatrix} \theta_m \\ \omega_m \\ d_v \\ v_v \\ \theta_{bl} \\ F_{load} \end{bmatrix} \tag{6.62}$$

$$\mathbf{y} = \begin{bmatrix} \theta_m \\ \omega_m \end{bmatrix} \tag{6.63}$$

$$\mathbf{u} = \begin{bmatrix} T_m \\ F_{stic} \end{bmatrix} \tag{6.64}$$

As can be seen in matrix (6.65) and (6.66) the extra state adds mostly zeros to the model, so the additional calculation required is minimal. Although after converting to discrete form a lot of the zero entries end up with small values which cannot always be omitted.

$$\mathbf{A_{co}} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{k_g}{J_m.n^2} & -\frac{b_m + {}^{c_g}/_{n^2}}{J_m} & \frac{k_g}{J_m.n.r} & \frac{c_g}{J_m.n.r} & \frac{k_g}{J_m.n} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{k_g}{m_v.n.r} & \frac{c_g}{m_v.n.r} & -\frac{k_g}{m_v.r^2} & -\frac{b_v + {}^{c_g}/_{r^2}}{m_v} & -\frac{k_g}{m_v.r} & -\frac{1}{m_v} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.65}$$

$$\mathbf{A_{bl}} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{b_m}{J_m} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\frac{b_v}{m_v} & 0 & -\frac{1}{m_v} \\ \frac{k_g}{c_g.n} & \frac{1}{n} & -\frac{k_g}{c_g.r} & -\frac{1}{r} & -\frac{k_g}{c_g} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.66}$$

$$\mathbf{B_{co}} = \mathbf{B_{bl}} = \begin{pmatrix} 0 & 0 \\ \frac{1}{J_m} & 0 \\ 0 & 0 \\ 0 & \frac{1}{m_v} \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{6.67}$$

$$\mathbf{C_{co}} = \mathbf{C_{bl}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.68}$$

### 6.5.2 Position Estimation Simplification - 4 states

As calculation efficiency is very important, reducing the number of states is always a useful improvement. The backlash state is fixed when in the contact mode and only updated in the backlash mode; when in the backlash mode though the estimator operates open loop so it only ever is calculated using the model directly. A simplification is to remove the backlash state from the state space model and calculate it separately still using the estimated states. Also the two remaining position states can be combined as only the position difference is required. This now does mean that position cannot be used as a feedback, but does mean that the position wrap around does not need to be calculated as it is no longer continuously incremented. As each of the position terms were previously multiplied by the axle stiffness ($\sim 10,000$) it often came close to the fixed point limitation, so using position difference avoided this.

This new state is equal to the difference of the other previous position states (6.69), if all reference back to the motor.

$$\theta_{diff} = \theta_m - \theta_s - \theta_{bl} \tag{6.69}$$

The states are now motor speed $\omega_m$, vehicle speed $v_v$, position difference $\theta_{diff}$ referenced to motor, load force $F_{load}$, see (6.70). The inputs (6.72) and the same and the only feedback is now motor speed (6.72).

$$\mathbf{x} = \begin{bmatrix} \omega_m \\ v_v \\ \theta_{diff} \\ F_{load} \end{bmatrix} \tag{6.70}$$

$$\mathbf{y} = \begin{bmatrix} \omega_m \end{bmatrix} \tag{6.71}$$

$$\mathbf{u} = \begin{bmatrix} T_m \\ F_{stic} \end{bmatrix} \tag{6.72}$$

Similar to previously the backlash state space (6.74) has no stiffness and damping compared to contact modes (6.73).

$$\mathbf{A_{co}} = \begin{pmatrix} -\frac{b_m+{}^{c_g}/_{n^2}}{J_m} & \frac{c_g}{J_m.n.r} & \frac{k_g}{J_m.n^2} & 0 \\ \frac{c_g}{m_v.n.r} & -\frac{b_v+{}^{c_g}/_{r^2}}{m_v} & -\frac{k_g}{m_v.r^2} & -\frac{1}{m_v} \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.73}$$

$$\mathbf{A_{bl}} = \begin{pmatrix} -\frac{b_m}{J_m} & 0 & 0 & 0 \\ 0 & -\frac{b_v}{m_v} & 0 & -\frac{1}{m_v} \\ 0 & 0 & -\frac{k_g}{c_g} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.74}$$

$$\mathbf{B_{co}} = \mathbf{B_{bl}} = \begin{pmatrix} \frac{1}{J_m} & 0 \\ 0 & \frac{1}{m_v} \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{6.75}$$

$$\mathbf{C_{co}} = \mathbf{C_{bl}} = \begin{pmatrix} 1 & 0 & 0 & 0 \end{pmatrix} \tag{6.76}$$

The backlash now has to be calculated outside of the estimator, but this is only required when in the backlash state. In (6.65) it can be seen the backlash position state (5th row) is made up of zeros during the contact phase. The calculation for the backlash position is simply the same as row 5 in (6.66).

An additional optimisation could be attained by using a reduced order estimator; removing the motor speed state. This would however lead to increased noise for the motor speed feedback.

### 6.5.3 Simulation Results - 4 state estimator

The same set of simulation tests have been carried out for this new model, as were used in sections 6.2.7, 6.4.4 and 6.3.7. With no mass or gradients shown in Fig. 6.48, the speed estimation error is small with only a small spike when decelerating.

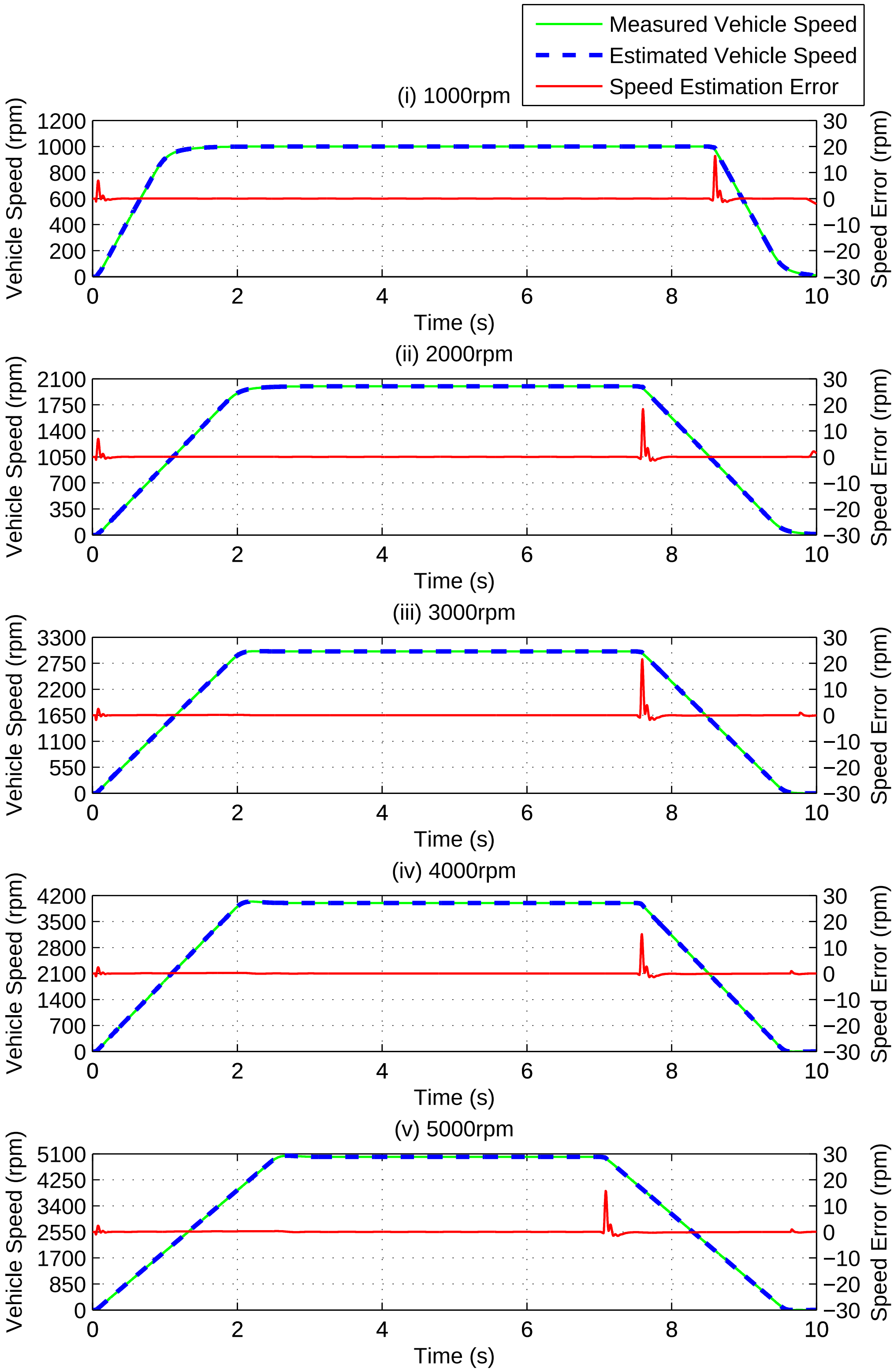

Figure 6.48: Simulation results of speed estimation (4 state) with no load mass and no gradient

When the load mass is increased shown in Fig. 6.49, there is a larger transient error, especially after the initial acceleration. This could be greatly reduced or removed if the mass change was estimated and compensated for during the acceleration, as this error would no longer be present for each acceleration or deceleration after this.

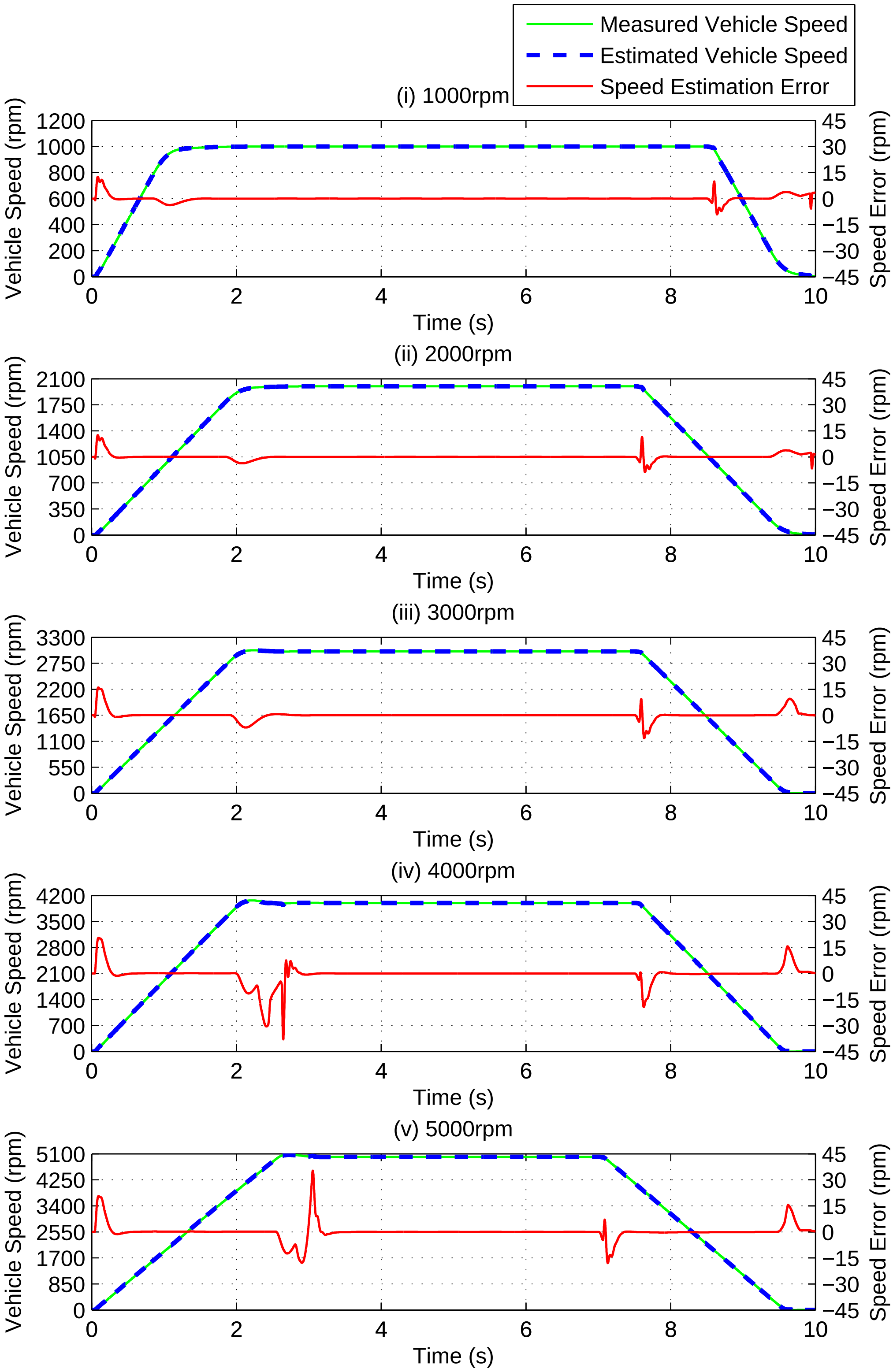

Figure 6.49: Simulation results of speed estimation (4 state) with 200$kg$ load and no gradient

The results shown in Fig. 6.50 with a $500kg$ mass change show the same as the previous graph, but with a larger error.

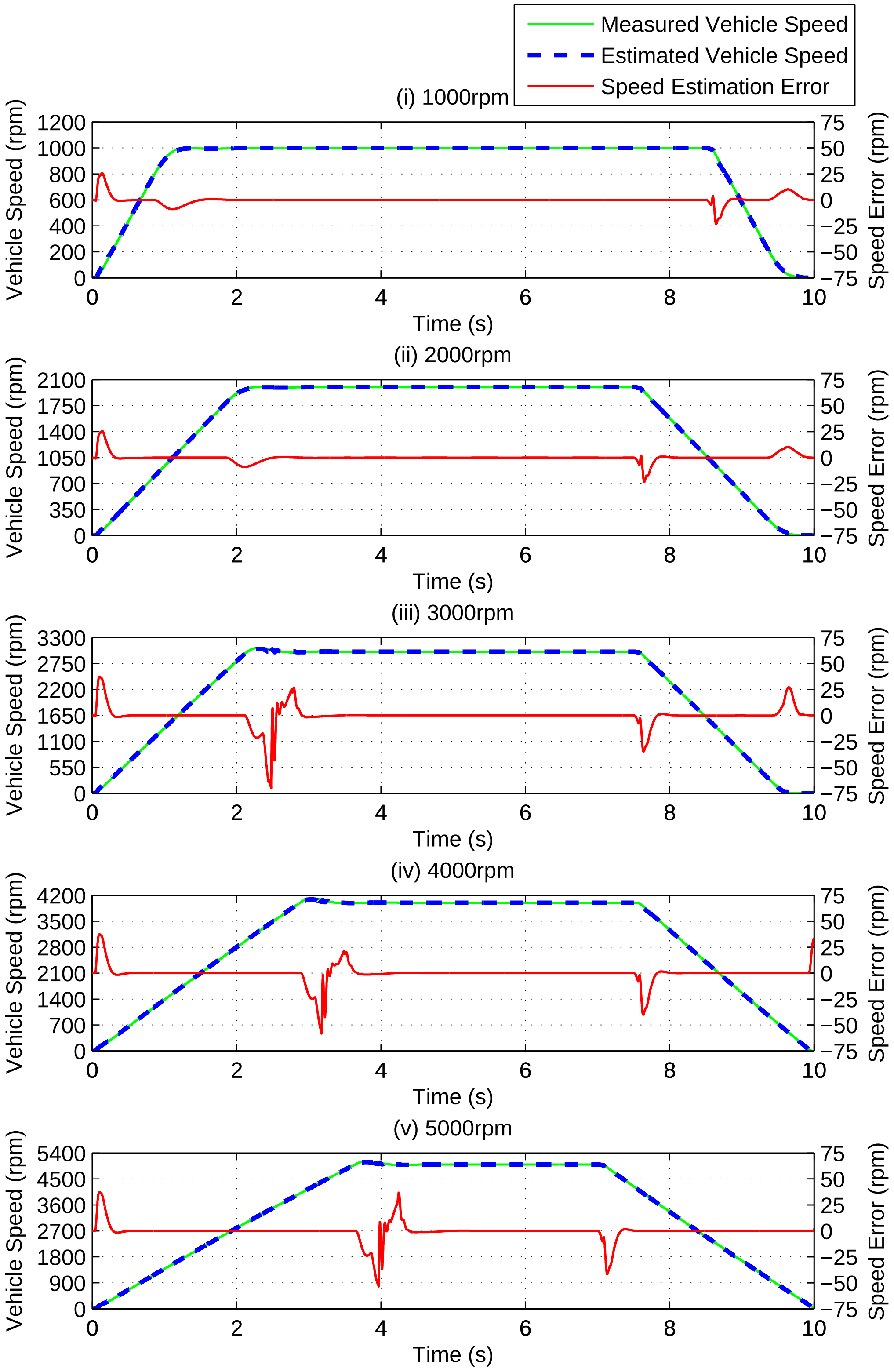

Figure 6.50: Simulation results of speed estimation (4 state) with 500$kg$ load and no gradient

The gradient test shown in Fig. 6.51 shows good speed estimation unlike any of the previous estimation methods, as the load force is able to compensate for this.

In the 1000$rpm$ trace (i) the gradient starts after 3.38 seconds, in the 2000$rpm$ trace (ii) it is 3.87 seconds, in the 3000$rpm$ trace (iii) it is 2.91 seconds, in the 4000$rpm$ trace (iv) 3.14 seconds and in the 5000$rpm$ trace (v) the gradient starts after 3.53 seconds.

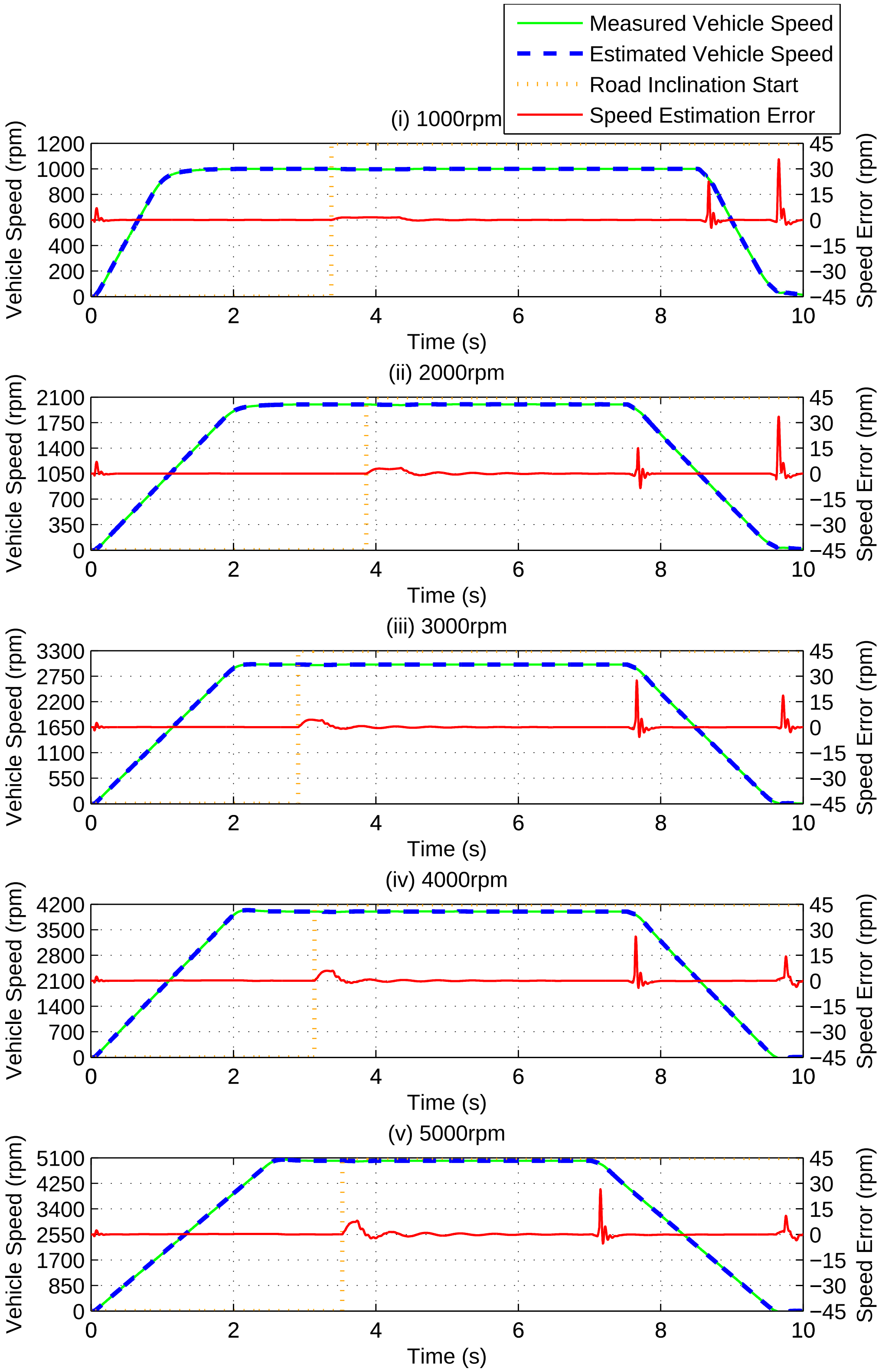

Figure 6.51: Simulation results of speed estimation (4 state) with no load mass and 5 degree gradient after 5$m$ (i), 10$m$ (ii & iii), 15$m$ (iv) and 20$m$ (v)

## 6.6 Summary

This chapter has looked at various methods for estimating vehicle speed and other drivetrain states, such as if the gearbox is in the backlash. The first estimator (5 state) was initially tuned using a trial and error method which gave acceptable results as long as there was no mass or gradient change. The mass changes were then compensated using an additional mass estimator as this gave a much more optimal system (in terms of calculation time) compared to what would be required if using a single non-linear estimator such as the Extended Kalman Filter.

Instead of compensating for mass changes, an alternative idea was considered that compared a number of methods for tuning the Kalman Filter noise matrices to give robustness to mass changes. Although the vehicle speed estimator output can be corrected, during transients and when in the backlash as the estimator has to run in open loop mode, this incorrect mass will give incorrect results, especially if knowledge of the estimated backlash entry and exit times are required.

A different model was finally used to both reduce the order of the system whilst adding an extra load force state. This extra state is able to compensate for non-modelled losses. It is also able to be used for estimating mass, as during acceleration it is equal to the extra torque (or force) not used to accelerate the unloaded mass, and during steady state speed it is equal to gradients and the tyre stiction (if not correct).

The best performance and accuracy was given by the 4 state estimator. It sometimes had large error spikes during transients for less than $500ms$ but overall the speed error was less than all other examples looked at (which all used the 5 state model): trial and error tuned, trial and error tuned with RLS, genetic algorithm tuned, least squares tuned and minimisation search tuned. Another advantage is that it performed well on gradients and the additional load force state could be used to improve the mass and gradient estimation. This estimator also requires the least amount of processing requirements.

Some other external forces affecting the vehicle estimator are use of the mechanical foot brake and the losses due to cornering. More work needs to be done to test this but the foot brake will be estimated as a greater force in the load force state. A great number of industrial vehicles have steering angle sensors to give feedback of the current steering angle. This could be used with another loss coefficient to compensate for cornering losses.

# Chapter 7

# Vehicle Control Improvements using Estimated States

## 7.1 Introduction

The vehicle dynamics issues chapter (2) demonstrated a number of undesirable driving conditions such as oscillations caused by torque reversals, loss of tyre traction and poor speed control performance. The previous chapter has shown that it is possible to estimate a number of unmeasured vehicle states and parameters. This chapter will now look at how they can be used to improve the vehicle drivability; especially concentrating on torque reversals, gear backlash, smooth vehicle acceleration and avoiding loss of tyre traction. The first three issues are considered in sections 7.2, 7.3 and 7.5 with the tyre slip considered in section 7.4. The first section 7.2 initially considers a scheme that does not require the estimators, only using the motor speed feedback, but it has a number of disadvantages that can be improved upon by using the state feedback. The second section 7.3 uses the vehicle speed estimate as speed feedback compared to using the motor speed. The third section 7.4 looks at preventing tyre slip and the final section 7.5 investigates generating a torque waveform to cancel out the effect of the gearbox backlash.

## 7.2 Motor Acceleration Compensator

### 7.2.1 Introduction

As shown in section 2.3, in the graphs in Fig. 2.6 and Fig. 2.7, it is quite common for torque reversals and fast changes in motor torque to lead to oscillations and non-smooth acceleration: on some applications with extremely low damping this has been seen to persist for 5 seconds or more. This motor speed oscillation leads to a poor vehicle response for the driver as its affect on the acceleration can be obviously noticed.

It is typical to use torque control mode for on-road vehicles and a fast response rate is frequently required. Improving the transition from driving to braking torques, or vice versa, would enhance the response of the vehicle, especially if the scheme requires no knowledge of any drivetrain parameters. Within this section an oscillation compensator is developed that is easy to configure or tune.

### 7.2.2 Theory

If no estimated states are to be used, the only feedback that is available is the measured motor speed. Generally in torque control mode, the speed feedback would only have an effect when exceeding the vehicle speed limit. For damping oscillations, as without estimating anything there is nothing to compare the speed feedback to such as the vehicle speed, it is difficult to use motor speed as a control feedback. Calculating the differential of the motor speed will remove the speed offset from the motor speed and also help predict the future oscillations before they develop and dampen them out. A low pass filter with a cut-off of $25Hz$ is used to ensure that the derivative controller is less affected by noise and responds mainly to the $5Hz$ to $15Hz$ oscillations.

It is important that the derivative controller cannot change the direction of torque being applied by the motor (as it would cause the backlash to be entered); this can be seen as a clamp on the torque damping, so it cannot exceed the normal torque reference, shown in Fig. 7.1.

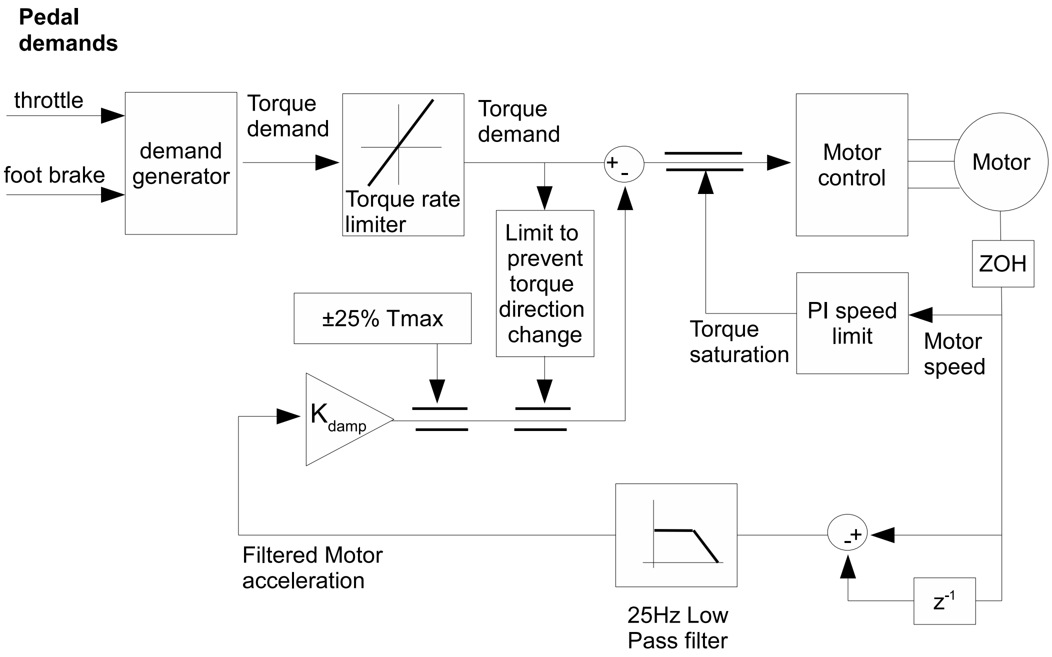

Figure 7.1: Torque mode with acceleration damping algorithm diagram

-

### 7.2.3 Experimental Results

The test vehicle response under normal situations is shown in Fig. 7.2.

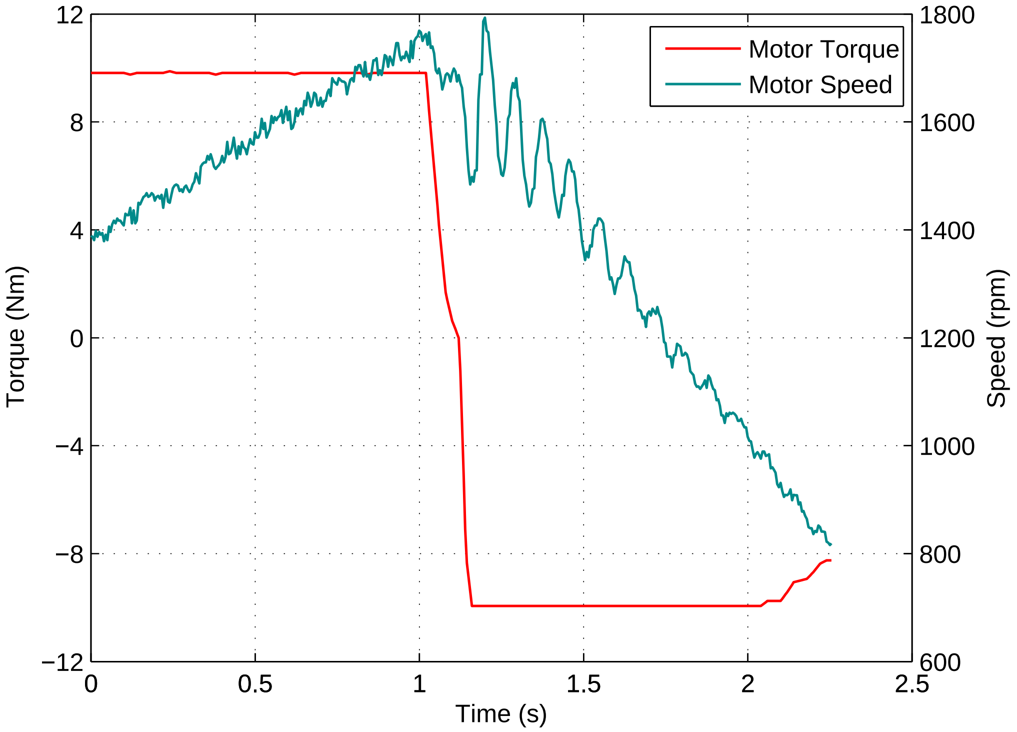

Figure 7.2: Experimental results of response with standard torque mode control

After applying the algorithm the response has improved dramatically, shown in Fig. 7.3. There is one large spike in the speed when the motor goes through the backlash. Although filtered, as the motor speed has noise, this has caused a large amount of noise to be introduced on the torque demand.

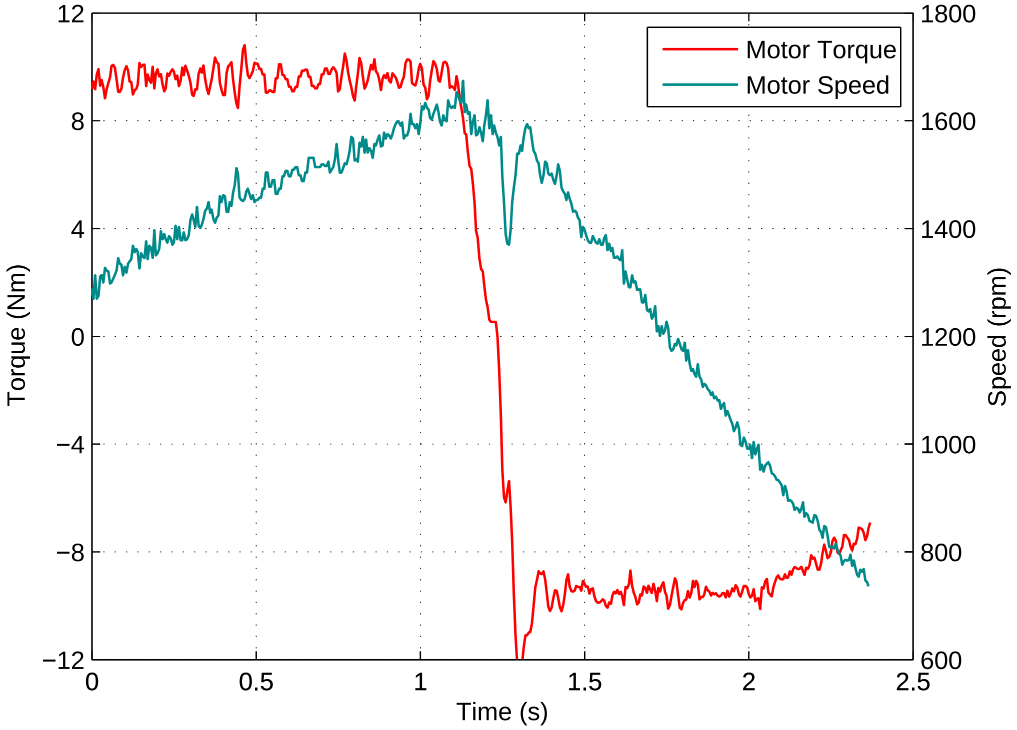

Figure 7.3: Experimental results with the damping algorithm giving improved response

### 7.2.4 Discussion

This scheme has the advantage that only one parameter ($k_{damp}$) has to be tuned, but it does not help improve the speed control response. The torque demand also now has a great deal of noise introduced onto it from the motor acceleration feedback. This makes it more difficult for the current control especially when the available voltage is very limited due to field weakening.

## 7.3 Speed Compensator

### 7.3.1 Introduction

Previously published work has shown that improved speed control response can be achieved using load speed feedback, for applications different than vehicles. Fitting additional sensors is not desirable and the previous chapter 6 has shown that it is possible to estimate vehicle speed. This section will now use this estimate to show how speed control can be greatly improved.

### 7.3.2 Theory

At present the speed control performance is reduced by the bandwidth limitation of using motor speed for control. This leads to a system that is very difficult to tune (see section 2) to give acceptable performance to meet the criteria given in section 1.4.

The overall system is shown in Fig. 7.4, it includes all the proposed solutions in the previous sections of this work. The RLS mass estimator (section 6.3) gives an estimate of mass that is used for correcting the KF and stiction calculation, along with initialising (or adding to) the PID speed loop integrator when there is an acceleration demand change (section 7.3). The Kalman Filter (section 6.2) estimates the vehicle speed that is used by the speed controller and the damping algorithm. The backlash indication is also used to improve the discontinuity in acceleration caused by traversing the backlash (section 7.3), as it is used to limit the torque during the backlash and indicate when it has exited to reset the speed demand and integrator. The sample rates of each estimation or control loop are indicated by $Ts$ in the diagram.

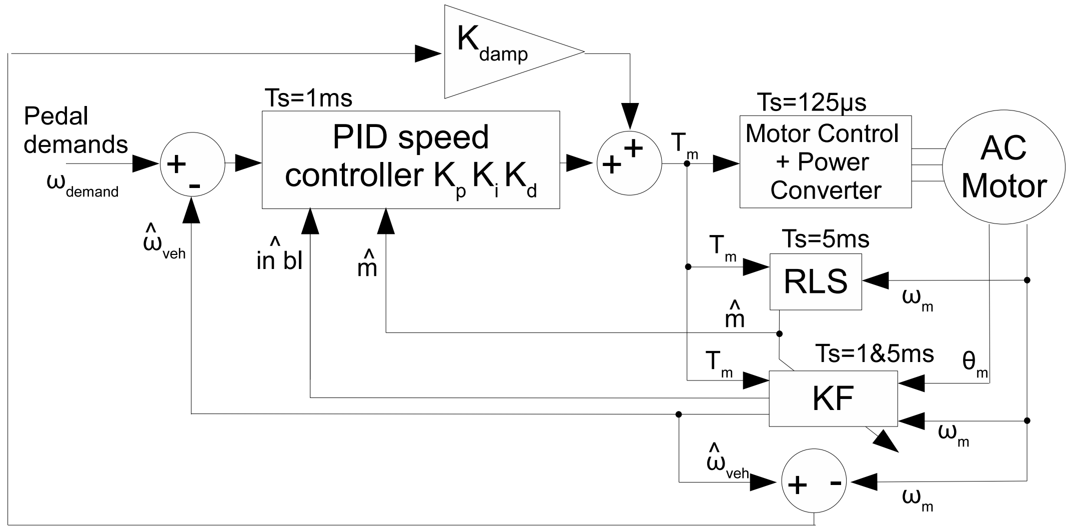

Figure 7.4: Speed mode PID scheme with Kalman Filter and Recursive Least Squares estimation

It is also possible to apply this estimator and control scheme to the speed limit in torque mode, shown in Fig. 7.5.

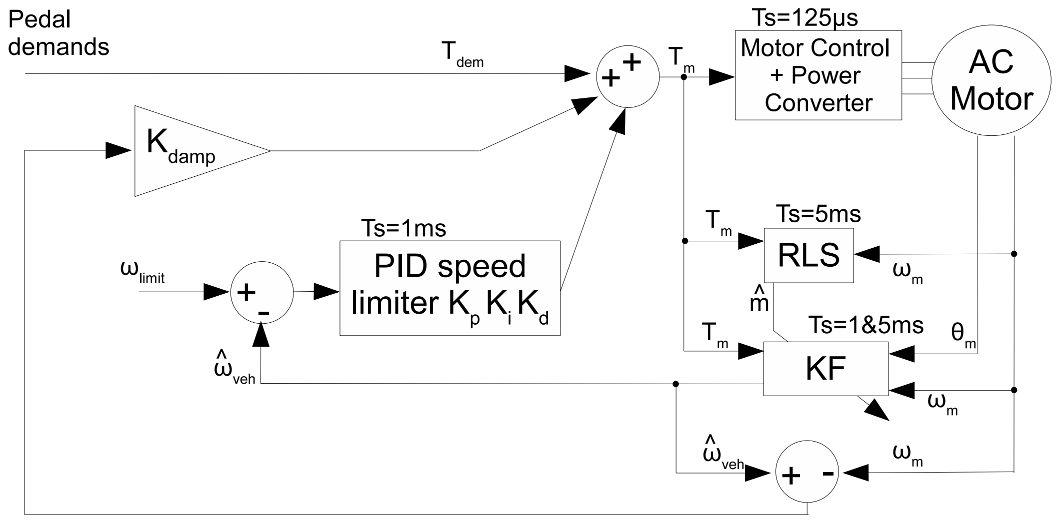

Figure 7.5: Torque mode PI speed limit scheme with Kalman Filter and Recursive Least Squares estimation

### 7.3.3 Simulation Results

The speed control for an industrial EV generally uses speed feedback from the motor encoder, as the sensor is already present for AC motor flux vector control. PI control is used as it can easily be manually tuned to provide satisfactory performance, as shown in Fig. 7.6. Issues arise from the fact that the gains have to be detuned to maintain stability, due to the transient error when using the motor speed feedback to measure vehicle speed and the fact this signal is prone to oscillation. The oscillation in motor speed directly affects the smoothness of the vehicle acceleration. The simulation results in Figs. 7.6, 7.7, 7.8 and 7.9 show the vehicle response when applying a speed demand with a fixed acceleration rate of $1000 rpm/second$ up to a speed of $1500 rpm$, and then decelerating at the same rate after 2 seconds at a fixed speed ($1500 rpm$). The acceleration rate is ramped down when approaching the target speed to help prevent overshoot. The demands have been plotted in Fig. 7.6 only.

#### Motor speed feedback

In Fig. 7.6, although the response is smooth, the acceleration never reaches the desired rate of $1000 rpm/sec$ for a constant time; the acceleration should be steady between 0 and 1.5 seconds. This would give the vehicle a sluggish response that will deteriorate when the vehicle is loaded.

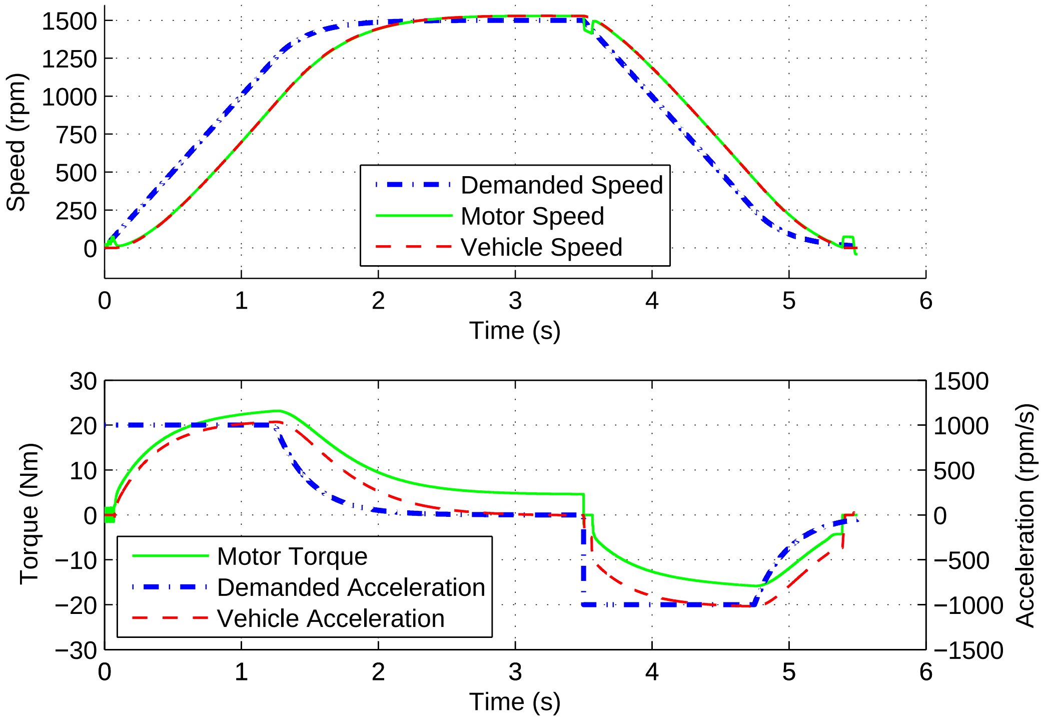

Figure 7.6: Simulation of speed response with motor speed feedback

### Vehicle speed feedback

It has been proposed that improved performance can be obtained through using the measured (or estimated) vehicle speed as speed feedback, shown in Fig. 7.7. Much higher speed gains can be used as the feedback is no longer affected by the motor and drivetrain dynamics. This gives much tighter control of the vehicle speed, and therefore acceleration. It also allows for the control to now include the differential term on speed error (PID controller); previously not possible due to noise on the speed feedback.

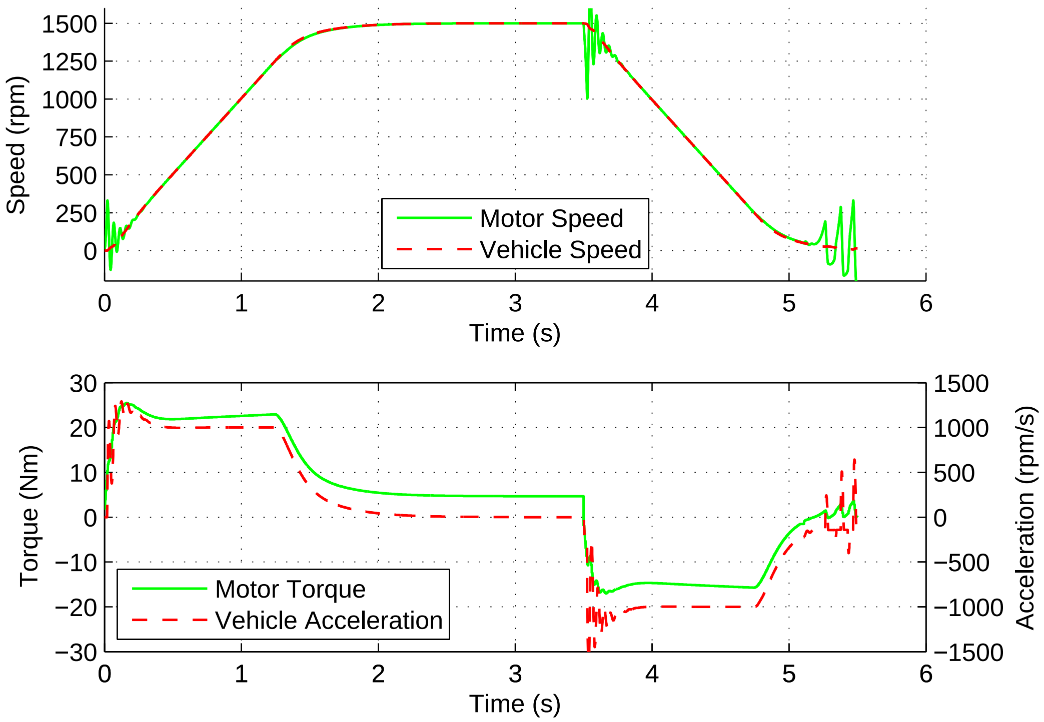

Figure 7.7: Simulation of speed response with vehicle speed feedback

### Vehicle speed feedback with damping gain

Some consideration now needs to be given to the fact that larger gains will lead to faster torque changes and therefore more oscillation in the drivetrain, shown in Fig. 7.7. A simple proportional compensator with control gain $K_{damp}$ is introduced that reduces this significantly, see diagrams in Fig. 7.4 and Fig. 7.5. Note that this gain is different from the one presented in section 7.2. The compensator tries to control the motor speed to be equal to the vehicle speed, shown in Fig. 7.8. This also has the advantage that during the backlash traversal, the torque demand is limited as the motor speed differs greatly from the vehicle speed, the compensator therefore acts to smooth the transition when the gear cogs impact.

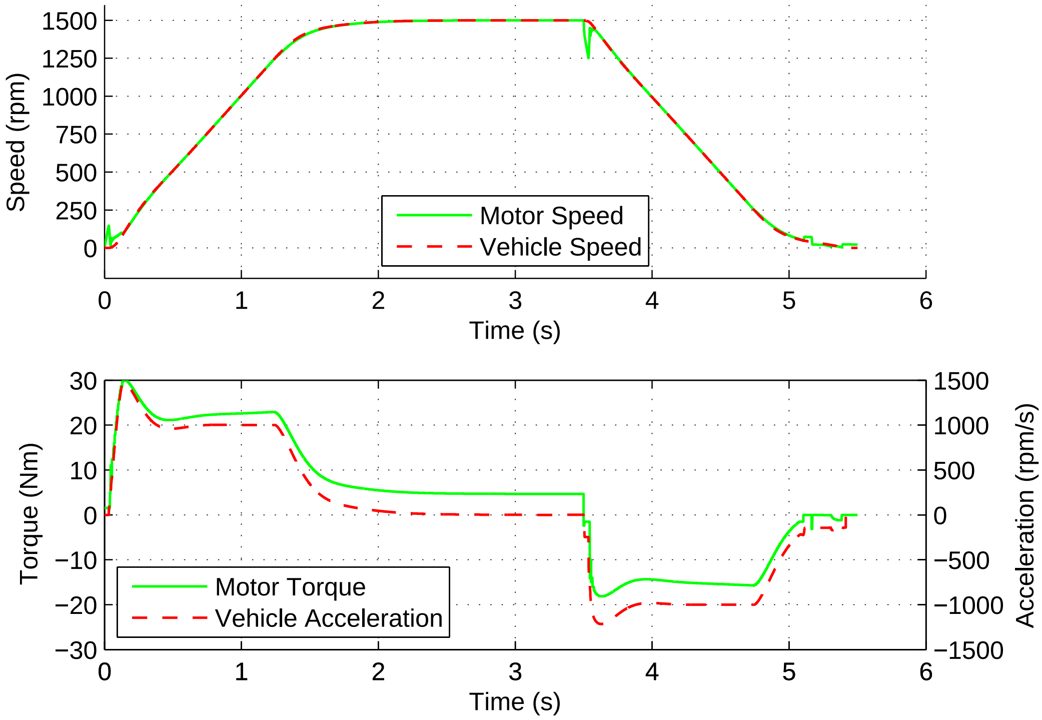

Figure 7.8: Simulation of speed response with vehicle speed feedback and compensator

### Vehicle speed feedback with damping and modified PI

This gives a good speed response as it follows the desired speed demand closely, but the acceleration now has an initial disturbance (overshoot) before settling on the steady state value. This will be felt by the driver as a jerk in the vehicle's response. It is caused by the actual vehicle speed lagging behind the demand whilst going through the backlash, as the motor output torque has no effect on the vehicle during this time due to the disconnect. Then, when in contact mode, the tightly controlled speed loop closes the error quickly.

In order to improve the response further some modifications to the speed control have been proposed. These are:

- Initialising the integrator (or adding to it) to the calculated required torque from the demanded acceleration and vehicle mass. This removes the delay of waiting for the integrator to wind up, giving instant acceleration.
- Resetting the speed demand to the actual vehicle speed when exiting the backlash mode. This prevents the speed loop from having a large error when the backlash closes.

It is now possible to get a very accurately controlled speed response, but with the acceleration remaining level, shown in Fig. 7.9. This will give the vehicle drivability a feel of being very responsive and accurate, but smooth in its acceleration.

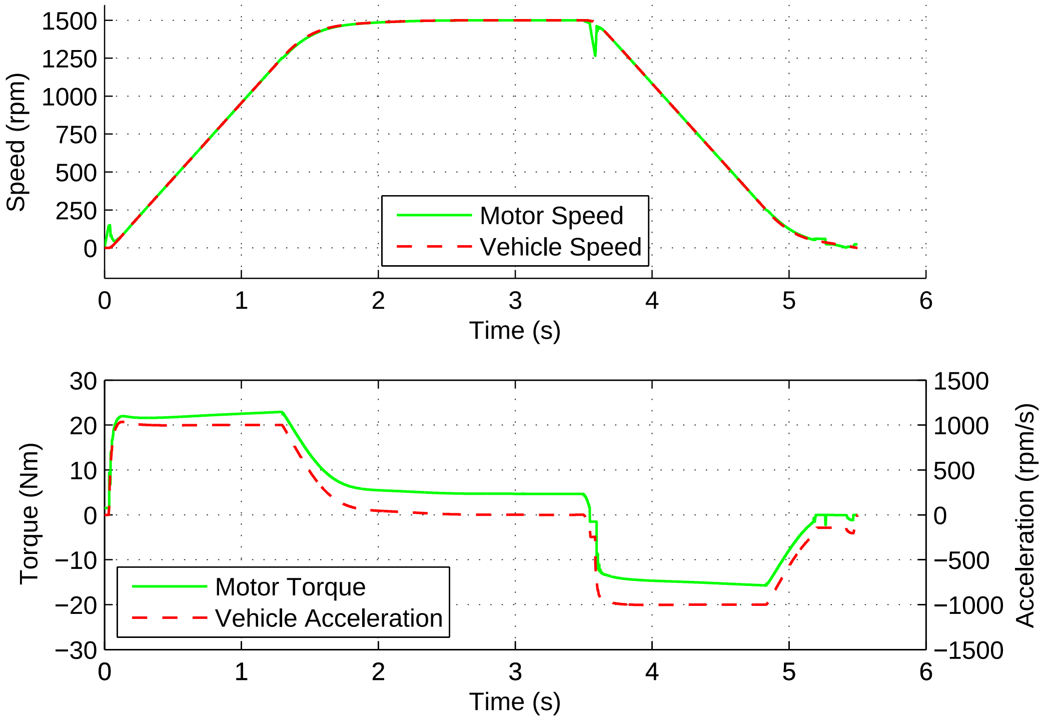

Figure 7.9: Simulation of speed response after implementing all the proposed changes

### 7.3.4 Experimental Results

The issue of a large surge in torque as the speed loop tries to catch up with the demand is shown in Fig. 7.10. The improvements shown in the final part of the previous simulation results section would remove this issue.

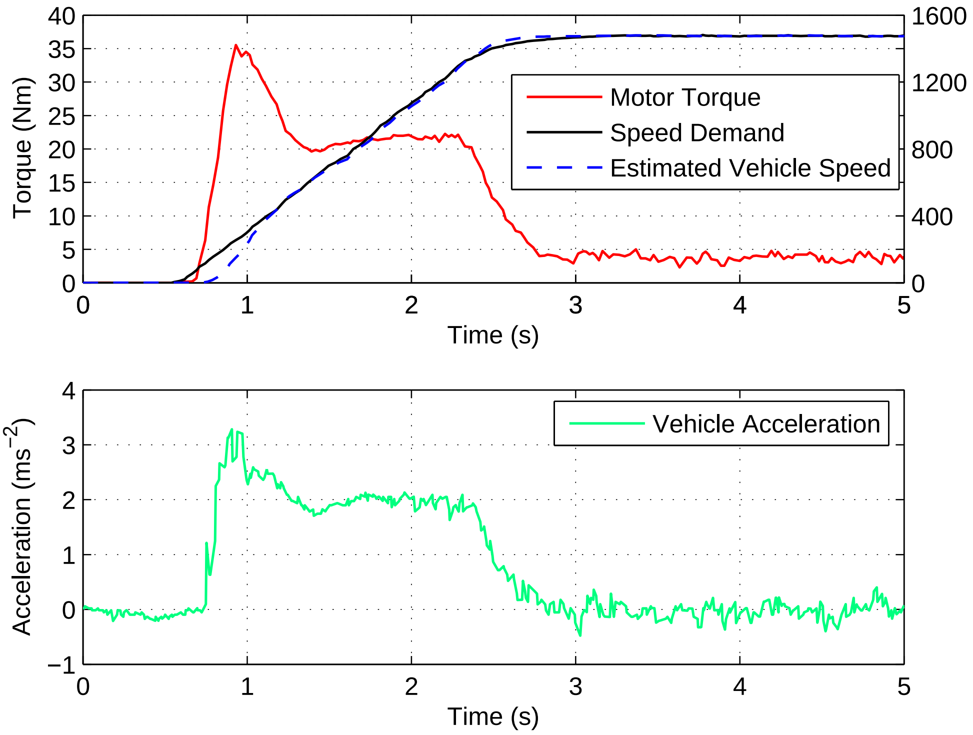

Figure 7.10: Experimental results with speed control closed around estimated vehicle speed - change in torque at change in speed demand

When the mass estimate has been found it changes the gradient of the vehicle speed estimate meaning that it instantly starts to drift away from the demand as the correct mass speed estimate will respond slower, shown in Fig. 7.11. This causes a surge in torque again although this is quite small and could be fixed by applying the new mass over a number of steps.

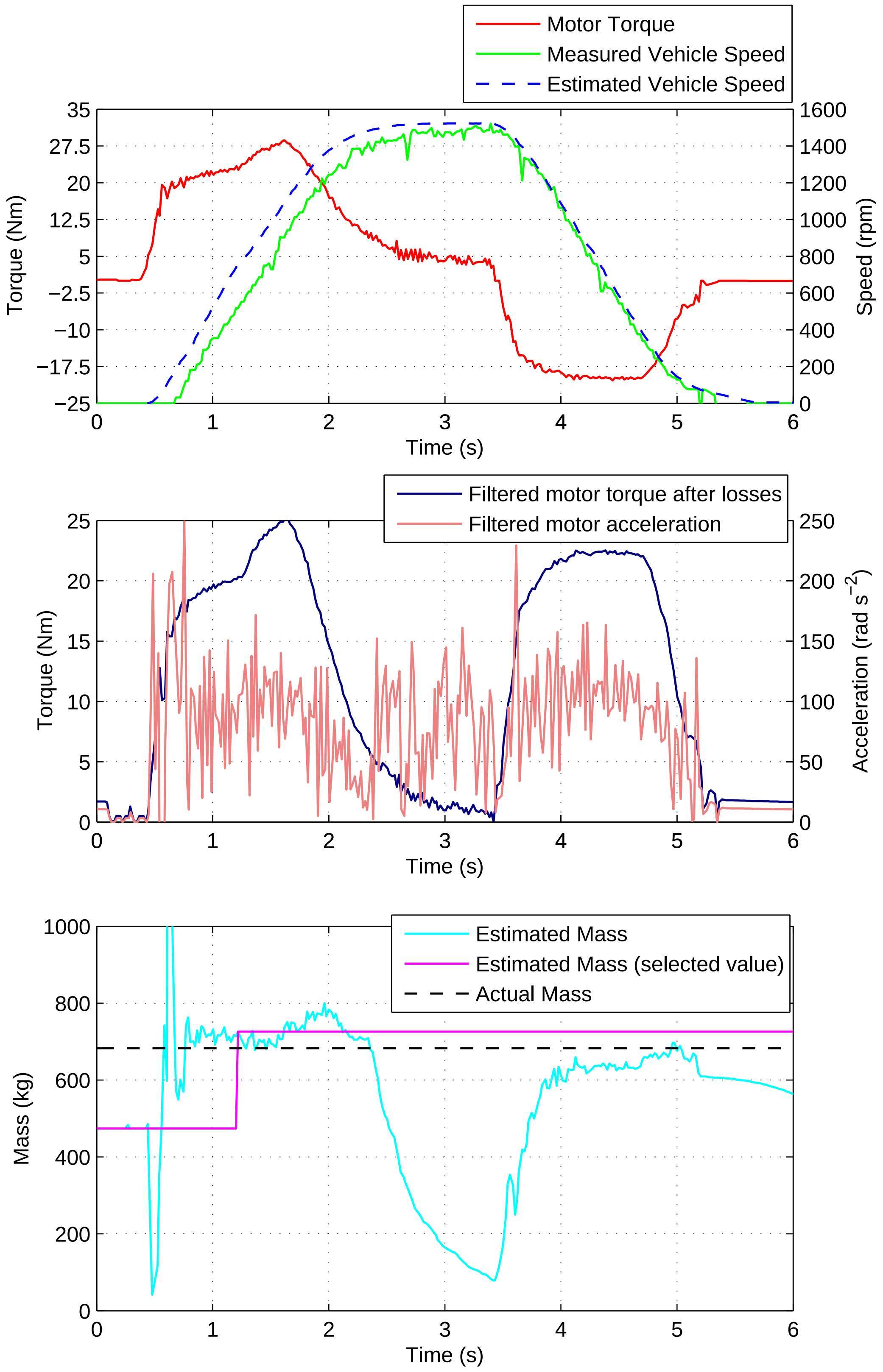

Figure 7.11: Experimental results with speed control closed around estimated vehicle speed - change in torque after new mass estimate

### 7.3.5 Discussion

The main improvement is that much tighter control can be obtained over the standard approach using motor speed feedback. This will avoid the vehicle feeling too sluggish or getting overshoots when reaching the desired speed, as this gives a very poor vehicle performance for the driver, as it leads to the motor to apply braking torques.

Generally the speed loops on the vehicle are very difficult to tune and one of the more time consuming things to set up during vehicle commissioning. Being able to run the speed loops faster and also having feedback from the load that is being controlled greatly improve the range of suitable gains that gives acceptable performance.

## 7.4 Tyre Slip

### 7.4.1 Introduction

There is always a limit to the amount of traction between the road and the tyres. Section 4.5 gave some examples of typical tyre force-slip curves under differing road conditions. This shows that there is always some amount of slip although this is typically ignored as is very small until you cross the top of the tyre slip curve. Electric vehicles commonly use regenerative braking to increase the vehicle range; this includes both to recreate the feel of engine braking and provide stronger braking when the foot-brake is pressed. A great deal of electric vehicles are rear wheel drive which leads to much lower available traction levels during braking due to weight transfer. Ideally the largest amount of regenerative braking torque would be used to maximise energy recovery.

If the braking torque is too high then loss of traction can occur, leading to loss of vehicle control, especially an issue when cornering. If this is caused by, for instance, the regenerative braking torque being demanded from the level of foot-brake push, the wheel lock situation can easily be removed by releasing the pedal. It is much more serious if a situation is encountered where wheel lock causes the wheels to remain locked. In speed control mode it is important that the speed demand tracks the speed feedback when decelerating. Normally when braking in speed mode the speed demand will be reduced at a specified ramp rate. If some external force such as driving up a gradient or use of the foot-brake causes the vehicle to decelerate faster than the desired deceleration rate, it is important the controller does not output drive torque and try to oppose the torque caused by the mechanical foot-brake. This is normally done by forcing the speed demand to follow the speed feedback when braking (if the speed feedback is decreasing faster). If the foot-brake is released then the demanded speed can continue to ramp from the measured speed, causing regenerative braking to resume instantly. If during braking the wheels lock briefly due to use of the foot-brake or driving over a poor traction surface, this will mean the speed demand will be set to zero and the wheels will remain locked until the vehicle comes to a halt.

In torque control mode this is less of an issue, but it is still common to use some speed control functionality in torque mode; examples include hill hold and speed limit. Normally if the speed feedback has indicated the vehicle has come to a stop the hill hold mode can be used to control zero speed to stop the vehicle rolling back on an incline. If the controller thinks the vehicle has stopped as the wheels have briefly locked, it will lead to a similar situation as in the speed mode example above. This usually prevents the vehicle hill hold functionality from being used for on-road vehicles, as with the higher speeds involved the danger from wheel lock increases.

### 7.4.2 Theory

A scheme has been developed that allows for preventing wheel lock without adding any additional sensors to the vehicle. It works by detecting differences in estimated motor and vehicle speeds, shown in Fig. 7.12. If the speeds look to be moving apart, more than what is normally seen during transients. The estimator correction needs to be tuned so that the vehicle speed is not too heavily corrected by the motor speed error, otherwise the two speeds (motor and vehicle) will not separate when the wheels are about to lock.

As the vehicle speed estimator does not model tyre slip, it is important that this scheme can prevent the tyre slip rapidly so that ignoring it remains valid. This is achieved by the estimator switching to open loop as soon as traction is lost, so that the large change in motor speed does not wrongly correct the vehicle speed estimation. Including tyre slip in the estimator adds a third non-linearity, along with the backlash and stiction already included, so this is obviously not desirable.

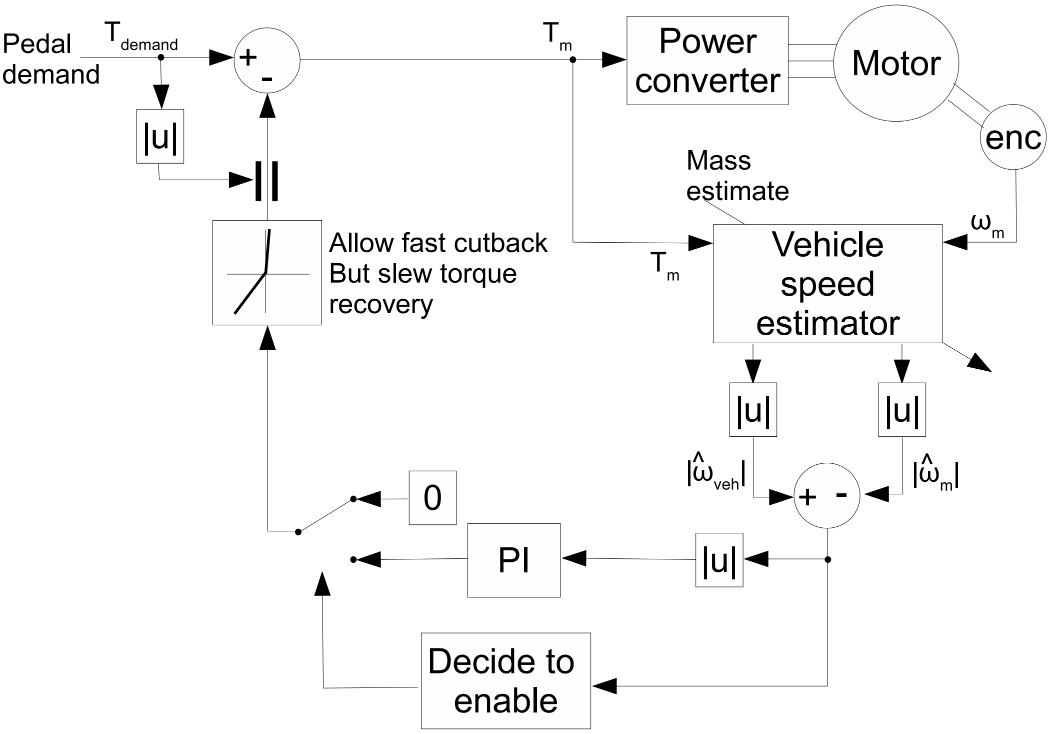

Figure 7.12: Diagram of anti wheel lock scheme

### 7.4.3 Experimental Results

As shown in chapter 2 (accelerating and braking in both directions) and in Figs. 7.13 and 7.14, large braking torques can cause the tyres to lose traction and the wheels to lock. The actual torque demand applied to the motor is a ramped version of the torque demand in orange.

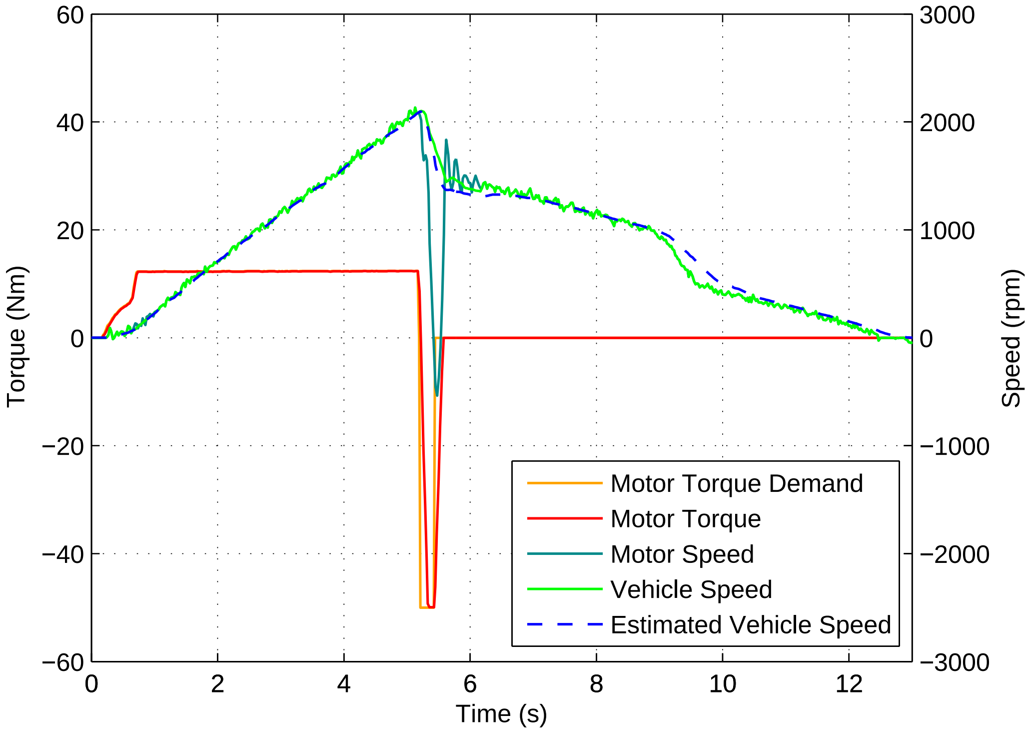

Figure 7.13: Experimental results showing the tyres losing traction with large braking torques

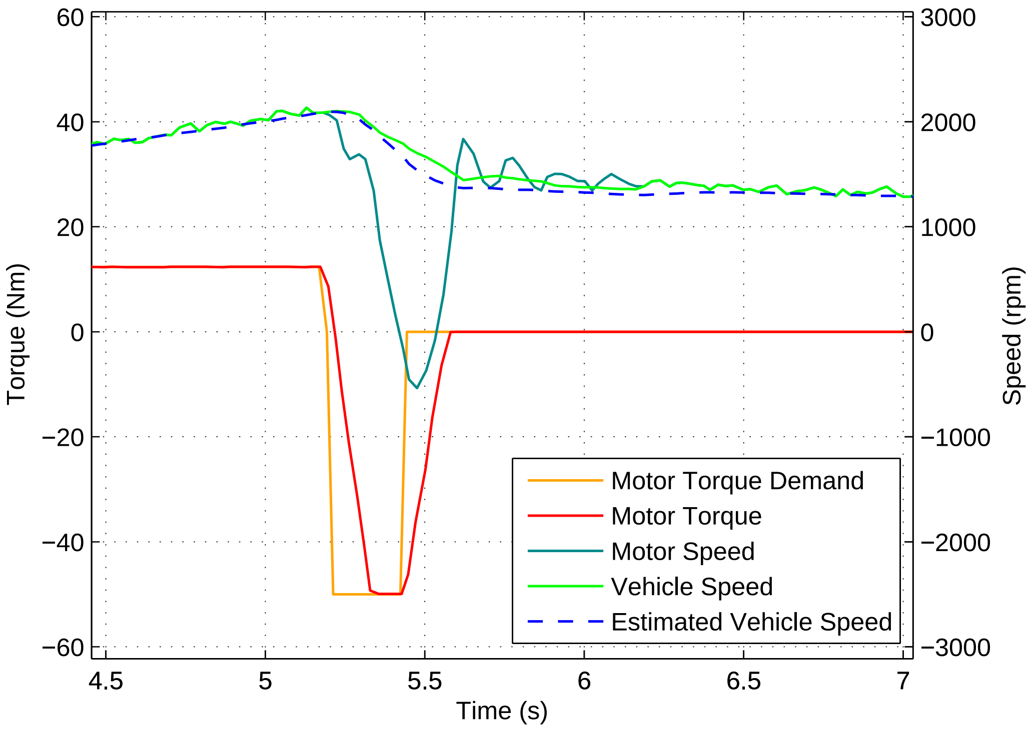

Figure 7.14: Experimental results showing the tyres losing traction with large braking torques (zoomed)

When the scheme described above is used, it is possible to control the braking torque level to prevent the complete loss of traction. When driving forwards, shown in Figs. 7.15 and 7.16, braking torque is cut back to approximately half the requested level, which prevents the motor (rear wheel) speed from decreasing rapidly. For these 2 graphs the vehicle speed comes from the accelerometer during transients and motor speeds at other times; it was carried out before the wheel speed sensors were fitted to the test vehicle.

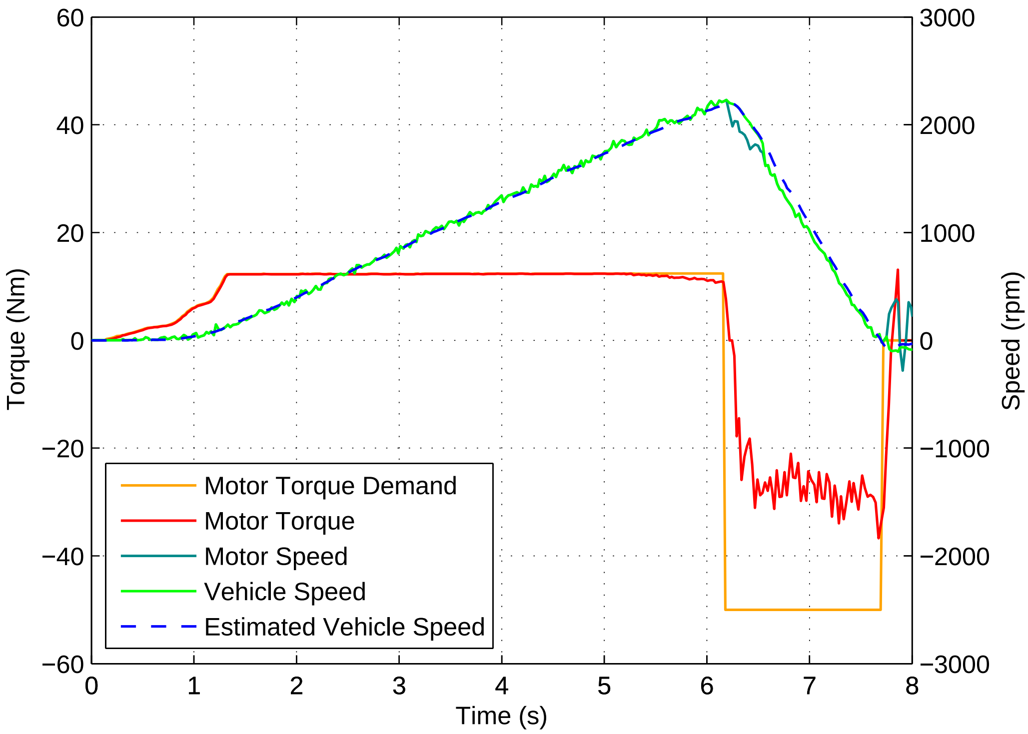

Figure 7.15: Experimental results showing traction being maintained in forward direction

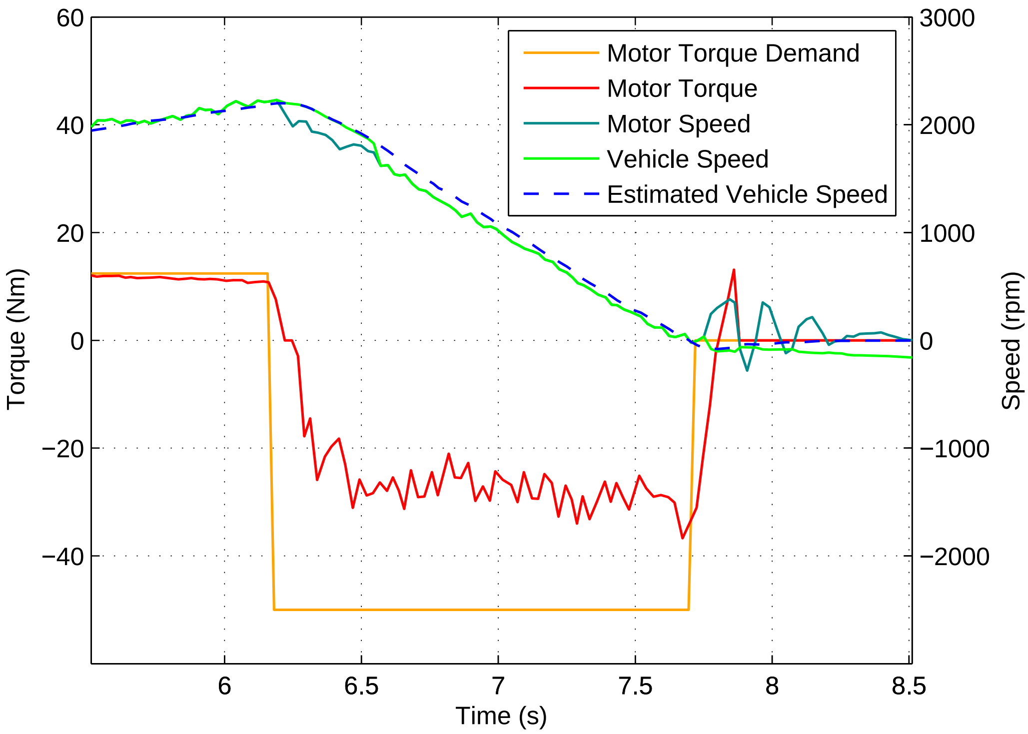

Figure 7.16: Experimental results showing traction being maintained in forward direction (zoomed)

For the following result shown in Fig. 7.17, the wheel speed sensor (from the non-driven wheel) were used for vehicle speed feedback.

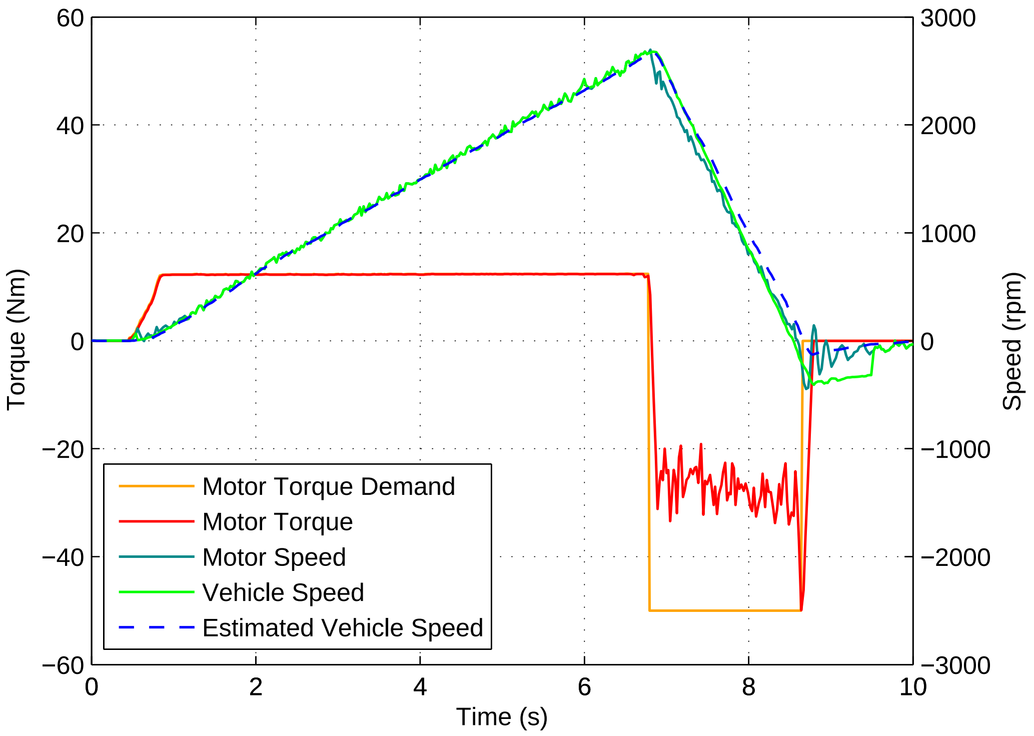

Figure 7.17: Experimental results showing traction being maintained in forward direction - 2nd result

For the reverse direction, shown in Figs. 7.18 and 7.19, as the driving wheels are now effectively at the front of the vehicle, it is equivalent to braking being applied by the front wheels (of a front wheel drive vehicle driving forwards). This allows for much higher traction than the forward direction above and it can be seen that the torque level applied is automatically higher by the algorithm. The graph also shows the torque level increasing as the vehicle slows, this is because the tyre friction coefficient increases at lower speeds, allowing more braking torques to be applied. For these 2 graphs the vehicle speed comes from the accelerometer during transients and motor speeds at other times; it was carried out before the wheel speed sensors were fitted.

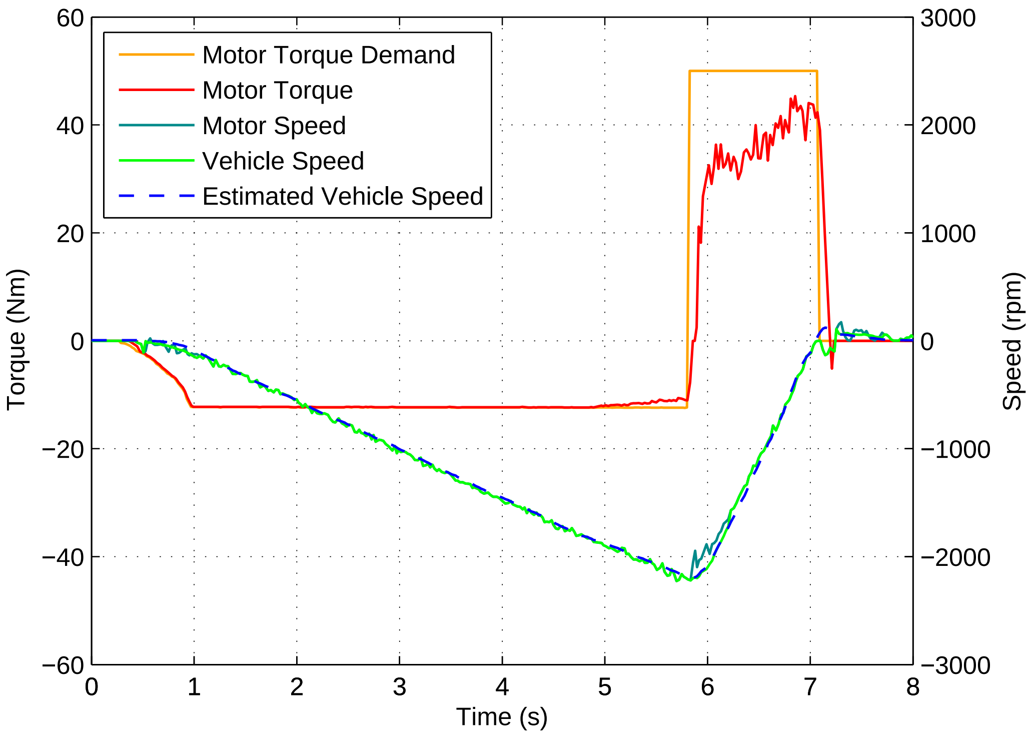

Figure 7.18: Experimental results showing traction being maintained in reverse direction

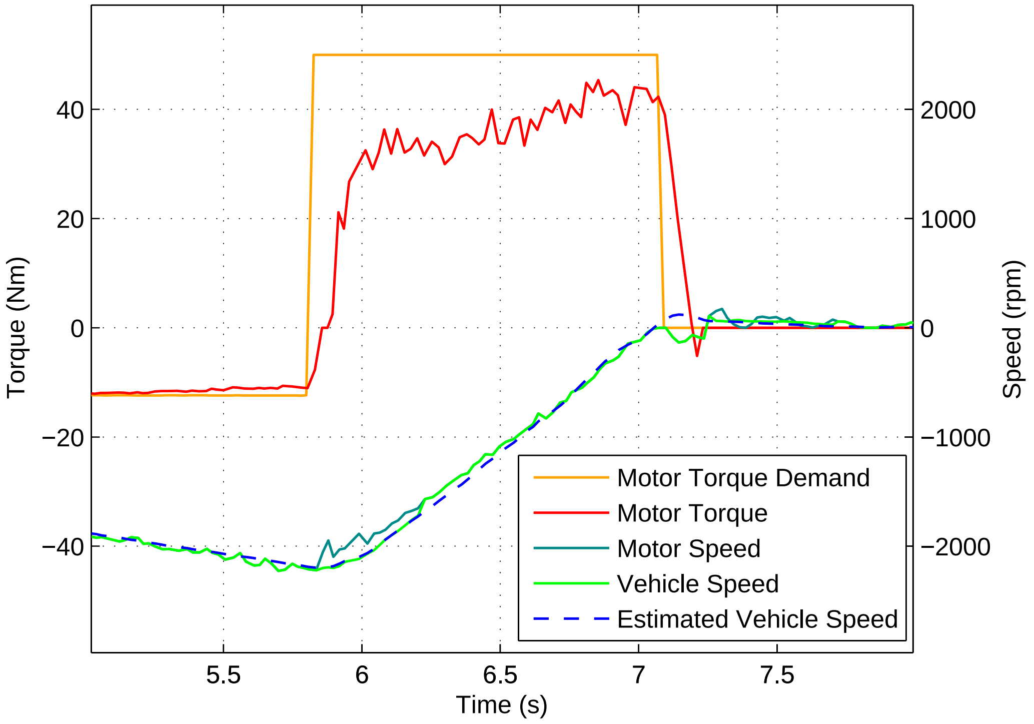

Figure 7.19: Experimental results showing traction being maintained in reverse direction (zoomed)

For the following result shown in Fig. 7.20, it shows similar performance to above, but includes measured vehicle speed from the wheel speed sensor on the non-driven wheel.

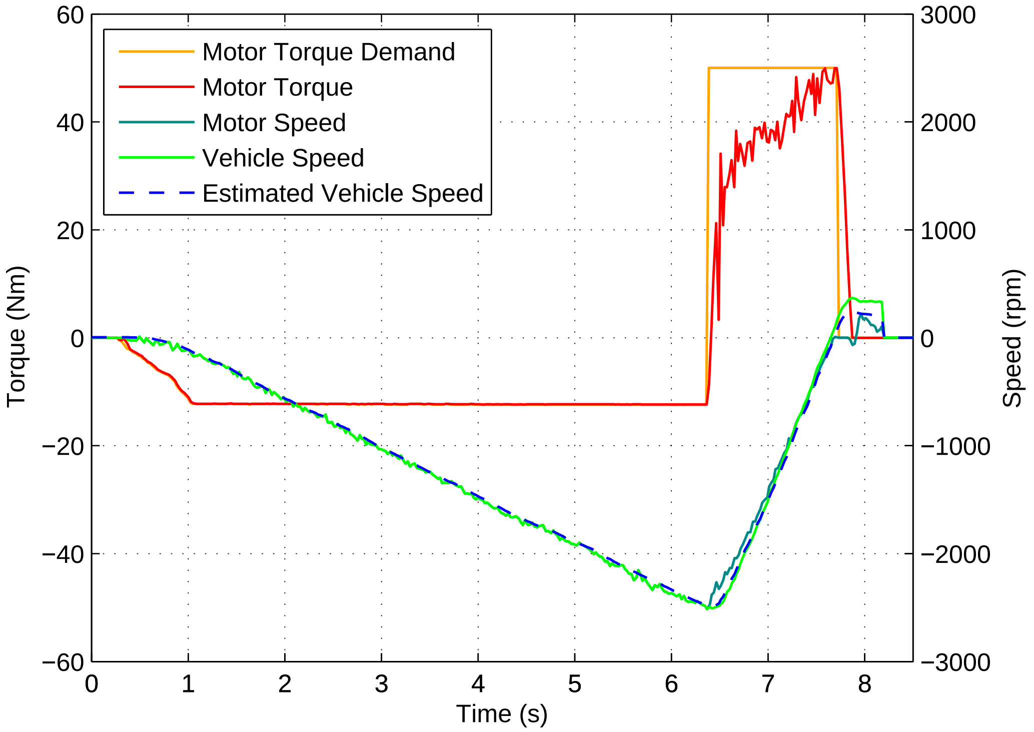

Figure 7.20: Experimental results showing traction being maintained in reverse direction - 2nd result

### 7.4.4 Discussion

A scheme has been presented that allows for preventing wheel lock up and loss of traction during braking. It does not use any additional sensors such as wheel speed sensors. Other published work either uses wheel speed sensors or accelerometers, or does not consider that vehicle mass can change. The speed estimator is compensated using the estimated vehicle mass so that it works correctly when in open loop mode. The tyre slip is not included in the estimator, but this was found to be acceptable as long as the loss of traction is prevented. This is because if traction is maintained then there is no need to estimate loss of traction conditions.

The speed estimator does not fully compensate for road gradients and more work and testing should be done to allow the algorithm to be applied to this.

## 7.5 Open Loop Backlash

### 7.5.1 Introduction

When a change in direction of motor torque is required, the gear backlash must be traversed; to allow the other side of the cogs to transfer the torque, see section 4.2.4. It is common, due to the much lower inertia of the motor compared to the vehicle, for the motor speed to increase greatly during the backlash traversal time. The torque can be limited during the reversal to reduce this effect, but this will lead to a greater time taken before contact is made on the other side of the backlash and therefore before the full requested torque can be applied which is not desirable.

If nothing is done to limit the torque during the backlash, it can lead to a large difference in velocity upon impact between the gear cogs. This impact causes all of the kinetic energy in the motor (and gearbox input shaft and cog) to be almost instantly transferred to the vehicle, causing a sudden jerk in the vehicle acceleration. This will also wear out and damage the gearbox over time. Ideally the speeds of the two sides of the gear cogs would be equal or at least almost equal at the point of contact, but normally in order to do this a very slow backlash traversal is required, or alternatively position feedback from both sides of the gearbox.

This section describes a new control scheme that allows for faster backlash traversal times and also practically zero speed difference upon impact leading to a smooth torque reversal with less jerk in acceleration or oscillation. As most of the parameters regarding the gearbox are known and fixed, such as backlash size and motor inertia, it is possible to calculate an open loop torque waveform to move the gear cog a fixed distance. However on a real system there are other effects going on that have an impact on the torque waveform required to do this accurately.

### 7.5.2 Theory and Simulation Results

The model used is the standard two-mass system that was first derived in chapter 4, shown again in Fig. 7.21.

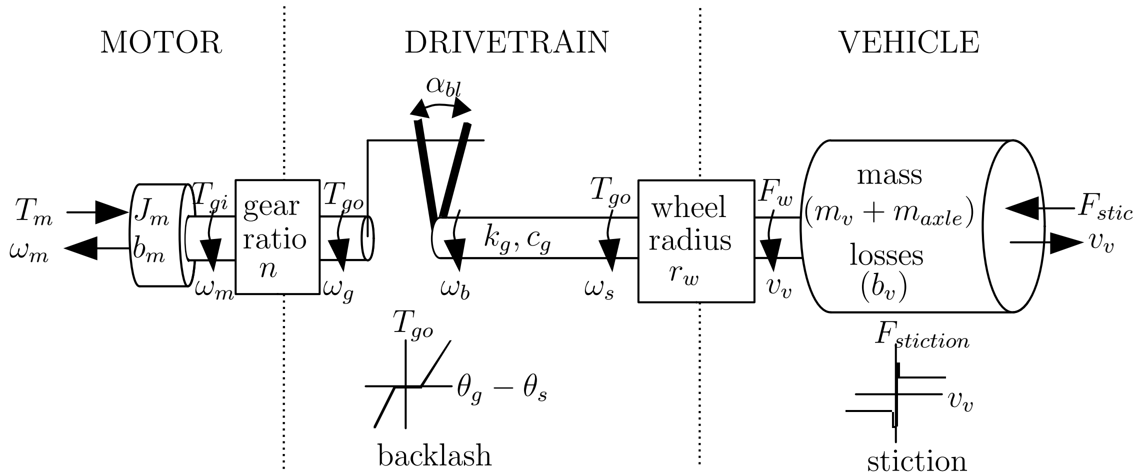

Figure 7.21: Backlash diagram - two-mass system

The gear backlash has been redrawn linear shown in Fig. 7.22. It basically shows that there is a fixed gap between the gears and for the torque to be applied in a different direction this gap needs to closed. The entire system used for this section is given in Fig. 7.21.

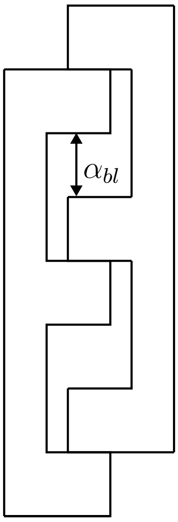

Figure 7.22: Backlash diagram - drawn linear

In order to give a fixed positive change in rotational position, assuming that the starting speed is zero, applying a square wave of positive torque and then an equal in magnitude and duration negative torque pulse, will return it to zero speed ignoring all losses. In Fig. 7.23 it is shown that a square wave in torque gives a constant acceleration and then when followed by a deceleration pulse it gives a constant deceleration, giving a triangle wave of speed. As the torque pulses have equal areas, the same amount of energy is put into accelerating as is removed when decelerating as losses are ignored.

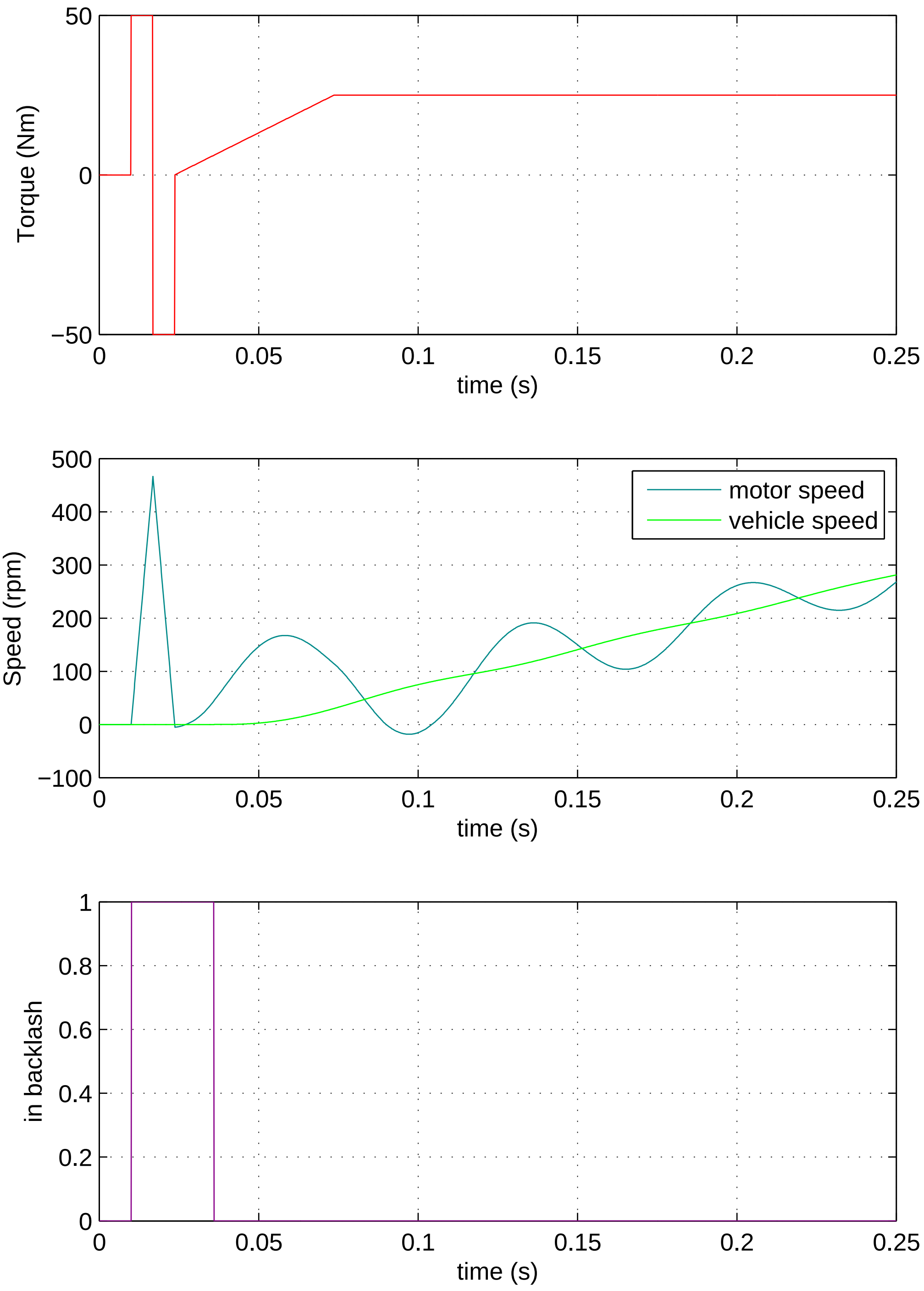

Figure 7.23: Torque waveform to accelerate and decelerate gear cog

The calculation for the required torque pulse duration $t_{on}$ shown in Fig. 7.23 is shown in equation (7.1) for a given backlash size $\alpha_{bl}$, motor and gearbox input inertia $J_m$ and motor torque $T_m$. This equation was derived by rearranging the area of a triangle for the speed graph shown in Fig. 7.23.

$$t_{on} = \sqrt{\frac{\alpha_{bl} \times J_m}{T_m}} \tag{7.1}$$

In reality though, this would require an instant change in motor torque and therefore current; not possible due to inductance and response times of the current control. An alternative is to ramp on the torque and off again in one direction, then repeat this in the opposite direction, shown in Fig. 7.24.

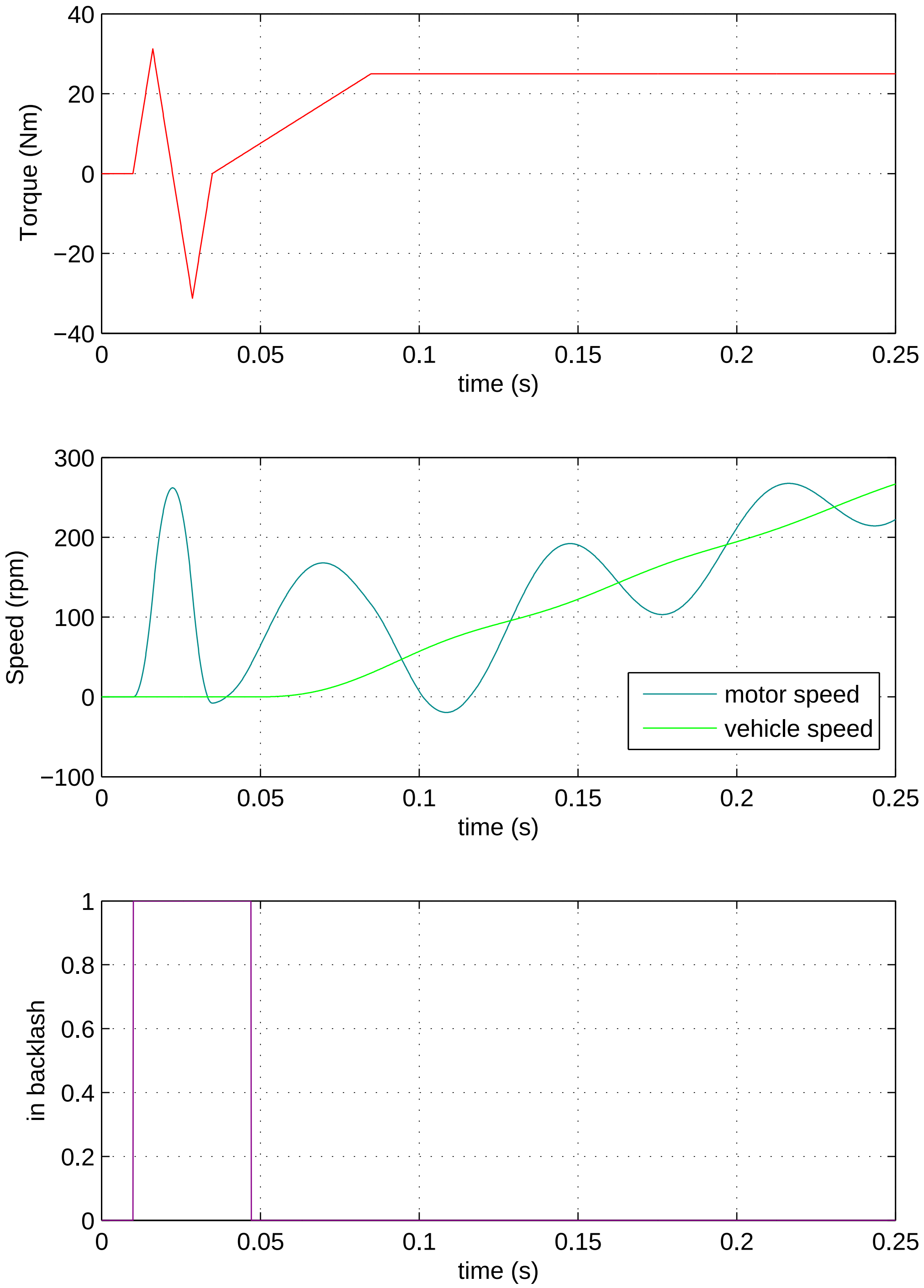

Figure 7.24: Torque waveform to accelerate and decelerate gear cog

The calculation becomes slightly more difficult in this case, as the speed graph is no longer a simple triangle waveform. As the system is a simple single inertia model the speed graph can be approximated to a triangle as equal areas are below this approximation (triangle)in the first quarter as are exceeding the triangle in the second quarter, shown in Fig. 7.24. In order to calculate the required on-time for the positive triangle wave of torque is given in equation (7.2), which is the same as (7.1) apart from it is equivalent to double the backlash angle due to less torque being applied for the same time.

$$t_{on} = \sqrt{\frac{2 \times \alpha_{bl} \times J_m}{T_{m-peak}}} \tag{7.2}$$

The above equation assumes that the peak torque is known, but this will depend on the torque ramp rate $T_{m-rate-limit}$ and the on-time $t_{on}$. Ideally the required on time needs to be calculated without knowledge of the final peak torque, but using the torque rate limit as in equation (7.3).

$$t_{on} = \sqrt[3]{\frac{\alpha_{bl} \times J_m}{2 \times T_{m-rate-limit}}} \tag{7.3}$$

As the torque can only be changed every $125\mu s$, it is more useful to have the on-time in steps and this can be calculate in (7.4), rounding down to the nearest whole step.

$$s_{on} = \frac{t_{on}}{125 \times 10^{-6}} \tag{7.4}$$

The two methods above assume that both sides of the cog are stationary at the start and end of the sequence. This is valid if they are both at the same speed at the start and end of the applied torque waveform, this can be assumed if there is no friction losses or vehicle speed change throughout the sequence. As the motor friction losses are very small compared to the torques being applied and the time the algorithm is being applied over is so short, the vehicle speed wont change significantly.

### 7.5.3 Theory and Simulation Results - axle energy

If a torque is being applied prior to changing the backlash side required, as is typical in most situations, there is energy stored within the rotational flex of the axle. This can be calculated using (7.5):

$$E_{axle} = 0.5 \times k_g \times (\theta_g - \theta_s - \theta_{bl})^2 \tag{7.5}$$

The gear position $\theta_g$ is calculated from motor position $\theta_m$ using (7.6):

$$\theta_g = \frac{\theta_m}{n} \tag{7.6}$$

Ideally the energy can be calculated directly from the torque being applied, as for a given torque will always be the same amount of axle flex, in steady state conditions (7.7):

$$E_{axle} = 0.5 \times \frac{1}{k_g} \times (T_m \times n)^2 \tag{7.7}$$

For a range of torque levels the calculated stored energy is, increasing with the square of the torque level:

- For $10Nm$ gives an energy of $0.8J$
- For $25Nm$ gives an energy of $5.2J$
- For $50Nm$ gives an energy of $20.7J$

When the motor torque is removed suddenly, this energy is released with equal torques being applied to both sides of the axle. The large difference in inertias though means that the motor is accelerated to a much larger speed change than the vehicle. If there was no damping within the axle all of the energy would be transferred into both the vehicle and motor inertias, but the damping reduces the energy that is recovered. It also requires that the applied motor torque is removed faster than the axle ‘unwinds’, else the acceleration effect is lost. It is quite difficult to solve how much energy will be recovered, so the estimator derived in the previous chapter and used above in the previous two sections is used to give this information with no additional calculation required. The estimator also gives the indication as to when the backlash is entered.

The following parameters are used for the backlash algorithm:

$s$ = step count

$T_s$ = sample time ($125\mu s$)

$accel$ = accelerate - the first torque pulse that increases the speed difference between the gear cogs

$decel$ = decelerate - the second torque pulse that matches the speeds between the gear cogs

$T_{m-rate-limit}$ = torque ramp rate limit, in this case $100 \times T_{pk}/sec = 5000Nm/s$

$\omega_{diff\,from\,est\,k+1}$ = speed difference from the estimator at instant when backlash is entered

$\theta_{diff\,from\,est\,k+1}$ = position difference from the estimator at instant when backlash is entered

The algorithm is in the following calculation steps as follows:

**Step 1** - Ramp off the previous torque demand at the $5000Nm/s$ rate.

**Step 2** - Start to ramp the torque in the other direction at the same rate and increment step counts in (7.8) every time step and convert to time in (7.9):

$$s_{accel\,k+1} = s_{accel\,k} + 1 \tag{7.8}$$

$$t_{accel\,k+1} = s_{accel\,k+1} \times T_s \tag{7.9}$$

**Step 3** - Calculate the extra speed contribution from the acceleration pulse currently being applied, in (7.10).

$$\omega_{accel\,pulse\,k+1} = {t_{accel\,k+1}}^2 \times T_{m-rate-limit} \times \frac{1}{J_m} \tag{7.10}$$

**Step 4** - Calculated the speed contribution from just the energy released from the axle in (7.11), the equation part $\omega_{diff\,from\,est\,k+1}$ can only be updated until the backlash is entered:

$$\omega_{axle\,energy\,k+1} = \omega_{diff\,from\,est\,k+1} - (0.5 \times \omega_{accel\,pulse\,k+1}) \tag{7.11}$$

**Step 5** - Can now calculate the total peak speed difference in (7.12):

$$\omega_{pk\,diff\,k+1} = \omega_{accel\,pulse\,k+1} + \omega_{axle\,energy\,k+1} \tag{7.12}$$

**Step 6** - The number of steps required to decelerate from the peak speed difference above to give a zero impact speed is calculated (7.13), the number of steps are rounded down, as it is better to have a small impact than stop short of closing the backlash - this adds greatly to the backlash traversal time:

$$s_{decel\,k+1} = \frac{1}{T_s}\sqrt{\omega_{pk\,diff\,k+1} \times J_m \times \frac{1}{T_{m-rate-limit}}} \tag{7.13}$$

**Step 7** - The steps are converted to a time (7.14):

$$t_{decel\,k+1} = s_{decel\,k+1} \times T_s \tag{7.14}$$

**Step 8** - The speed difference from the deceleration pulse is then calculated (7.15):

$$\omega_{decel\,pulse\,k+1} = {t_{decel\,k+1}}^2 \times T_{m-rate-limit} \times \frac{1}{J_m} \tag{7.15}$$

**Step 9** - The position change from the acceleration torque pulse (7.16):

$$\begin{aligned}\theta_{accel\,k+1} = &\left(2 \times \omega_{axle\,energy\,k+1} \times t_{accel\,k+1}\right)\\ &+ \left(\omega_{accel\,pulse\,k+1}\left(t_{accel\,k+1} - \frac{T_s}{2}\right)\right)\end{aligned} \tag{7.16}$$

**Step 10** - The position changes from the deceleration torque pulse (7.17):

$$\begin{aligned}\theta_{decel\,k+1} = &\left(2\left(\omega_{pk\,diff\,k+1} - \omega_{decel\,pulse\,k+1}\right) \times t_{decel\,k+1}\right)\\ &+ \left(\omega_{decel\,pulse\,k+1}\left(t_{decel\,k+1} - \frac{T_s}{2}\right)\right)\end{aligned} \tag{7.17}$$

**Step 11** - The total angle movement (7.18):

$$\theta_{total\,k+1} = \theta_{accel\,k+1} + \theta_{decel\,k+1} \tag{7.18}$$

**Step 12** - This can be checked against the total backlash size, taking into account if the backlash was entered before the shaft had unwound, as this requires the gap to crossed to be larger (7.19):

$$IF\ \theta_{total\,k+1} > \left(\alpha_{bl} + \theta_{diff\,from\,est}\right) \tag{7.19}$$

**Step 13** - After all these calculations if the condition above (7.19) becomes true, the acceleration pulse is stopped being incremented and then ramped down, followed by the calculated braking pulse.

Although the algorithm has a lot of calculations, it is not required to run every $125\mu s$, and the prediction can calculate more steps ahead whilst running less frequently.

The new algorithm was simulated to compare it to the previous response. For a $10Nm$ torque reversal previously with a slow $500Nm/sec$ ramp rate there was a number of oscillations with the backlash entered repeatedly, shown in Fig. 7.25 and 7.26. The new method shown in Fig. 7.27 and Fig. 7.28, still shows some oscillations, but these are greatly reduced and the gearbox does not repeatedly jump into and out of the backlash, this will give a smoother vehicle acceleration. The results here have no other damping algorithm applied, so the subsequent oscillations after the backlash could be reduced using either of the previous schemes in this chapter.

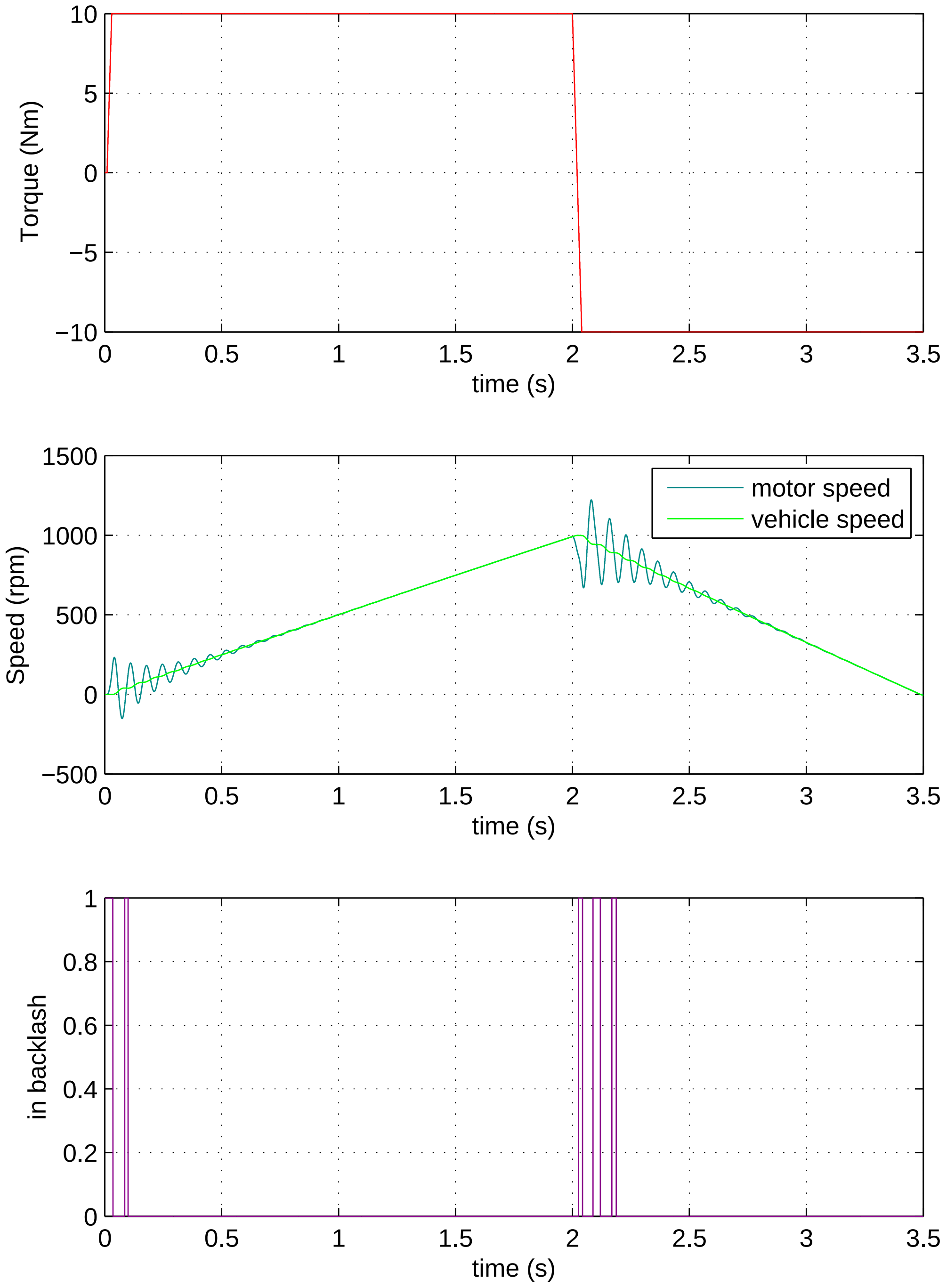

Figure 7.25: Torque direction change normally with $10Nm$ starting torque

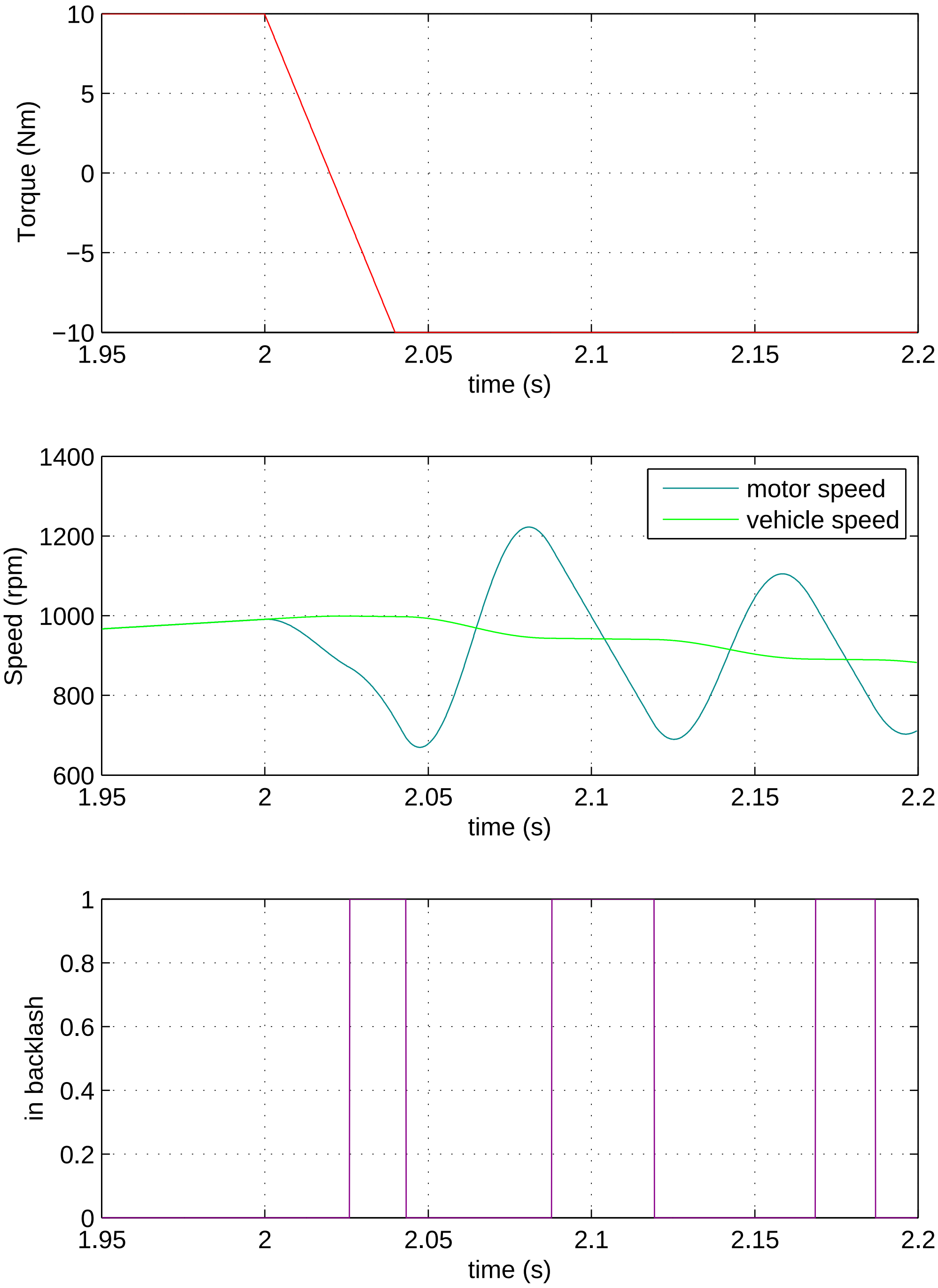

Figure 7.26: Torque direction change normally with $10Nm$ starting torque - zoomed

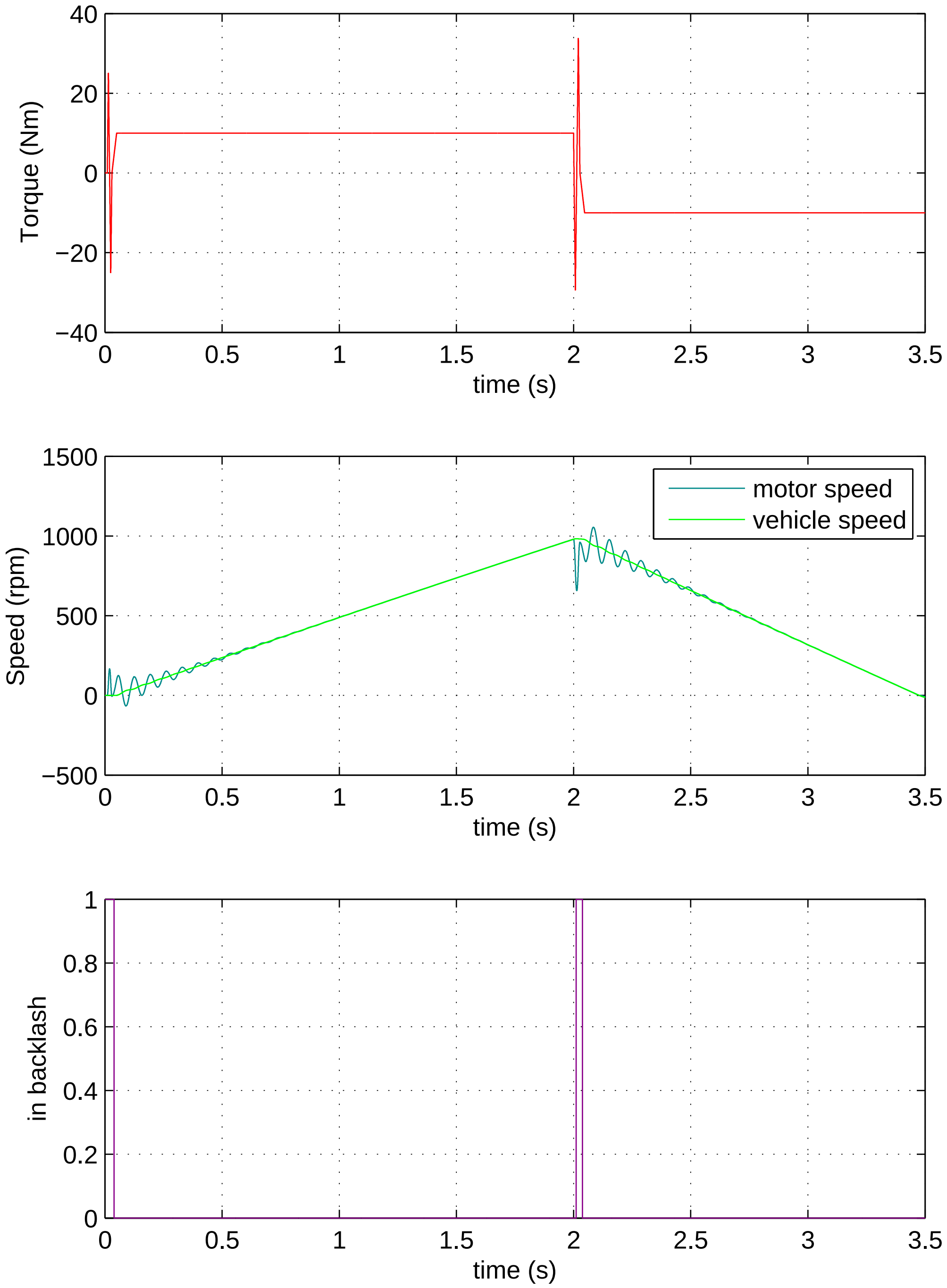

Figure 7.27: Torque waveform to accelerate and decelerate gear cog with $10Nm$ starting torque

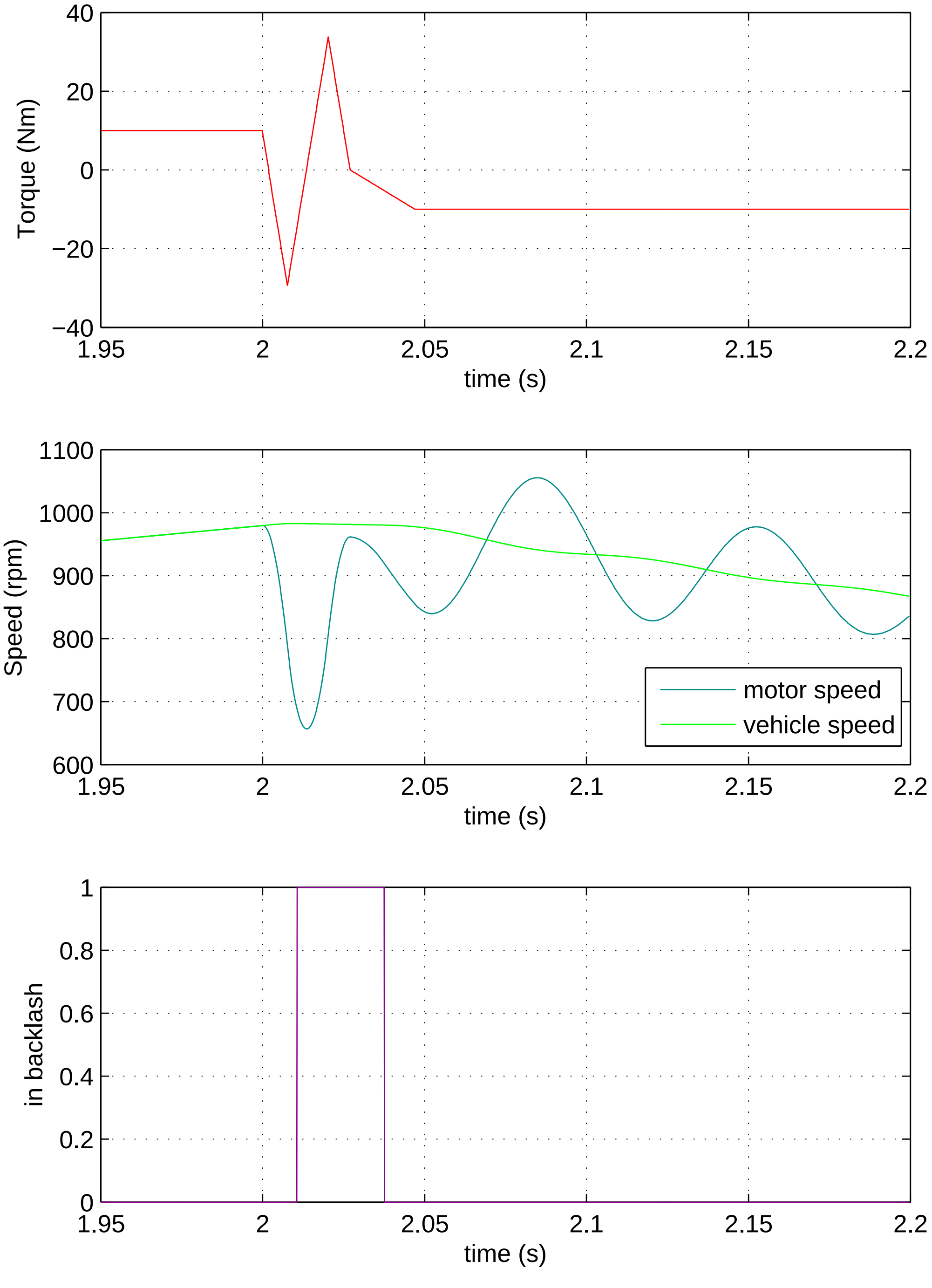

Figure 7.28: Torque waveform to accelerate and decelerate gear cog with $10Nm$ starting torque - zoomed

For a $25Nm$ torque reversal the normal response shown in Fig. 7.29 and Fig. 7.30 does not switch in and out of backlash, as the higher torque level applied will prevent this. The difference though between the magnitude of the oscillations though for the normal response and the one with the backlash scheme shown in Fig. 7.31 and Fig. 7.32 is quite significant.

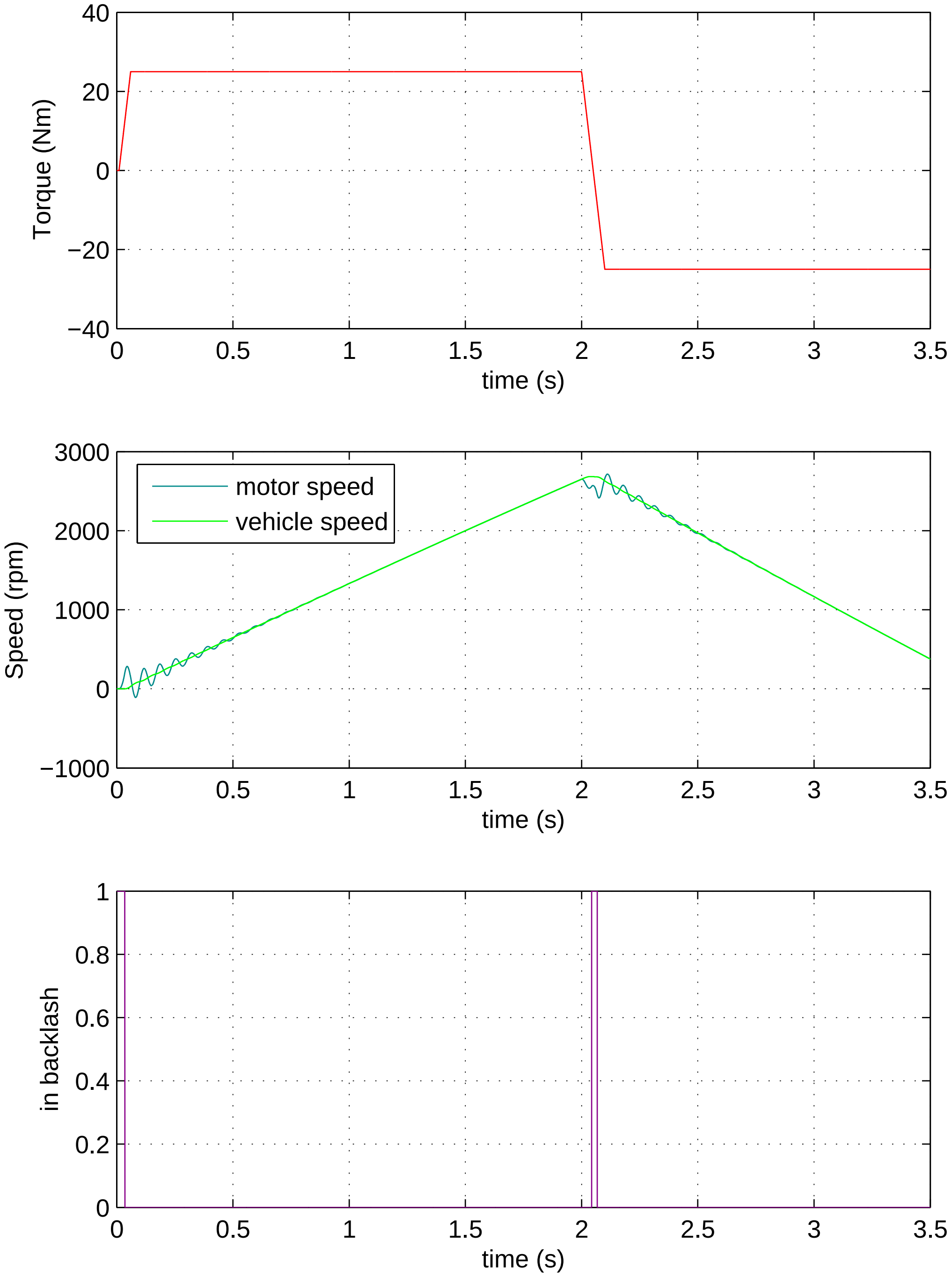

Figure 7.29: Torque direction change normally with $25Nm$ starting torque

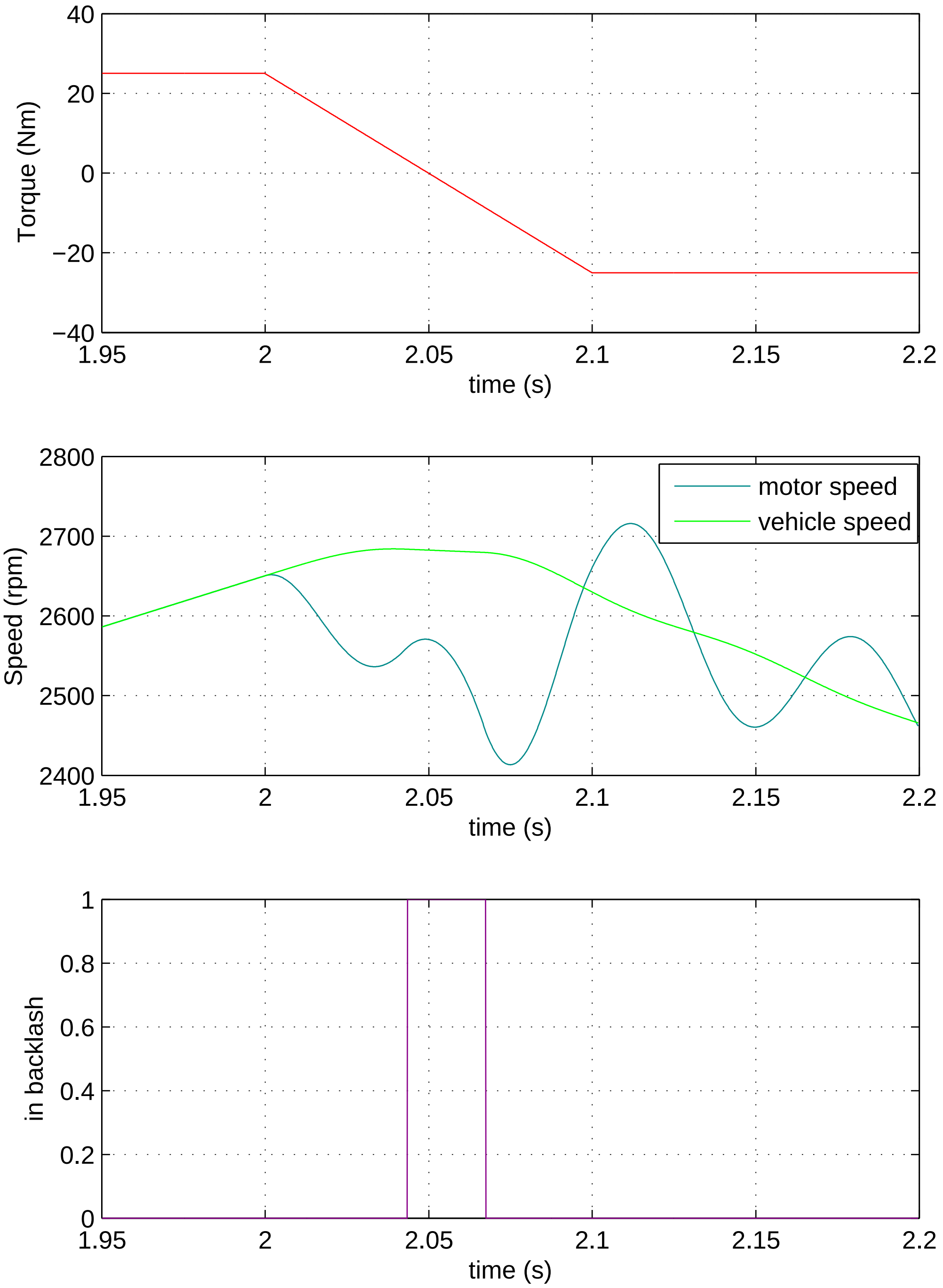

Figure 7.30: Torque direction change normally with $25Nm$ starting torque - zoomed

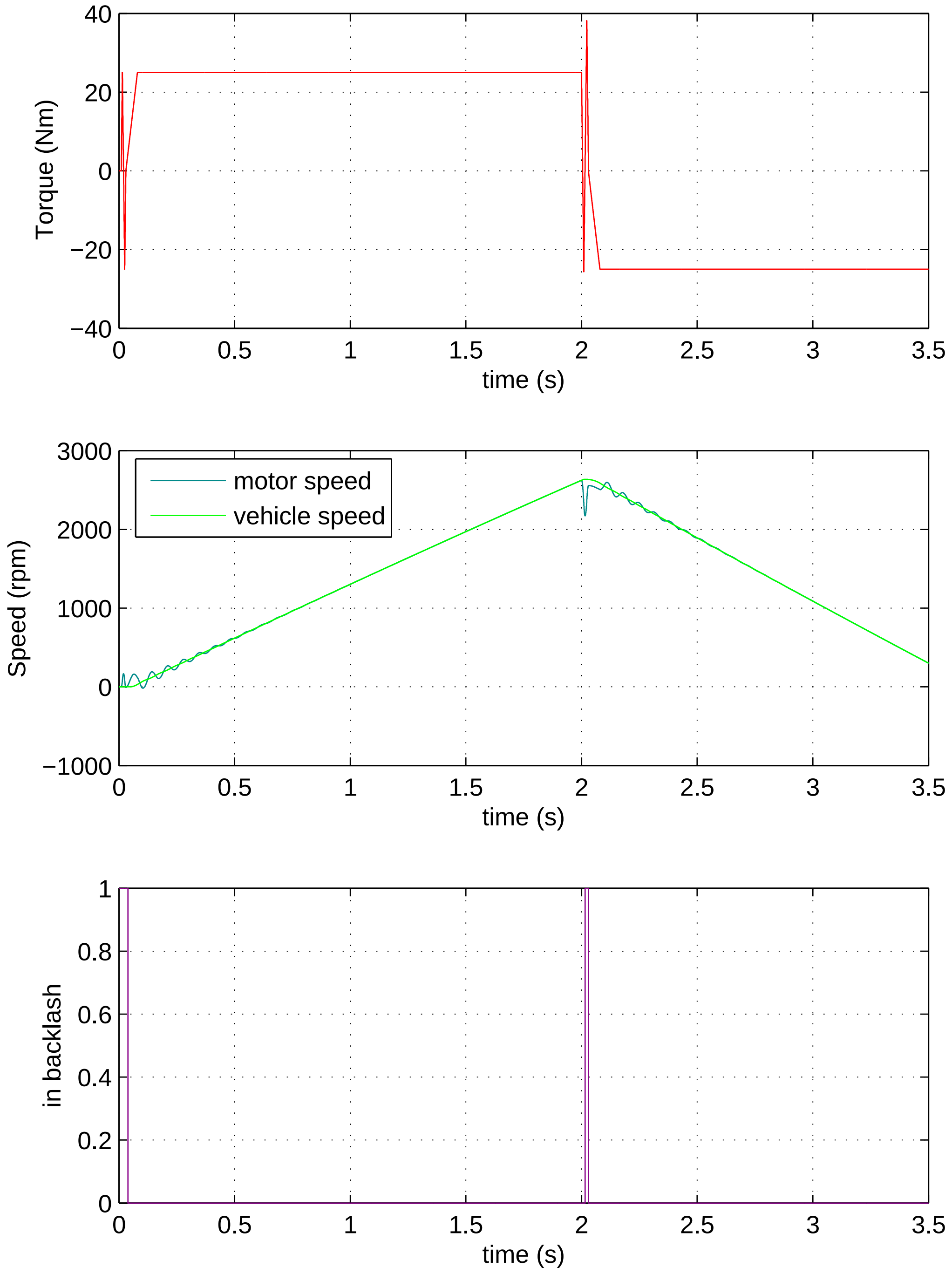

Figure 7.31: Torque waveform to accelerate and decelerate gear cog with 25$Nm$ starting torque

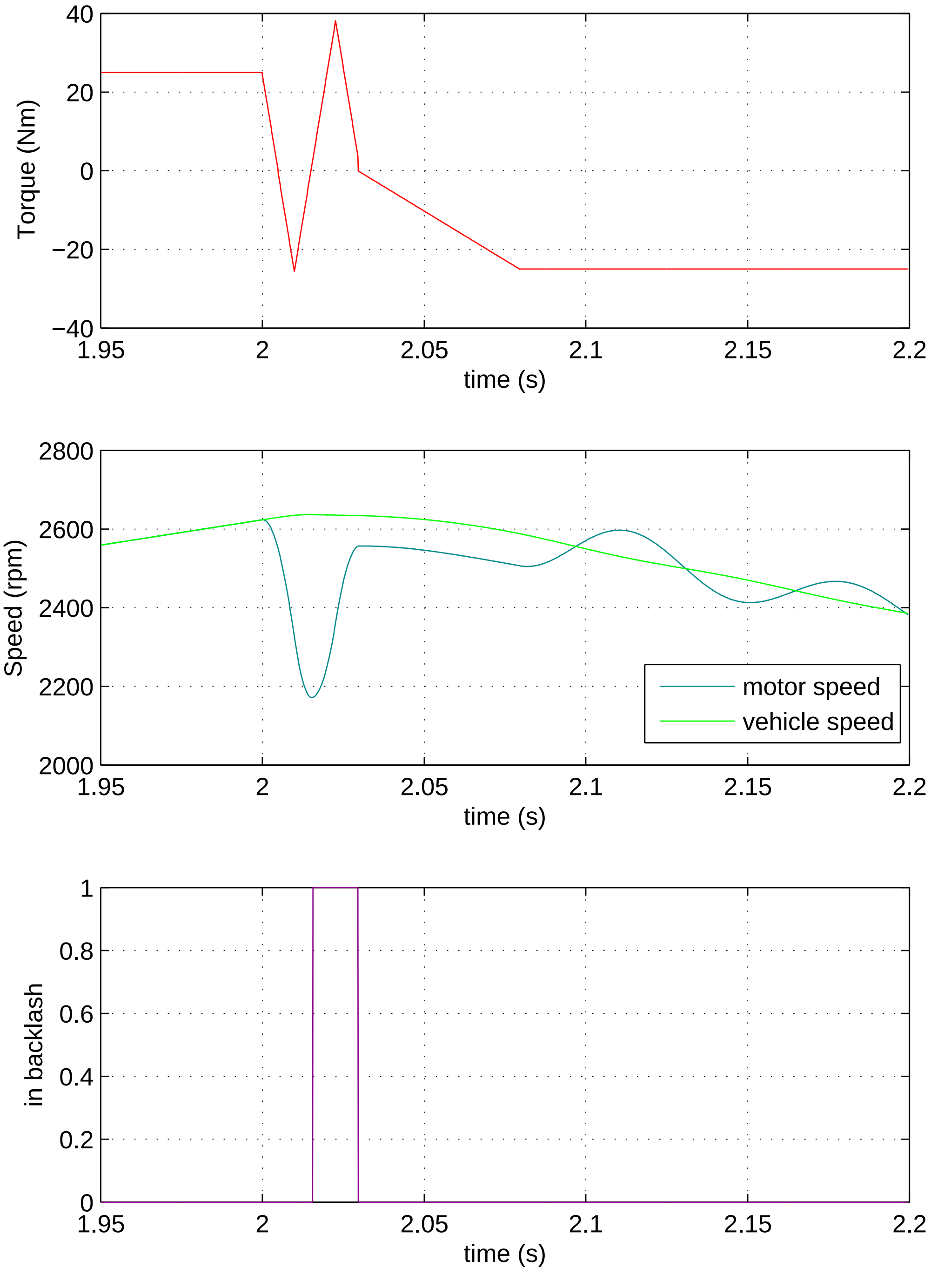

Figure 7.32: Torque waveform to accelerate and decelerate gear cog with $25Nm$ starting torque - zoomed

For a $50Nm$ torque reversal the normal response shown in Fig. 7.33 and Fig. 7.34 is quite poor, but with backlash algorithm there is not that much improvement shown in Fig. 7.35 and Fig. 7.36. This is because the backlash exit time has been miscalculated and the full braking torque pulse was not applied.

There are two important events in the algorithm, when the backlash is entered and when the acceleration torque is stopped being increased and is started to be decremented and then reversed. The algorithm assumes that the backlash is entered before the acceleration torque is stopped increasing, as the amount of speed contribution from the estimator is not known until the backlash is entered and it is required to calculated the braking pulse required. In order to get around this the speed contribution from axle energy should be predicted ahead of what the estimator is saying. In this example the speed difference between both side of the gearbox was $667rpm$ (referenced to the motor after taking into account gear ratio) before the backlash is even entered.

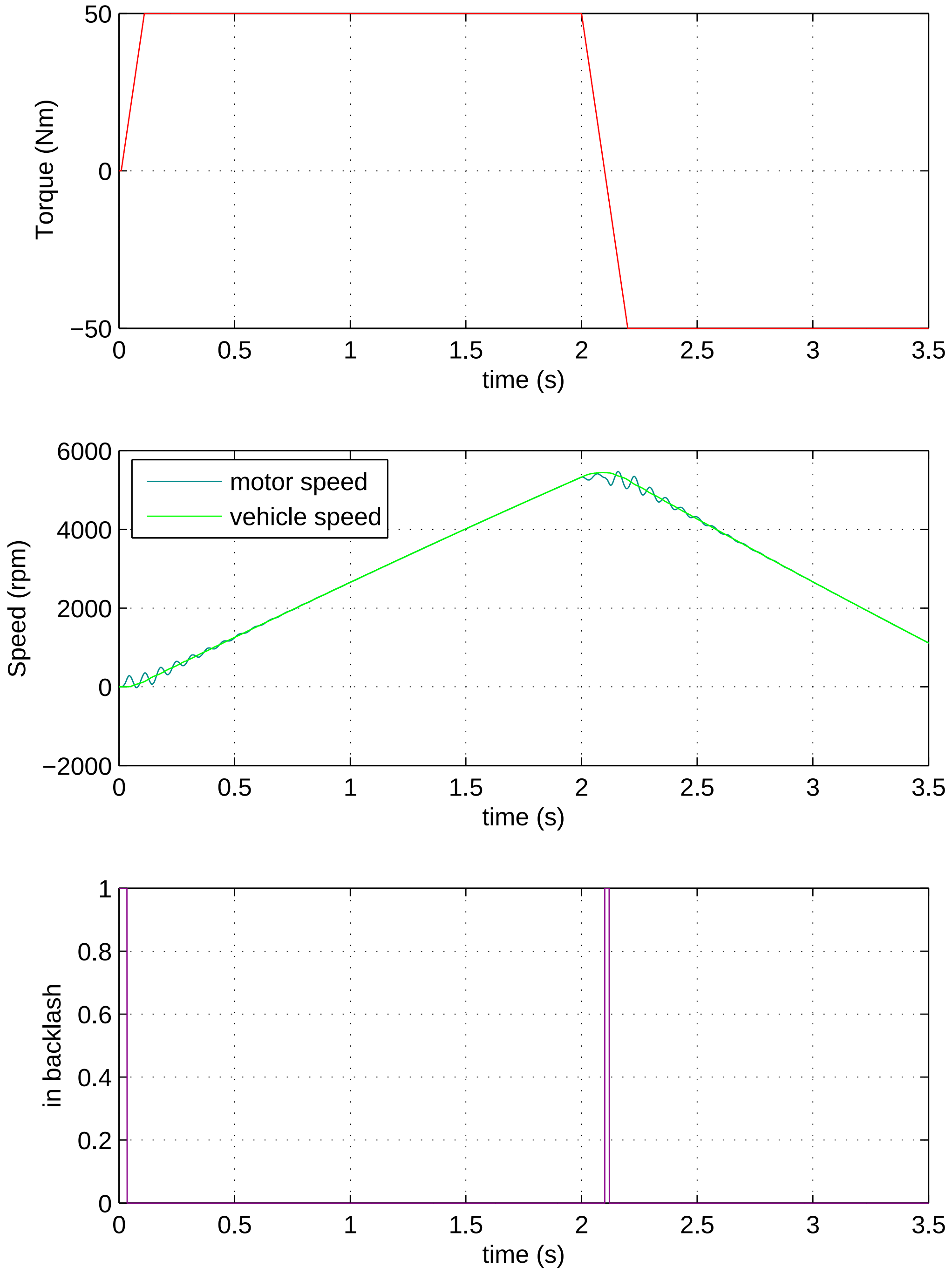

Figure 7.33: Torque direction change normally with $50Nm$ starting torque

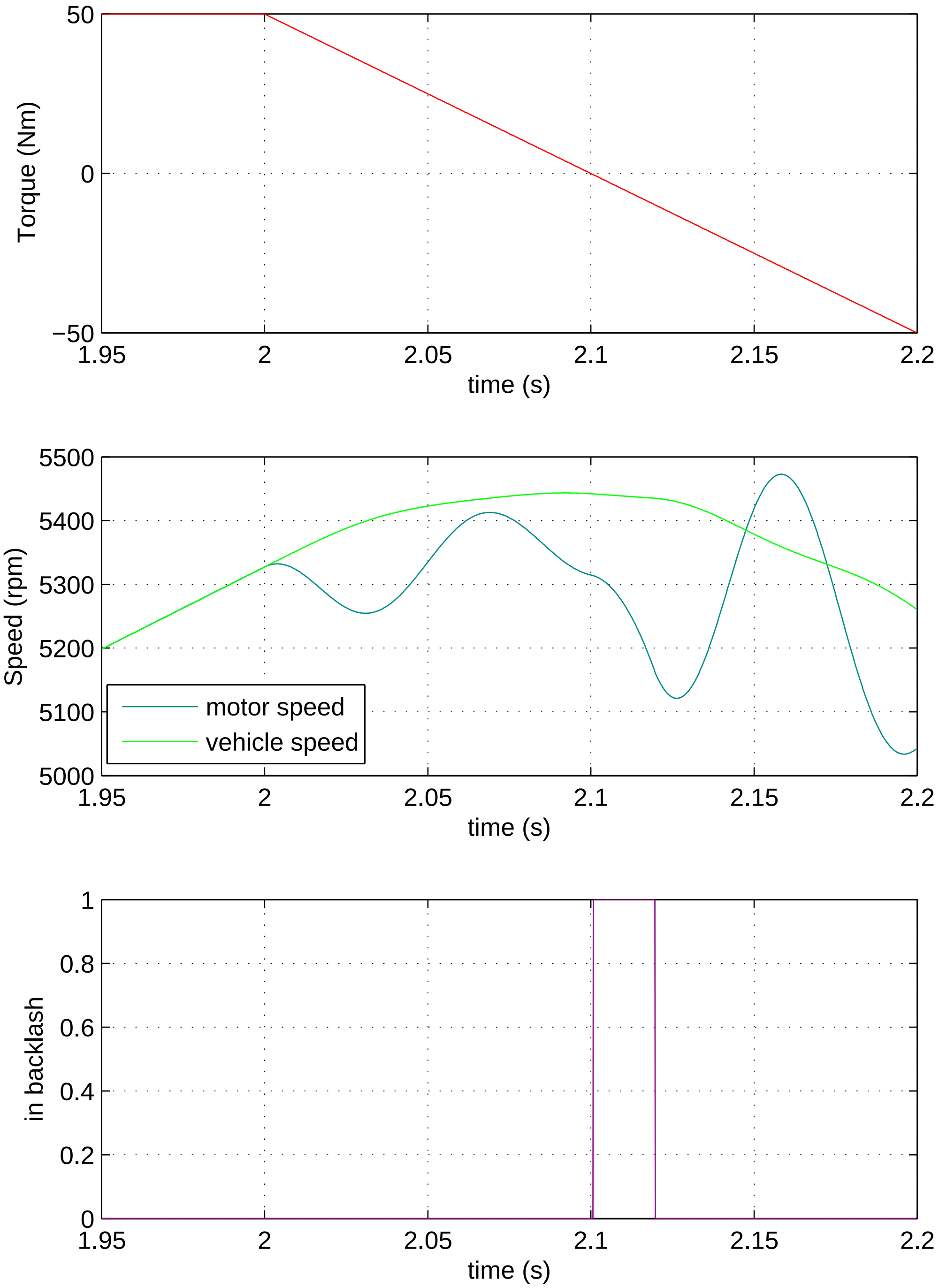

Figure 7.34: Torque direction change normally with 50$Nm$ starting torque - zoomed

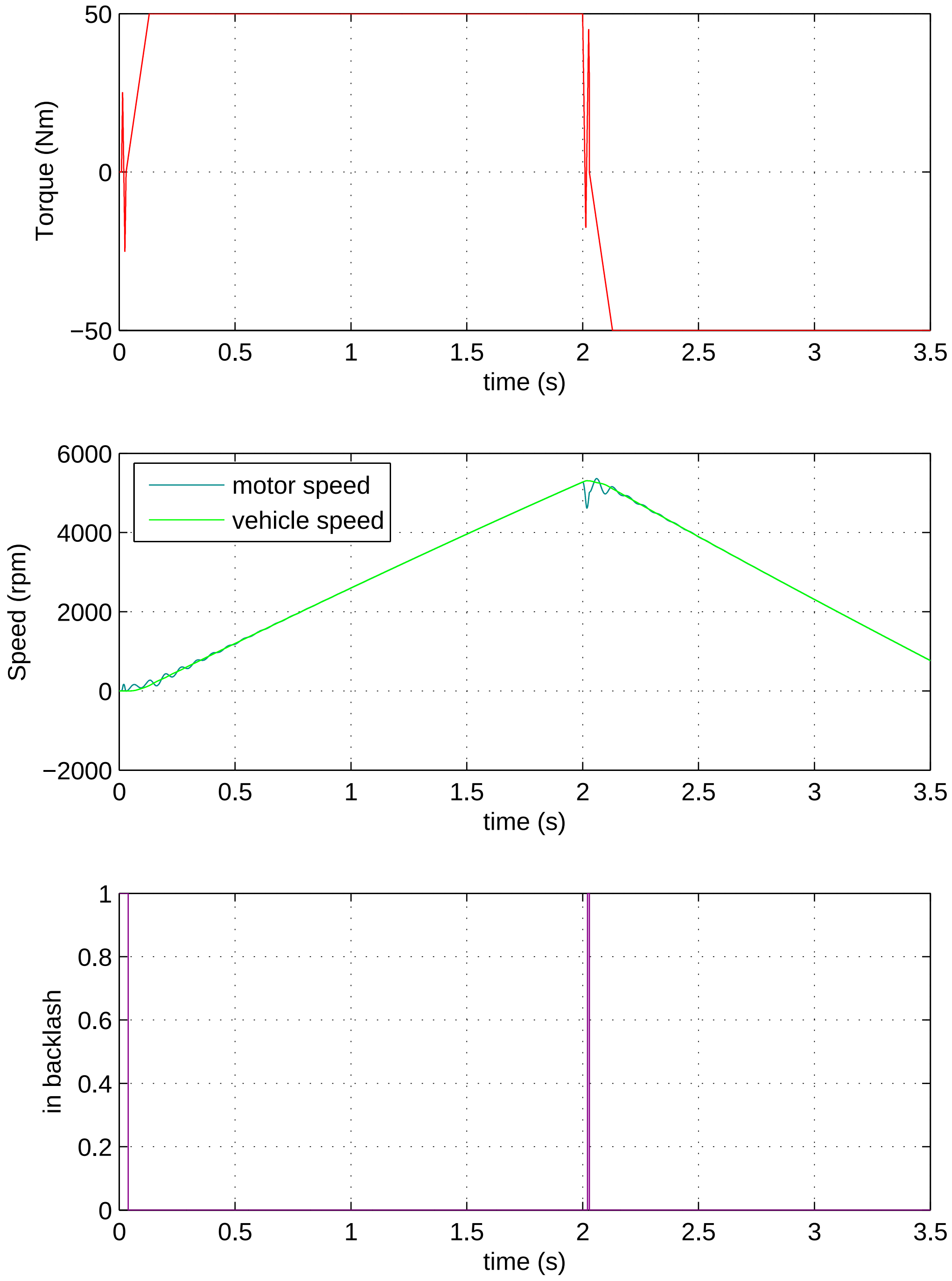

Figure 7.35: Torque waveform to accelerate and decelerate gear cog with $50Nm$ starting torque

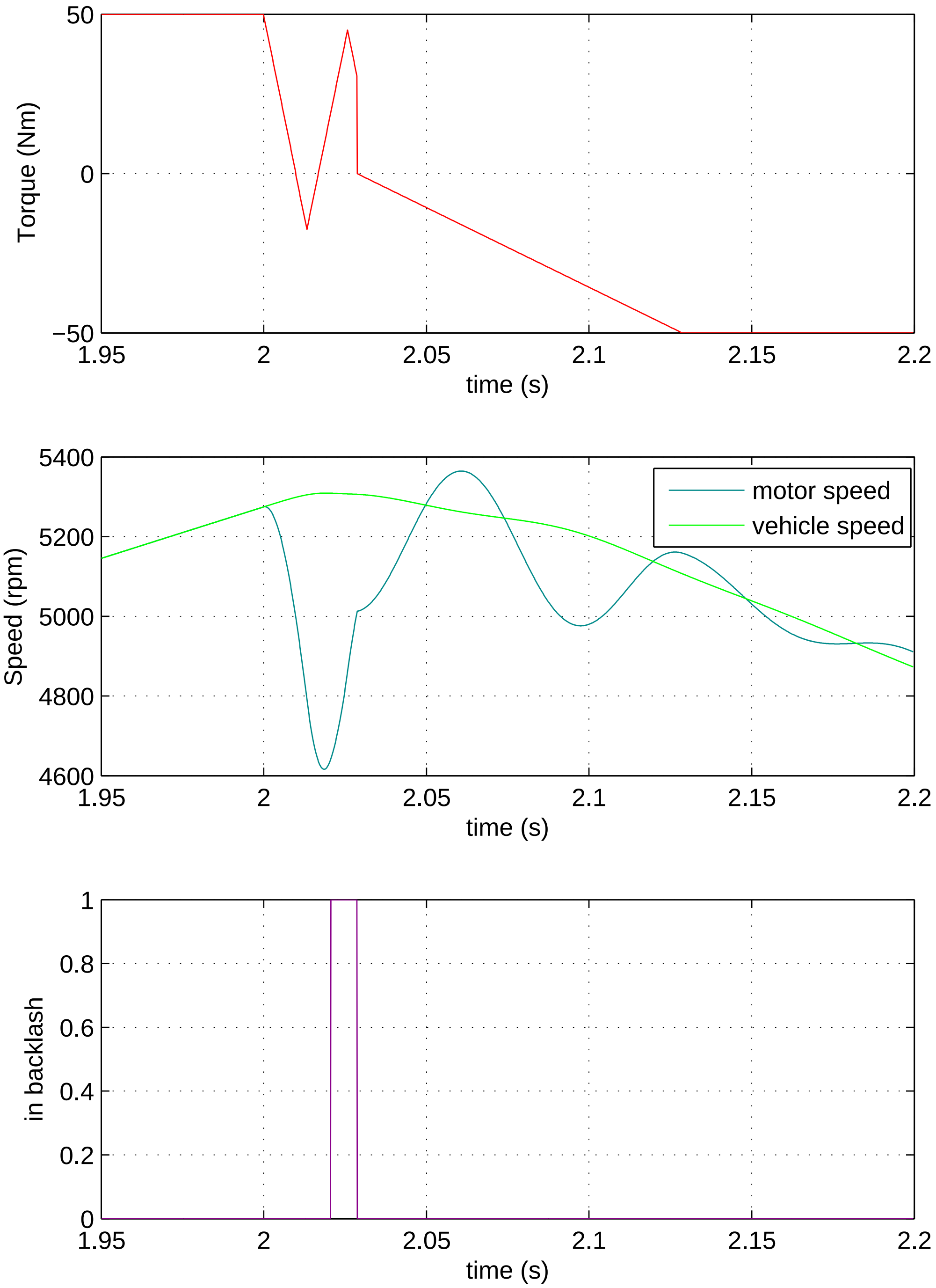

Figure 7.36: Torque waveform to accelerate and decelerate gear cog with $50Nm$ starting torque - zoomed

In table 7.1, where normal performance with torques ramped at $500Nm/s$ compared to the Open Loop Backlash (OLBL) scheme with torque changes at $5000Nm/s$, the three important things to compare are the speed difference if gear cogs when the backlash closes, the time spent in the backlash (BL), the time taken from when the torque crosses zero until the backlash (BL) is closed and rate of change in vehicle acceleration when that backlash is closed:

| | **Typical** $10Nm$ | **OLBL** $10Nm$ | **Typical** $25Nm$ | **OLBL** $25Nm$ | **Typical** $50Nm$ | **OLBL** $50Nm$ |
|---|---|---|---|---|---|---|
| **Impact speed** ($rad\,s^{-1}$) | 13.01 | 6.45 | 23.67 | 8.10 | 28.66 | 30.71 |
| **Time in BL** ($sec$) | 0.0184 | 0.0270 | 0.0241 | 0.0140 | 0.0190 | 0.0082 |
| **Time to BL close** ($sec$) | 0.0167 | 0.0357 | 0.0176 | 0.0247 | 0.0198 | 0.0189 |
| **Vehicle jerk** ($ms^{-3}$) | 917 | 454 | 1668 | 571 | 2692 | 4322 |

Table 7.1: Backlash scheme results comparison

The table confirms what is seen in the graphed results above, for the $10Nm$ and $25Nm$ case the impact speed and subsequent vehicle jerk is reduced for the new scheme and the time taken to cross the backlash is comparable. The $50Nm$ has a much higher vehicle jerk of $4322ms^{-3}$ which obviously is not desirable.

The algorithm can be summarised in the following flow diagram shown in Fig. 7.37:

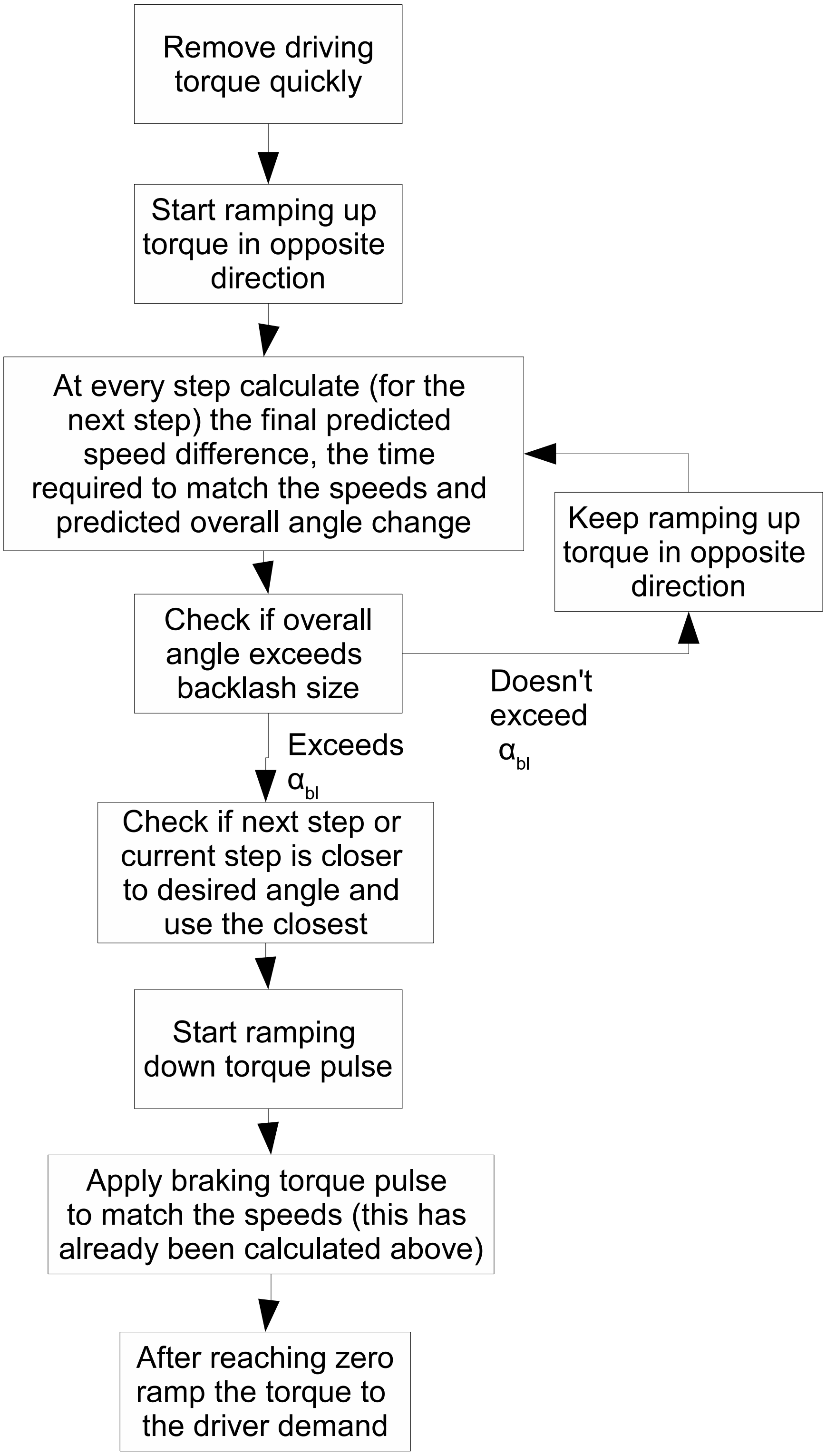

Figure 7.37: Backlash scheme flow chart

### 7.5.4 Experimental Results

The open loop backlash scheme was implemented on the test vehicle, the results show the torque as a quadrature axis current, where $315A$ equates to $50Nm$ (full torque). There is an example for 25% torque ($12.5Nm$) shown in Fig. 7.38 and 50% ($25Nm$) torque shown in Fig. 7.39. In both cases the current is able to be ramped at the fast required rate of $31,500A/sec$ and remain in control. The highest speed for these results though was $1700rpm$, where field weakening is not being applied. If the speed was much higher ramping the current so fast with less voltage headroom would be more difficult and a slower ramp rate would be required.

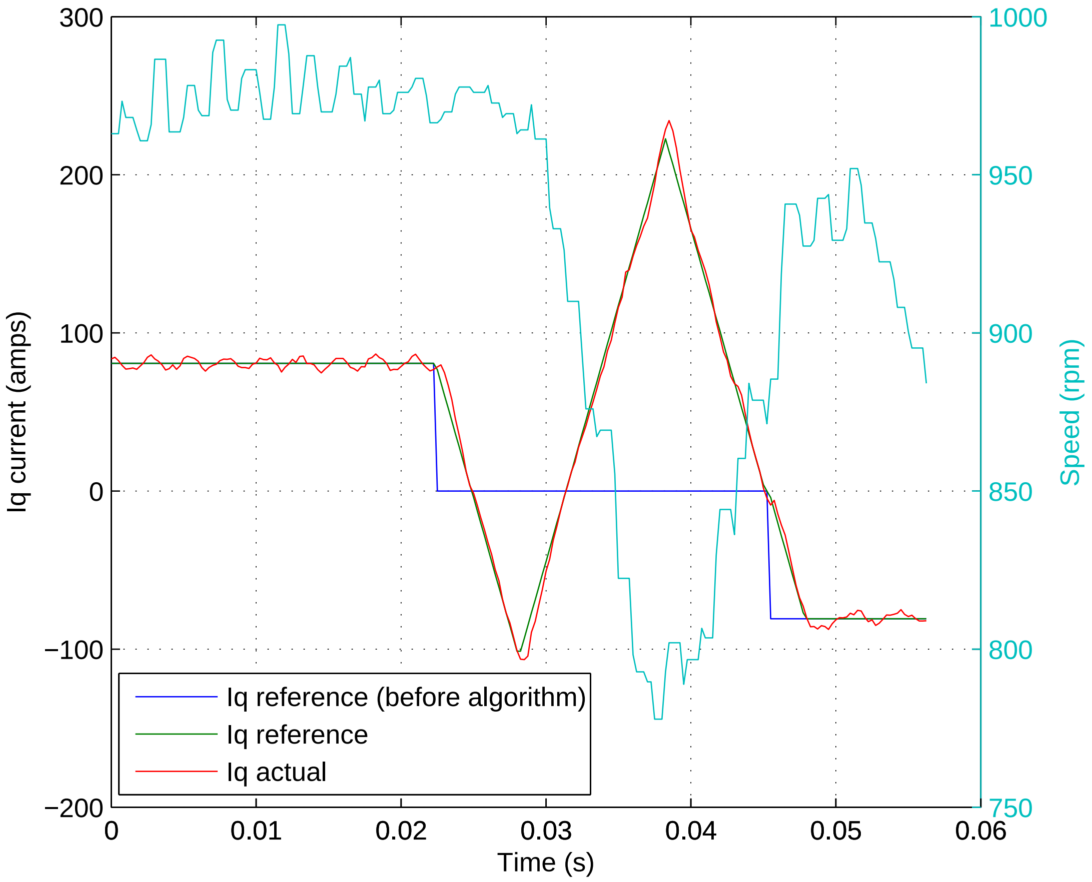

Figure 7.38: Experimental results at 25% torque reversal

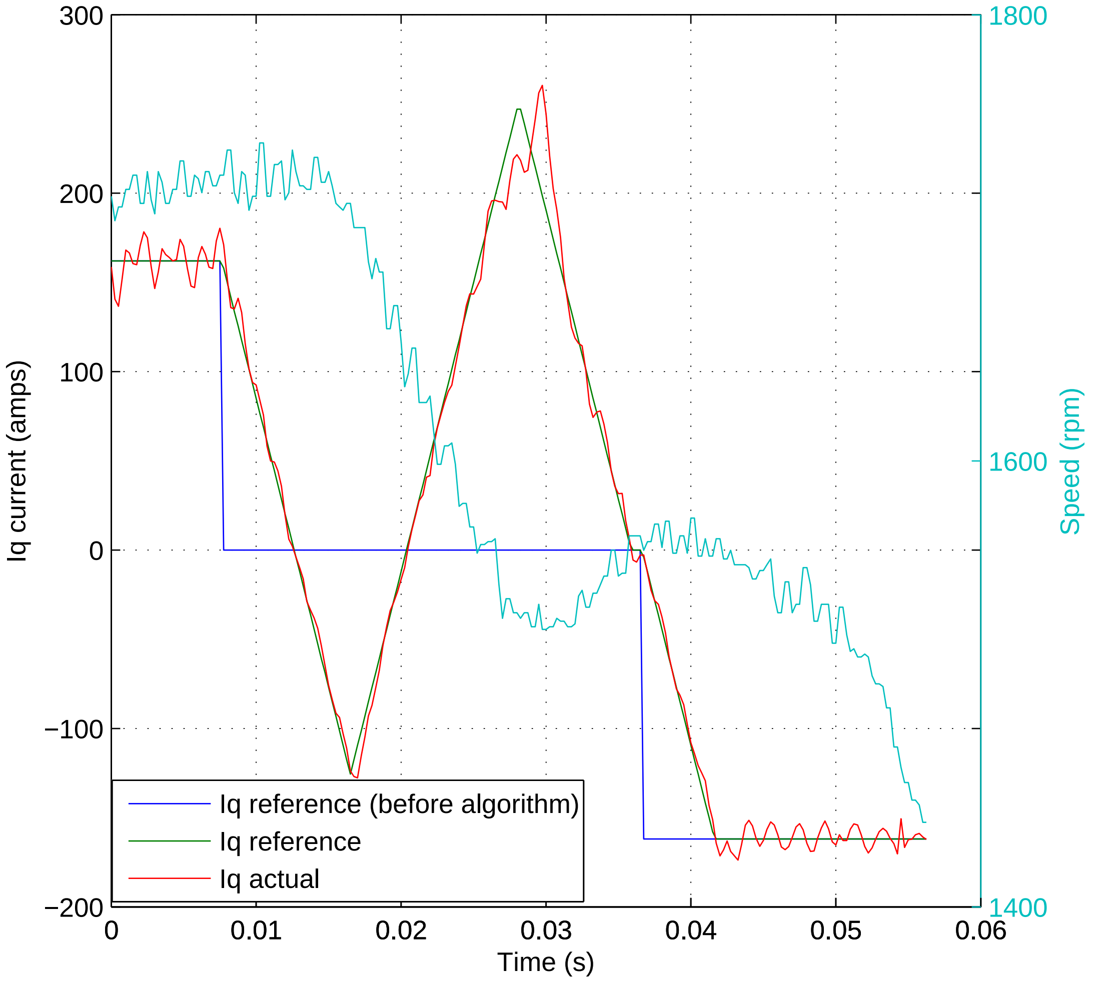

Figure 7.39: Experimental results at 50% torque reversal

### 7.5.5 Discussion

An open loop backlash scheme has been implemented that aims to reduce the impact speed when the gear cogs mesh after traversing the backlash, as high impact speed leads to an oscillation in the drivetrain and a jerk (large change in acceleration) in the vehicles response. Unlike previously published work it utilises and considers the energy stored in the axle flex/twist to accelerate the motor across the backlash (it will also accelerate the vehicle, but due to the inertia differences this will be insignificant), by ramping the torque off faster than the axle ‘unwinds’.

The estimator already included in the software is used to calculate the energy recovery, as this is derived from the physical gearbox backlash model it allows the backlash to be entered before shaft unwinds. It therefore can start applying the acceleration torque pulse before axle has unwound, as normally it is assumed that when the torque crosses zero the backlash will be entered, which is not correct for systems with damping.

It also considers the motor torque (current) ramp rate limitations, as any ramp rate can be fed into the algorithm and the waveform recalculated. This will be required at higher motor speeds, where the smaller voltage headroom (due to larger back EMF) will give less volts to drive the current for the same inductance.

Some assumptions are that the friction losses in the motor are zero and also assuming that the vehicle speed is constant. Also assumes that the axle has stopped oscillating by the time contact (backlash gap closed) is regained, although in the model the axle itself has zero inertia.

## 7.6 Summary

In chapter 2 it was shown that there are a number of undesirable issues with the vehicle drivability. In section 2.3.1 torque reversals were seen to cause significant oscillations in the motor speed that then caused a similar oscillation in the vehicle acceleration, shown in Fig. 2.6 and Fig. 2.7. In section 7.2 a scheme is applied that doesn't require state estimation but improves the smoothness of the acceleration response. When comparing Fig. 7.2 and Fig. 7.3, it can be seen in the second graph that the oscillations that continued almost until the vehicle had stopped have been removed and there is only one small step in speed when entering the backlash.

The speed control response was shown in section 2.3.3 to be difficult to tune, but in section 7.3 above, the estimated vehicle speed is used along with the mass estimate and 'in backlash' indication to improve the responsiveness and smoothness of the vehicle. In Fig. 7.10 the vehicle speed can be seen to track the demand very close during acceleration.

Loss of tyre traction is another issue that was raised in section 2.3.2. Large braking torques, especially when driving forwards, are seen to lock the wheels of the rear wheel driven test vehicle. In section 7.4 above, a scheme is developed to ensure that traction is not lost no matter what level of braking torque is requested on any surface, without fitting any additional sensors. In Fig. 7.16 the algorithm is shown to work when braking whilst driving forwards and also similarly in Fig. 7.19 although braking in reverse. For the reverse driving result (Fig. 7.19) much larger torques are shown to be applied as there is now more traction available. In reverse the driving axle is at the side of the vehicle closest to the direction of travel, the transfer of the vehicle centre of gravity when braking will increase the traction available (when braking in reverse) and not decrease it as was seen when braking in the forward direction.

Finally an algorithm to traverse the gearbox backlash quickly, but also greatly reducing (in most cases) the impact speed and therefore improving the vehicle's response is described in section 7.5. It also improves the vehicle torque response issues shown in section 2.3.1.

# Chapter 8

# Conclusion

## 8.1 Estimation

A solution has been proposed for improving the speed control of an electric vehicle by using a Kalman Filter. It is computationally efficient as the Kalman Gain is fixed and calculated off-line. Even in the case where the Recursive Least Squares mass estimate is used to correct the matrices within the Kalmam Filter, this only has to be done once after each time the vehicle moves away from stationary. This avoids having to use the Extended Kalman Filter with all its extra on-line processing requirement.

It has been shown that it is possible to estimate the vehicle speed correctly when there is a significant change in the vehicle mass. Using a separate Recursive Least Squares mass estimator has been tested and also tuning the process noise matrix to remove the requirement of using the Recursive Least Squares estimator. The importance of accurately tuning the process noise matrix $\mathbf{Q}$ has been shown and has successfully been carried out using a Genetic Algorithm, which also takes into account the noise reduction of the Kalman Filter.

Finding the correct mass is still of importance, as although with the correctly tuned Kalman Gain the estimator appears to be accurate, when it operates in open loop mode when traversing the backlash or when the tyres are slipping, the incorrect mass will lead to poor estimation of backlash timings or acceleration rates.

A model requiring less states, but retaining the same accuracy and also offering improved robustness to mass and gradient changes is proposed. Its performance is compared to the earlier vehicle model with more states, that was used in the rest of this chapter.

Previous work considered in the literature review usually either uses the full Extended Kalman Filter, or ignores the effect of mass changes.

## 8.2 Control

The issue of oscillatory vehicle response and poor speed control is addressed and the estimation plays an important role in this. Previously lack of knowledge of the vehicle drivetrain states and dynamics could lead to poor control. The standard method of using motor speed feedback rather than load has been shown to be incorrect under a number of circumstances.

Tyre slip is a factor that generally limits the amount of regenerative braking allowed on a vehicle, therefore causing more energy to be wasted using the friction brakes. Regenerative braking is also used to relatively high levels for a number of industrial vehicles. A scheme has been presented that reduces the risk of wheel locking and tyre slip and allows braking levels to be set as high as required without the risk of loss of traction. This should work if the vehicle mass has changed significantly, due to the mass estimator feedback, which other work has not considered.

The backlash is the most significant issue in the drivetrain, causing delays before reversal torques can be applied and initiating oscillations. A scheme that focusses on just getting across the backlash is presented without causing impacts in the gearbox, reducing wear and an unpleasant jerk in vehicle acceleration.

## 8.3 Further Work

The full affect of gradients have not been considered and further work needs to be carried out to include the estimation of road incline. There is also the assumption that rolling resistance is proportional to mass changes, but for example, changes in tyre temperature can also change the rolling resistance coefficients. The Genetic Algorithm tuned estimator has been shown to be fairly robust to significant mass changes ($2000kg$), greater than 400% of the unloaded mass, so the effects of these assumptions should be quite small in comparison. The impact of a mass change for a vehicle trailer has not been investigated.

There are a few areas of work that have not been considered, one of the main ones is cornering dynamics and its effect on the estimation and therefore also the tyre slip algorithm. There is the potential for too much braking torque to cause lateral tyre slip as well as longitudinal when cornering.

Although using vehicle speed feedback greatly improves the ease that the speed control can be tuned, a dedicated tuning mechanism has not been considered to fully automate this procedure.

In order to further validate the results on-road tests should be carried out under real driving conditions, and also simulation under more realistic driving conditions. This could then cover: gradients, different acceleration rates, the effect of the mechanical brakes and cornering forces, as the effect of these has not been fully investigated.

Only one vehicle was used experimentally for this work, a comparison with other vehicle types should be done to ensure that the control and estimation algorithm are just as effective for other vehicle types.

# Appendix A

# Appendix A

## A.1 Estimation

Discrete matrices used throughout this thesis after rescaling so that all speeds are in $rpm$ referenced to the motor and torques are 100 times larger than motor torque in $Nm$.

### A.1.1 Vehicle Model matrices - 5 state

$$\mathbf{A_{d-co}} = \begin{pmatrix} 0.9959 & 0.0009864 & 0.004067 & 1.315e-005 & 0.004067 \\ -8.094 & 0.9717 & 8.094 & 0.02754 & 8.094 \\ 0.0001841 & 5.951e-007 & 0.9998 & 0.0009994 & -0.0001841 \\ 0.3663 & 0.001246 & -0.3663 & 0.9987 & -0.3663 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.1}$$

$$\mathbf{A_{d-bl}} = \begin{pmatrix} 1 & 0.0009996 & 0 & 0 & 0 \\ 0 & 0.9992 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.001 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0.2917 & 0.0009996 & -0.2917 & -0.001 & 0.7083 \end{pmatrix} \tag{A.2}$$

$$\mathbf{B_d} = \begin{pmatrix} 6.761e-006 & -2.643e-009 \\ 0.01346 & -8.119e-006 \\ 2.643e-009 & -3.085e-007 \\ 8.119e-006 & -0.0006168 \\ 0 & 0 \end{pmatrix} \tag{A.3}$$

$$\mathbf{C_{d-co}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} \tag{A.4}$$

## A.1.2 Noise matrices and Kalman gain - 5 state

### A.1.2.1 Trial and Error

Trial and error:

$$\mathbf{Q_{TE}} = \begin{pmatrix} 0.00533 & 0 & 0 & 0 & 0 \\ 0 & 0.00533 & 0 & 0 & 0 \\ 0 & 0 & 3.118 & 0 & 0 \\ 0 & 0 & 0 & 3.118 & 0 \\ 0 & 0 & 0 & 0 & 1.375e-006 \end{pmatrix} \tag{A.5}$$

$$\mathbf{R_{TE}} = \begin{pmatrix} 0.5 & 0 \\ 0 & 5 \end{pmatrix} \tag{A.6}$$

$$\mathbf{K_{d-co-TE}} = \begin{pmatrix} 0.0330 & 0.000406 \\ 0.00406 & 0.885 \\ 0.0312 & 0.0865 \\ 0.0169 & 0.0437 \\ 5.882e-009 & 1.231e-009 \end{pmatrix} \tag{A.7}$$

### A.1.2.2 Genetic Algorithm

Genetic algorithm:

$$\mathbf{Q_{GA}} = \begin{pmatrix} 8.698e-005 & 0 & 0 & 0 & 0 \\ 0 & 0.0881 & 0 & 0 & 0 \\ 0 & 0 & 2.758e-004 & 0 & 0 \\ 0 & 0 & 0 & 0.177 & 0 \\ 0 & 0 & 0 & 0 & 8.995e-006 \end{pmatrix} \tag{A.8}$$

$$\mathbf{R_{GA}} = \begin{pmatrix} 0.5 & 0 \\ 0 & 5 \end{pmatrix} \tag{A.9}$$

$$\mathbf{K_{d-co-GA}} = \begin{pmatrix} 0.0133 & 2.078e-004 \\ 0.00208 & 0.313 \\ 0.0125 & 0.00691 \\ 0.104 & 0.139 \\ 1.0633e-006 & 2.137e-007 \end{pmatrix} \tag{A.10}$$

### A.1.2.3 Least Squares and Minimisation Search

Non-linear least squares:

$$\mathbf{Q_{LS}} = \begin{pmatrix} 7.151e-007 & 0 & 0 & 0 & 0 \\ 0 & 9.172e-005 & 0 & 0 & 0 \\ 0 & 0 & 0.00496 & 0 & 0 \\ 0 & 0 & 0 & 0.311 & 0 \\ 0 & 0 & 0 & 0 & 1.165e-004 \end{pmatrix} \tag{A.11}$$

$$\mathbf{R_{LS}} = \begin{pmatrix} 0.5 & 0 \\ 0 & 5 \end{pmatrix} \tag{A.12}$$

$$\mathbf{K_{d-co-LS}} = \begin{pmatrix} 0.00337 & 0.000725 \\ 0.00725 & 0.512 \\ 0.00321 & 0.0225 \\ 0.00213 & 0.150 \\ 4.709e-005 & 2.075e-0060 \end{pmatrix} \tag{A.13}$$

Minimisation search:

$$\mathbf{Q_{MS}} = \begin{pmatrix} 0.186 & 0 & 0 & 0 & 0 \\ 0 & 1e-015 & 0 & 0 & 0 \\ 0 & 0 & 1e-0150 & 0 & \\ 0 & 0 & 0 & 0.0563 & 0 \\ 0 & 0 & 0 & 0 & 1e-015 \end{pmatrix} \tag{A.14}$$

$$\mathbf{R_{MS}} = \begin{pmatrix} 0.5 & 0 \\ 0 & 5 \end{pmatrix} \tag{A.15}$$

$$\mathbf{K_{d-co-MS}} = \begin{pmatrix} 0.338 & -0.0427 \\ -0.427 & 0.759 \\ 0.0154 & 0.00347 \\ 0.218 & 0.0118 \\ -2.773e-014 & -5.273e-015 \end{pmatrix} \tag{A.16}$$

### A.1.3 Jacobian matrices - 5 state

$$\mathbf{J_{A_{d-co}}} = \begin{pmatrix} 3.115e-004 & 4.205e-007 & -3.115e-004 & -4.352e-007 & -3.115E-004 \\ 0.990 & 0.00141 & -0.990 & -0.00146 & -0.990 \\ 0.0839 & 1.210e-004 & -0.0839 & -1.250e-004 & -0.0839 \\ 167.324 & 0.269 & -167.324 & -0.277 & -167.324 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{A.17}$$

$$\mathbf{J_{A_{d-bl}}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -3.945e-006 & 0 \\ 0 & 0 & 0 & -0.00789 & 0 \\ 0 & 0 & 0 & 3.945e-006 & 0 \end{pmatrix} \tag{A.18}$$

$$\mathbf{J_{B_d}} = \begin{pmatrix} 1.336e-009 & -5.189e-007 \\ 5.737e-006 & -0.00165 \\ 5.189e-007 & -1.399e-004 \\ 0.00165 & -0.280 \\ 0 & 0 \end{pmatrix} \tag{A.19}$$

$$\mathbf{J_{K_{d-co-TE}}} = \begin{pmatrix} -9.063e-006 & -2.657e-005 \\ -2.657e-004 & -0.0325 \\ 1.338e-004 & 6.612e-004 \\ -0.217 & -18.814 \\ -1.523e-007 & 2.957e-007 \end{pmatrix} \tag{A.20}$$

### A.1.4 Vehicle Model matrices - 4 state

$$\mathbf{A_{d-co}} = \begin{pmatrix} 0.9723 & 0.02755 & -8.097 & -8.12e-006 \\ 0.001246 & 0.9987 & 0.3663 & -0.0006168 \\ 0.0009862 & -0.0009862 & 0.9957 & 3.058e-007 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.21}$$

$$\mathbf{A_{d-bl}} = \begin{pmatrix} 0.9999 & 0 & 0 & 0 \\ 0 & 1 & 0 & -0.0006172 \\ 0 & 0 & 0.7083 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.22}$$

$$\mathbf{B_d} = \begin{pmatrix} 0.01346 & -8.12e-006 \\ 8.12e-006 & -0.0006168 \\ 6.76e-006 & 3.058e-007 \\ 0 & 0 \end{pmatrix} \tag{A.23}$$

$$\mathbf{C_{d-co}} = \begin{pmatrix} 1 & 0 & 0 & 0 \end{pmatrix} \tag{A.24}$$

### A.1.5 Noise matrices and Kalman gain - 4 state

Trial and error:

$$\mathbf{Q} = \begin{pmatrix} 0.456 & 0 & 0 & 0 \\ 0 & 0.0456 & 0 & 0 \\ 0 & 0 & 0.000137 & 0 \\ 0 & 0 & 0 & 500 \end{pmatrix} \tag{A.25}$$

$$\mathbf{R} = \begin{pmatrix} 5 \end{pmatrix} \tag{A.26}$$

$$\mathbf{K_{d-co}} = \begin{pmatrix} 0.357 \\ 0.240 \\ -0.00518 \\ -8.021 \end{pmatrix} \tag{A.27}$$

# References

[1] "Collins dictonary." [Online]. Available: http://www.collinsdictionary.com/dictionary/english/driveability 2

[2] P. Templin, "Simultaneous estimation of driveline dynamics and backlash size for control design," in *Control Applications, 2008. CCA 2008. IEEE International Conference on*, 2008, pp. 13–18. 26, 27

[3] M. Berriri, P. Chevrel, and D. Lefebvre, "Active damping of automotive powertrain oscillations by a partial torque compensator," *Control Engineering Practice*, vol. 16, no. 7, pp. 874 – 883, 2008. 26, 27, 33, 34

[4] H. E. M. Menday, M. T.; Rahnejat, "Clonk: an onomatopoeic response in torsional impact of automotive drivelines," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 213, no. 4, pp. 349–358, 1999. 27

[5] A. Lagerberg, "Control and estimation of automotive powertrains with backlash," Ph.D. dissertation, Department of Signals and Systems, Chalmers University of Technology, 2004. 27, 28

[6] M. Valenzuela, J. Bentley, and R. Lorenz, "Evaluation of torsional oscillations in paper machine sections," *Industry Applications, IEEE Transactions on*, vol. 41, no. 2, pp. 493–501, 2005. 27, 33

[7] R. Morselli, R. Zanasi, and A. Visconti, "Generation of acceleration profiles for smooth gear shift operations," in *IECON 02 [Industrial Electronics Society, IEEE 2002 28th Annual Conference of the]*, vol. 2, 2002, pp. 1681–1686 vol.2. 27

[8] M. Iwasaki, M. Miwa, and N. Matsui, "Ga-based evolutionary identification algorithm for unknown structured mechatronic systems," *Industrial Electronics, IEEE Transactions on*, vol. 52, no. 1, pp. 300–305, 2005. 27

[9] A. Lagerberg and B. Egardt, "Backlash estimation with application to automotive powertrains," *Control Systems Technology, IEEE Transactions on*, vol. 15, no. 3, pp. 483 –493, may 2007. 27, 28, 56, 57, 122, 123, 126

[10] R. Zanasi, A. Visconti, G. Sandoni, and R. Morselli, “Dynamic modeling and control of a car transmission system,” in *Advanced Intelligent Mechatronics, 2001. Proceedings. 2001 IEEE/ASME International Conference on*, vol. 1, 2001, pp. 416 –421 vol.1. 28, 40

[11] L. Imsland, T. A. Johansen, T. I. Fossen, H. F. Grip, J. C. Kalkkuhl, and A. Suissa, “Vehicle velocity estimation using nonlinear observers,” *Automatica*, vol. 42, no. 12, pp. 2091 – 2103, 2006. 28

[12] M. Wada, K. S. Yoon, and H. Hashimoto, “High accuracy road vehicle state estimation using extended kalman filter,” in *Intelligent Transportation Systems, 2000. Proceedings. 2000 IEEE*, 2000, pp. 282–287. 28

[13] M. Tanelli, S. Savaresi, and C. Cantoni, “Longitudinal vehicle speed estimation for traction and braking control systems,” in *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*, 2006, pp. 2790–2795. 28

[14] K. Kobayashi, K. C.Cheok, and K. Watanabe, “Estimation of absolute vehicle speed using fuzzy logic rule-based kalman filter,” in *American Control Conference, Proceedings of the 1995*, vol. 5, 1995, pp. 3086–3090 vol.5. 28, 37

[15] D. Kun, L. Kaijun, and X. Qunsheng, “Application of unscented kalman filter for the state estimation of anti-lock braking system,” in *Vehicular Electronics and Safety, 2006. ICVES 2006. IEEE International Conference on*, 2006, pp. 130–133. 29

[16] J. Mora, A. Torralba, and L. Franquelo, “An adaptive speed estimator for induction motors based on a kalman filter with low sample time,” in *Power Electronics Specialists Conference, 2001. PESC. 2001 IEEE 32nd Annual*, vol. 2, 2001, pp. 794–798 vol.2. 29

[17] R. Dhaouadi, N. Mohan, and L. Norum, “Design and implementation of an extended kalman filter for the state estimation of a permanent magnet synchronous motor,” *Power Electronics, IEEE Transactions on*, vol. 6, no. 3, pp. 491–497, 1991. 29, 175

[18] Z. Peroutka, “Development of sensorless pmsm drives: Application of extended kalman filter,” in *Industrial Electronics, 2005. ISIE 2005. Proceedings of the IEEE International Symposium on*, vol. 4, 2005, pp. 1647–1652. 29

[19] B. S. P. Lingman, “Road slope and vehicle mass estimation using kalman filtering,” *Vehicle System Dynamics*, vol. 37, pp. 12–23, 2002. 29, 126

[20] H. Ohnishi, J. Ishii, M. Kayano, and H. Katayama, "A study on road slope estimation for automatic transmission control," {*JSAE*} *Review*, vol. 21, no. 2, pp. 235 – 240, 2000. 30, 145

[21] J. C. Bae, Hong S.; & Gerdes, "Parameter estimation and command modification for longitudinal control of heavy vehicle," UC Berkeley: California Partners for Advanced Transit and Highways (PATH), Tech. Rep., 2003. 30

[22] H. P. A. Vahidi, A. Stefanopoulou, "Recursive least squares with forgetting for online estimation of vehicle mass and road grade: theory and experiments," *Vehicle System Dynamics*, vol. 43, no. 1, pp. 31–35, jan. 2005. 30

[23] T.-J. Kweon and D.-S. Hyun, "High-performance speed control of electric machine using low-precision shaft encoder," *Power Electronics, IEEE Transactions on*, vol. 14, no. 5, pp. 838 –849, sep 1999. 31, 174

[24] S. Yan, D. Xu, G. Wang, M. Yang, Y. Yu, and X. Gui, "Low speed control of pmac servo system based on reduced-order observer," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, oct. 2006, pp. 4886 –4889. 31, 174

[25] K. Szabat and T. Orlowska-Kowalska, "Performance improvement of industrial drives with mechanical elasticity using nonlinear adaptive kalman filter," *Industrial Electronics, IEEE Transactions on*, vol. 55, no. 3, pp. 1075 –1084, march 2008. 31, 32, 34, 175

[26] G. Koch, T. Kloiber, E. Pellegrini, and B. Lohmann, "A nonlinear estimator concept for active vehicle suspension control," in *American Control Conference (ACC), 2010*, 2010, pp. 4576–4581. 31

[27] T. A. Wenzel, K. J. Burnham, M. V. Blundell, and R. A. Williams, "Dual extended kalman filter for vehicle state and parameter estimation," *Vehicle System Dynamics*, vol. 44, no. 2, pp. 153–171, 2006. 31

[28] V. Winstead and I. Kolmanovsky, "Estimation of road grade and vehicle mass via model predictive control," in *Control Applications, 2005. CCA 2005. Proceedings of 2005 IEEE Conference on*, 2005, pp. 1588–1593. 31

[29] M. Barut, S. Bogosyan, and M. Gokasan, "Speed sensorless direct torque control of {IMs} with rotor resistance estimation," *Energy Conversion and Management*, vol. 46, no. 3, pp. 335 – 349, 2005. 31

[30] M. Best, T. Gordon, and P. Dixon, "An extended adaptive kalman filter for real-time state estimation of vehicle handling dynamics," *Vehicle System Dynamics*, vol. 34, no. 1, pp. 57–75, 2000. 31

[31] S. Bolognani, L. Tubiana, and M. Zigliotto, "Extended kalman filter tuning in sensorless pmsm drives," *Industry Applications, IEEE Transactions on*, vol. 39, no. 6, pp. 1741–1747, 2003. 32

[32] R. Mehra, "On the identification of variances and adaptive kalman filtering," *Automatic Control, IEEE Transactions on*, vol. 15, no. 2, pp. 175–184, 1970. 32

[33] X.-R. Li and Y. Bar-Shalom, "Multiple-model estimation with variable structure," *Automatic Control, IEEE Transactions on*, vol. 41, no. 4, pp. 478–493, 1996. 32

[34] M. Karasalo and X. Hu, "An optimization approach to adaptive kalman filtering," *Automatica*, vol. 47, no. 8, pp. 1785 – 1793, 2011. 32

[35] K. Szabat and T. Orlowska-Kowalska, "Application of the extended kalman filter in advanced control structure of a drive system with elastic joint," in *Industrial Technology, 2008. ICIT 2008. IEEE International Conference on*, 2008, pp. 1–6. 32

[36] S. Gadoue, D. Giaouris, and J. Finch, "Artificial intelligence-based speed control of dtc induction motor drivesa comparative study," *Electric Power Systems Research*, vol. 79, no. 1, pp. 210 – 219, 2009. 32, 100, 176

[37] T. Ming, G. Lin, and L. Deliang, "Sensorless permanent magnet synchronous motor drive using an optimized and normalized extended kalman filter," in *Electrical Machines and Systems (ICEMS), 2011 International Conference on*, aug. 2011, pp. 1 –4. 32, 175

[38] K. L. Shi, T. Chan, Y. Wong, and S. Ho, "Speed estimation of an induction motor drive using an optimized extended kalman filter," *Industrial Electronics, IEEE Transactions on*, vol. 49, no. 1, pp. 124–133, 2002. 32

[39] Z. Pei, Q. Tong, L. Wang, and J. Zhang, "A median filter method for image noise variance estimation," in *Information Technology and Computer Science (ITCS), 2010 Second International Conference on*, 2010, pp. 13–16. 32

[40] K. Wooil, "Noise variance estimation for kalman filtering of noisy speech," *IEICE TRANSACTIONS on Information and Systems*, vol. 84, no. 1, pp. 155–160, 2001. 32

[41] T. Y. Um, J. G. Lee, S.-T. Park, and C. G. Park, "Noise covariances estimation for systems with bias states," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 36, no. 1, pp. 226 –233, jan 2000. 32

[42] B. M. kesson, J. B. Jrgensen, N. K. Poulsen, and S. B. Jrgensen, "A generalized autocovariance least-squares method for kalman filter tuning," *Journal of Process Control*, vol. 18, no. 78, pp. 769 – 779, 2008. 32

[43] E. Mohammadiasl, "Vibration detection and backlash suppression in machine tools," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, 2009, pp. 972–977. 33

[44] J. Pacas, A. John, and T. Eutebach, "Automatic identification and damping of torsional vibrations in high-dynamic-drives," in *Industrial Electronics, 2000. ISIE 2000. Proceedings of the 2000 IEEE International Symposium on*, vol. 1, 2000, pp. 201–206 vol.1. 33

[45] M. A. T. F. De Sousa, S. Caux, and M. Fadel, "Design of robust controllers for pmsm drive fed with pwm inverter with inertia load variation," in *Industrial Electronics, 2006 IEEE International Symposium on*, vol. 1, 2006, pp. 217 – 222. 33

[46] N. Amann, J. Bocker, and F. Prenner, "Active damping of drive train oscillations for an electrically driven vehicle," *Mechatronics, IEEE/ASME Transactions on*, vol. 9, no. 4, pp. 697 –700, dec. 2004. 33

[47] S. Thomsen and F. Fuchs, "Speed control of torsional drive systems with backlash," in *Power Electronics and Applications, 2009. EPE '09. 13th European Conference on*, sept. 2009, pp. 1 –10. 33

[48] S. Thomsen, N. Hoffmann, and F. W. Fuchs, "Pi control, pi-based state space control, and model-based predictive control for drive systems with elastically coupled loads-a comparative study," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 8, pp. 3647 – 3657, 2011. 33

[49] K. Szabat and T. Orlowska-Kowalska, "Vibration suppression in a two-mass drive system using pi speed controller and additional feedbacks - comparative study," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 2, pp. 1193 – 1206, 2007. 33

[50] R. Dhaouadi, K. Kubo, and M. Tobise, "Analysis and compensation of speed drive systems with torsional loads," in *Power Conversion Conference, 1993. Yokohama 1993., Conference Record of the*, 1993, pp. 271–277. 33

[51] R. Zhang, Y. Yang, Z. Chen, and C. Tong, "Torsional vibration suppression control in the main drive system of rolling mill by state feedback speed controller based on extended state observer," in *Control and Automation, 2007. ICCA 2007. IEEE International Conference on*, 2007, pp. 2172–2177. 33

[52] J.-K. Ji and S.-K. Sul, “Kalman filter and lq based speed controller for torsional vibration suppression in a 2-mass motor drive system,” *Industrial Electronics, IEEE Transactions on*, vol. 42, no. 6, pp. 564–571, 1995. 33

[53] P. Templin and B. Egardt, “Experimental results for a powertrain lqr-torque compensator with backlash handling,” in *2009 IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling*, 2009. 33

[54] P. Templin and B. Egardt, “A powertrain lqr-torque compensator with backlash handling,” in *Oil & gas science and technology*, vol. 66, no. 4, 2011, pp. 645–654. 33

[55] M. Cychowski, K. Szabat, and T. Orlowska-Kowalska, “Constrained model predictive control of the drive system with mechanical elasticity,” *IEEE Transactions on Industrial Electronics*, vol. 56, no. 6, pp. 1963 – 1973, 2009. 34

[56] M. Odai and Y. Hori, “Speed control of 2-inertia system with gear backlash using gear torque compensator,” in *Advanced Motion Control, 1998. AMC ’98-Coimbra., 1998 5th International Workshop on*, 1998, pp. 234–239. 34

[57] Y. Wu, K. Fujikawa, and H. Kobayashi, “A control method of speed control drive system with backlash,” in *Advanced Motion Control, 1996. AMC ’96-MIE. Proceedings., 1996 4th International Workshop on*, vol. 2, 1996, pp. 631–636 vol.2. 34

[58] Y. Nakayama, K. Fujikawa, and H. Kobayashi, “A torque control method of three-inertia torsional system with backlash,” in *Advanced Motion Control, 2000. Proceedings. 6th International Workshop on*, 2000, pp. 193–198. 34

[59] J. Baumann, D. D. Torkzadeh, A. Ramstein, U. Kiencke, and T. Schlegl, “Model-based predictive anti-jerk control,” *Control Engineering Practice*, vol. 14, no. 3, pp. 259 – 266, 2006. 34

[60] P. Stewart, J. C. Zavala, and P. J. Fleming, “Automotive drive by wire controller design by multi-objective techniques,” *Control Engineering Practice*, vol. 13, no. 2, pp. 257 – 264, 2005. 34

[61] M. Nordin and P.-O. Gutman, “Nonlinear speed control of elastic systems with backlash,” in *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, vol. 4, 2000, pp. 4060–4065 vol.4. 34

[62] M. Nordin and P.-O. Gutman, “Controlling mechanical systems with backlash - a survey,” *Automatica*, vol. 38, no. 10, pp. 1633 – 1649, 2002. 34

[63] G. Tao, “Adaptive control of systems with nonsmooth input and output nonlinearities,” *Automatic Control, IEEE Transactions on*, vol. 41, no. 9, pp. 1348–1352, 1996. 35

[64] A. Lagerberg and B. Egardt, “Evaluation of control strategies for automotive powertrains with backlash,” in *6th International Symposium on Advanced Vehicle Control : Hiroshima, Japan, 2002*, 2002, pp. 517–522. 35

[65] H. Mokhtari and F. Barati, “A new scheme for a mechanical load position control driven by a permanent magnet dc motor and a nonzero backlash gearbox,” in *Industrial Electronics, 2006 IEEE International Symposium on*, vol. 3, july 2006, pp. 2052 –2057. 35

[66] M. Warnecke and M. Jouaneh, “Backlash compensation in gear trains by means of open-loop modification of the input trajectory,” *Journal of Mechanical Design*, vol. 125, no. 3, pp. 620–624, 2003. 35

[67] K. Ezal, P. Kokotovic, and G. Tao, “Optimal control of tracking systems with backlash and flexibility,” in *Decision and Control, 1997., Proceedings of the 36th IEEE Conference on*, vol. 2, dec 1997, pp. 1749 –1754 vol.2. 35

[68] A. Lagerberg and B. Egardt, “Model predictive control of automotive powertrains with backlash,” in *16th IFAC world congress. Prague, Czech Republic*, 2005. 35

[69] H.-S. Tan and M. Tomizuka, “Discrete-time controller design for robust vehicle traction,” *Control Systems Magazine, IEEE*, vol. 10, no. 3, pp. 107–113, 1990. 36

[70] T. Johansen, I. Petersen, J. Kalkkuhl, and J. Ludemann, “Gain-scheduled wheel slip control in automotive brake systems,” *Control Systems Technology, IEEE Transactions on*, vol. 11, no. 6, pp. 799–811, 2003. 36

[71] M. Hancock, “Impact of regenerative braking on vehicle stability,” in *Hybrid Vehicle Conference, IET The Institution of Engineering and Technology, 2006*, 2006, pp. 173–184. 36, 37

[72] Z. Lei, L. Yugong, Y. Diange, L. Keqiang, and L. Xiaomin, “A novel brake control strategy for electric vehicles based on slip trial method,” in *Vehicular Electronics and Safety, 2007. ICVES. IEEE International Conference on*, 2007, pp. 1–6. 36

[73] C. Mi, H. Lin, and Y. Zhang, “Iterative learning control of antilock braking of electric and hybrid vehicles,” *Vehicular Technology, IEEE Transactions on*, vol. 54, no. 2, pp. 486–494, 2005. 36

[74] B. E. Stefan Solyom, “A method and device for regenerative braking in a vehicle,” European Patent EP 2 172 378 A1, 2010. 36

[75] C. Carlson and J. Gerdes, “Consistent nonlinear estimation of longitudinal tire stiffness and effective radius,” *Control Systems Technology, IEEE Transactions on*, vol. 13, no. 6, pp. 1010–1020, 2005. 36

[76] K. Fujii and H. Fujimoto, “Traction control based on slip ratio estimation without detecting vehicle speed for electric vehicle,” in *Power Conversion Conference-Nagoya, 2007. PCC’07.* IEEE, 2007, pp. 688–693. 36

[77] P. Ratiroch-Anant, H. Hirata, M. Anabuki, and S. Ouchi, “Adaptive controller design for anti-slip system of ev,” in *Robotics, Automation and Mechatronics, 2006 IEEE Conference on*, june 2006, pp. 1 –6. 37

[78] P. G. M. Nordin, J. Galic, “New models for backlash and gear play,” *International Journal of Adaptive Control and Signal Processing*, vol. 11, pp. 49–63, 2006. 56, 66, 122

[79] A. Haddoun, M. E. H. Benbouzid, D. Diallo, R. Abdessemed, J. Ghouili, and K. Srairi, “A loss-minimization dtc scheme for ev induction motors,” *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 1, pp. 81 –88, jan. 2007. 70, 73

[80] I. Husain and M. Islam, “Design, modeling and simulation of an electric vehicle system,” *SAE transactions*, vol. 108, no. 6; PART 2, pp. 2168–2176, 2000. 73

[81] A. Boyali, M. Demirci, T. Acarman, L. Guvenc, O. Tur, H. Ucarol, B. Kiray, and E. Ozatay, “Modeling and control of a four wheel drive parallel hybrid electric vehicle,” in *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*, 2006, pp. 155–162. 73

[82] J. Shao, L. Zheng, Y. Li, J. Wei, and M. Luo, “The integrated control of anti-lock braking system and active suspension in vehicle,” in *Fuzzy Systems and Knowledge Discovery, 2007. FSKD 2007. Fourth International Conference on*, vol. 4, aug. 2007, pp. 519 –523. 76, 77, 81, 82

[83] A. B. WILL and S. H. ZAK, “Modelling and control of an automated vehicle,” *Vehicle System Dynamics*, vol. 27, no. 3, pp. 131–155, 1997. 76

[84] H. B. Pacejka, *Tyre and Vehicle Dynamics.* Elsevier Butterworth-Heinemann, 2002. 81

[85] C. Canudas de Wit and P. Tsiotras, “Dynamic tire friction models for vehicle traction control,” in *Decision and Control, 1999. Proceedings of the 38th IEEE Conference on*, vol. 4, 1999, pp. 3746–3751 vol.4. 81

[86] S. T. K. Dutton and B. Barraclough, *The Art of Control Engineering.* Pearson - Prentice Hall, 1997. 117

[87] D. Giaouris, “Eee8013 - design of linear systems,” Lecture Notes, 2008. 117

[88] M. Armstrong, “Eee8007 - advanced control systems,” Lecture Notes, 2008. 118

[89] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Transactions of the ASME Journal of Basic Engineering*, vol. 82 (Series D), pp. 35–45, 1960. 119

[90] M. Grewal and A. Andrews, “Applications of kalman filtering in aerospace 1960 to the present [historical perspectives],” *Control Systems, IEEE*, vol. 30, no. 3, pp. 69–78, 2010. 119

[91] D. Simon, *Optimal State Estimation.* Wiley-Interscience, 2006. 120, 121, 127, 145