LUND UNIVERSITY

Model-Based Vehicle Dynamics Control for Active Safety

Schofield, Brad

2008

Document Version:
Publisher's PDF, also known as Version of record

Link to publication

Citation for published version (APA):
Schofield, B. (2008). *Model-Based Vehicle Dynamics Control for Active Safety*. [Doctoral Thesis (monograph), Department of Automatic Control]. Department of Automatic Control, Lund Institute of Technology, Lund University.

Total number of authors:
1

General rights
Unless other specific re-use rights are stated the following general rights apply:
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: https://creativecommons.org/licenses/

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

Model-Based Vehicle Dynamics Control for Active Safety

Model-Based Vehicle Dynamics Control for Active Safety

Brad Schofield

Department of Automatic Control
Lund University
Lund, September 2008

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

ISSN 0280–5316
ISRN LUTFD2/TFRT--1083--SE

© 2008 by Brad Schofield. All rights reserved.
Printed in Sweden by Media-Tryck.
Lund 2008

To the memory of my mother, Linda

Abstract

The functionality of modern automotive vehicles is becoming increasingly dependent on control systems. Active safety is an area in which control systems play a pivotal role. Currently, rule-based control algorithms are widespread throughout the automotive industry. In order to improve performance and reduce development time, model-based methods may be employed.

The primary contribution of this thesis is the development of a vehicle dynamics controller for rollover mitigation. A central part of this work has been the investigation of control allocation methods, which are used to transform high-level controller commands to actuator inputs in the presence of numerous constraints. Quadratic programming is used to solve a static optimization problem in each sample. An investigation of the numerical methods used to solve such problems was carried out, leading to the development of a modified active set algorithm.

Vehicle dynamics control systems typically require input from a number of supporting systems, including observers and estimation algorithms. A key parameter for virtually all VDC systems is the friction coefficient. Model-based friction estimation based on the physically-derived brush model is investigated.

Acknowledgments

I would like to begin by thanking my supervisor Tore Hägglund for his unwavering support, help and encouragement during my time at the department. The freedom to choose my own directions, in combination with his support, have made my research studies extremely enjoyable.

I would also like to thank everyone involved in the CEmACS project, in particular Dr. Jens Kalkkuhl at Daimler. I thoroughly enjoyed working on this project, and have learned a great deal. Thanks also go to everyone involved in the RFE II project, in particular to Jacob Svendenius, Stefan Solyom and Fredrik Bruzelius, for very rewarding cooperation. Many thanks to everyone at Modelon, in particular Magnus Gäfvert, for assistance with the Vehicle Dynamics library.

The Department of Automatic Control is an excellent workplace, which I have had the privilege of sharing for the past five years. I have made too many friends to list here, but would nevertheless like to thank certain people in particular. I enjoyed sharing a room with Ather Gattami during the first three years of our studies. In the beginning this often involved helping each other with teaching preparations, typically outside of normal working hours. Thanks go to Stéphane Velut, for introducing me to climbing, and to both him and Staffan Haugwitz, for memorable winter hikes featuring unreasonably large camp fires. Thanks also go to Martin Kjær, whose knowledge of beer never ceases to impress. Last but not least, thanks go to Oskar Nilsson, the only other person at the department with a sensible taste in music.

Thanks also go to those who keep the department running with such efficiency. These include Eva Schildt, Britt-Marie Mårtensson and Agneta Tuszynski, as well as Leif Andersson and Anders Blomdell. I would also like to thank Rolf Braun, for letting me borrow tools from his workshop. These have been used at least as much for the construction and repair of mountain bikes as for research purposes.

I would like to thank everyone with whom I have participated in a

host of activities in the past five years. There are far too many people and experiences to list, but the memories include sunbathing on the top of Glittertind, skiing in Jämtland, Vätternrundan, ice climbing in Norway, and the Arête des Cosmiques, to name a few.

This thesis would not have come into being were it not for the support of my family. My mother's endless support and encouragement have always been important for me, and will be sorely missed now that she is gone. My father's work ethic and drive have been an example and an inspiration. Finally, thanks to my brother, for showing what can be done, when one wants to.

Contents

1. Introduction 11
1.1 Background 11
1.2 Motivation 13
1.3 Scope 14
1.4 Contributions 16
1.5 Outline 16
1.6 Publications 16

2. Tire Modeling 18
2.1 Slip 18
2.2 Empirical Models 21
2.3 The Friction Ellipse 22
2.4 The Brush Model 23
2.5 Summary 29

3. Vehicle Modeling 31
3.1 Coordinate Systems 31
3.2 Linear Single-Track Model 33
3.3 Two-Track Model 36

4. Rollover Analysis and Detection 51
4.1 Static Rollover Analysis 51
4.2 Load Transfer 54
4.3 Methods for Rollover Detection 58

5. Control Design for Rollover Mitigation 64
5.1 Controller Architecture 65
5.2 Control Objectives 67
5.3 Roll Control 68
5.4 Yaw Control 70
5.5 Summary 75

6. Control Allocation 77

6.1 Problem Formulation 78
6.2 Convex Optimization 78
6.3 Solving Convex Optimization Problems 80
6.4 Control Allocator Design 82
6.5 Summary 94

7. Active Set Methods for Control Allocation 95
7.1 Problem Formulation 96
7.2 Active Set Algorithms 97
7.3 Modifications 99
7.4 Conclusions 107

8. Simulation Results For Rollover Mitigation 109
8.1 Test Maneuvers 109
8.2 Simulation Results 112
8.3 Real Time Performance 131
8.4 Discussion 131

9. Sensor Fusion for Yaw Rate Measurements 134
9.1 Introduction 134
9.2 Model 136
9.3 Estimation Methods for Sensor Fusion 137
9.4 Linear Parameter Varying Observer 142
9.5 Conclusions 144

10. Model-Based Friction Estimation 148
10.1 Friction Estimation Using The Brush Model 150
10.2 Change Detection 159
10.3 Friction Estimation for Heavy Vehicles 167
10.4 Conclusions 168

11. Conclusions 171
11.1 Summary of Contributions 171
11.2 Conclusions and Future Work 172

12. Bibliography 175

A. Nomenclature 183

1

Introduction

1.1 Background

Modern automotive vehicles are increasingly reliant on control systems for their operation. Internal combustion engines require advanced engine management systems in order to meet stringent emissions regulations. Vehicles with hybrid drivetrains require advanced algorithms to optimize the interaction between energy sources in order to maximize efficiency. Control of vehicle dynamics has also become extremely important, particularly within the context of active safety functions, several of which are now regarded as standard equipment on the majority of today's road vehicles.

Control systems for active safety began with the introduction of anti-lock braking systems (ABS) in the 1970's. ABS systems allow the driver to maintain control of the vehicle by preventing wheel lock-up during hard braking. More recently, higher-level active safety systems have become widespread, such as Electronic Stability Programs (ESP) which correct for excessive understeer and oversteer. Such systems, which govern the stability of the entire vehicle, may be described as Vehicle Dynamics Control (VDC) systems.

In addition to systems relating to vehicle stability, there is currently a great deal of development of systems relating to the interaction of the vehicle with the environment. An example of such a system which has recently entered series production is adaptive cruise control (ACC) [Bengtsson, 2001]. Another application, currently the subject of considerable research, is collision avoidance and mitigation [Jansson, 2005],

Presently, most production VDC systems are limited to yaw (steering plane) control. In recent years however, considerable attention has been paid to rollover accidents, caused either by extreme maneuvering or by the

vehicle striking an obstacle while skidding. Systems such as ESP have only a limited ability to mitigate such accidents. Because of this there exists an interest in the development of VDC systems capable of rollover mitigation. The investigation of such a system is the primary focus of this thesis.

Perhaps the most important piece of environmental information for vehicle dynamics control systems is the coefficient of friction between the tire and the road. The coefficient of friction effectively determines the maximum force available from the tires. Since the tire forces drive the dynamics of the vehicle, the available friction determines the range of maneuvers that can be performed. Obviously this is critical for systems dealing with active safety.

Due to its importance, a considerable amount of work has been done on the subject of friction estimation. In this thesis, development of a model-based approach to friction estimation is carried out.

There are are number of difficulties associated with automotive control and estimation which can transform even apparently simple tasks into considerable challenges. These difficulties include:

- **Uncertainty**. There is a large number of parameters, states and other variables, used by the controller, which either cannot be measured or are not known exactly. For cost reasons, sensors are typically kept to a minimum in production vehicles and as such it is often necessary to estimate variables, which gives rise to additional uncertainty.
- **Complexity**. Vehicles are complex systems, with many degrees of freedom. Control systems are often multivariable, with many actuators and output signals. The equations governing the behaviour of the system are often nonlinear, meaning that standard linear control design tools may not always be applicable.
- **Reliability and Robustness**. For production vehicles it is imperative that system operation can be guaranteed for a wide range of operating conditions. The high level of robustness required may mean that performance requirements are difficult to meet.

Research Projects

The research presented in this thesis has been carried out within the context of two projects. Work on VDC systems for rollover mitigation was done within *Complex Embedded Automotive Control Systems* (CEmACS), a European Union project comprising four academic and one industrial partner. Work on friction estimation is part of the *Road Friction Estimation II* (RFE II) project, part of the *Intelligent Vehicle Safety Systems* (IVSS) program, funded by the Swedish government.

CEmACS The CEmACS project [CEmACS, 2007] was a partnership between DaimlerChrysler Research and Technology, the Hamilton Institute, Lund University, Glasgow University and SINTEF, active between 2004 and 2007. The objective of CEmACS was to contribute to a systematic, modular, model-based approach for designing complex automotive control systems.

RFE II The Road Friction Estimation (RFE) II project [Vägverket, 2008b] is part of the Swedish Intelligent Vehicle Safety Systems (IVSS) program [Vägverket, 2008a]. IVSS is a joint venture involving Swedish public-sector agencies, companies and universities with an aim of shifting emphasis from passive safety solutions to active systems. The RFE II project is a continuation of the original RFE project, involving Volvo Car Corporation, Volvo AB, SAAB, Haldex Brake Products, and VTI (*Statens Väg och Trafikforskningsinstitut*). The project consists of several subprojects, of which model-based friction estimation is one.

The aims of RFE II include the extension of the region of operation of the friction estimators, extension to heavy vehicles, and implementation of change detection for situations involving sudden changes in road condition.

1.2 Motivation

Rollover Mitigation

Vehicle rollover accidents are typically very dangerous. Research by the National Highway Traffic Safety Administration (NHTSA) in the United States shows that rollover accidents are the second most dangerous form of accident in the Unites States, after head-on collisions. In 2000, 9,882 people were killed in the US in rollover accidents involving light vehicles [Forkenbrock *et al.*, 2002]. 8,146 of those were killed in accidents involving only a single vehicle.

Vehicle rollover accidents may be grouped into two categories, known as *tripped* and *untripped* rollovers. Tripped rollovers are caused by the vehicle coming into contact with an external obstacle. This type of rollover accident may sometimes be avoided by the use of ESP systems, which prevent oversteering and understeering, thus preventing collisions with external obstacles. Untripped rollovers are induced by extreme driving maneuvers, in which the forces at the tire-road contact point are sufficient to cause the vehicle to roll over.

While the majority of rollover accidents are tripped rollovers, it is clear that an active safety system capable of preventing untripped rollover accidents will save lives, and as such is worthy of investigation.

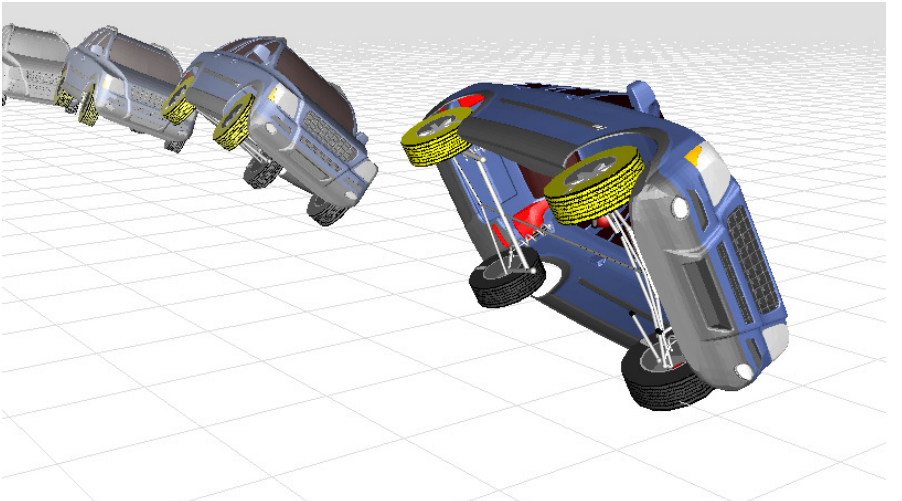

Figure 1.1 Simulation of a vehicle performing a 'fishhook' maneuver, with a road-tire friction coefficient of 1.2. The high friction allows large lateral tire forces which give rise to rollover.

Friction Estimation

The coefficient of friction between the tire and the road surface determines the maximum force available from the tire. Since the tire forces are the control input for VDC systems, knowledge of the friction is extremely important. Figures 1.1 and 1.2 illustrate the effects of performing the same deriving maneuver with different friction coefficients. In Figure 1.1, the friction coefficient is 1.2, not uncommon for good tires on dry asphalt. The high friction allows large lateral tire forces to be generated, causing rollover. In Figure 1.2, exactly the same maneuver leads to severe skidding when the friction coefficient is reduced to 0.6, a figure typical for wet asphalt. This example illustrates the importance of knowledge of the coefficient of friction for vehicle dynamics control applications.

1.3 Scope

The design of a complete vehicle dynamics control system involves many aspects of control engineering, including the selection of actuators, design of parameter estimation schemes and state observers, as well as the control design itself. The work presented in this thesis concentrates on the control design aspects. Only standard sensors are assumed.

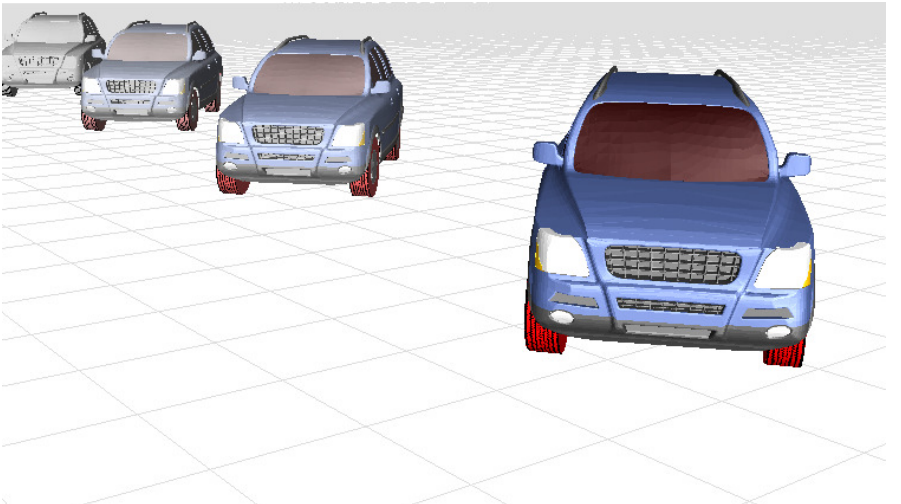

Figure 1.2 Simulation of a vehicle performing a 'fishhook' maneuver, with a road-tire friction coefficient of 0.6. The maneuver is the same as in Figure 1.1, but the lower friction means that the vehicle skids rather than rolls.

There is also considerable freedom in the choice of actuators to be used. Active steering, differential braking and active suspension, as well as combinations thereof may be considered. Active suspension is the most hardware-demanding actuator choice. Active steering and braking have been suggested for use in rollover prevention systems [Odenthal *et al.*, 1999]. However, active steering requires additional hardware, in the form of either a mechanical superposition actuator or a full steer-by-wire system. Differential braking, achieved by using a braking system with Electronic Brake force Distribution (EBD), is already in use in various production vehicles, and as such represents the most realistic actuator choice. This thesis uses differential braking as the actuator, although it will be seen that the methods developed may be easily extended to other types of actuators, as well as combinations of different actuators. The types of vehicle primarily considered in this work are small commercial vehicles such as vans and trucks. However, the ideas presented may be easily applied to other types of vehicle, such as larger trucks.

For slip-based friction estimation, there are a number of supporting systems required for generating the necessary signals. The operation of the majority of these systems are described in this thesis. A key theme of the work is that only existing sensors should be used. This is an important feature with regards to the automotive industry, where additional sensors are seldom allowed due to reasons of cost.

1.4 Contributions

The main contributions of this thesis can be summarised as:

- Development of a high-level rollover mitigation controller capable of simultaneously stabilizing roll and yaw motion
- Development of a general control allocation structure for conversion of resultant force and moment commands to individual wheel braking forces, including constraints, based on convex optimization
- Investigation of methods of solution of the control allocation optimization problem
- Proposition of a new numerical method for solution of bound constrained least squares problems
- Investigation of model-based friction estimation, focusing on the development of existing methods in aspects including change detection, operating region and vehicle type

1.5 Outline

Appropriate models are a prerequisite for successful model-based control design. The thesis begins with two chapters dedicated to vehicle modeling. A selection of tire models, which will be used in different contexts throughout the thesis, is presented in Chapter 2. Vehicle and chassis modeling is treated in Chapter 3. A number of models of varying complexity are derived. Chapter 4 deals with analysis of the rollover problem, as well as methods for detection of an oncoming rollover event. In Chapter 5 the design methodology is presented, and control design is performed. The concept of *control allocation*, on which the control strategy is based, is presented in Chapter 6. In Chapter 7, a new numerical method for solving control allocation problems is developed. Evaluation of the rollover mitigation controller using industrial vehicle simulation software is presented in Chapter 8. A sensor fusion method for improvement of the measured yaw rate signal is presented in Chapter 9. Model-based friction estimation is covered in Chapter 10.

1.6 Publications

The work on rollover mitigation, described in Chapters 5 through 8, is based on the following publications:

Schofield, B., T. Hägglund and A. Rantzer (2006): "Vehicle Dynamics Control and Controller Allocation for Rollover Prevention." In *Proceedings of the IEEE International Conference on Control Applications*, Munich, Germany.

Schofield, B. (2006): "Vehicle dynamics control for rollover prevention." Licentiate Thesis ISRN LUTFD2/TFRT--3241--SE. Department of Automatic Control, Lund University, Sweden.

Schofield, B. (2008): "On active set algorithms for solving bound-constrained least squares control allocation problems." In *Proceedings of the 2008 American Control Conference*. Seattle, USA.

Schofield, B. and T. Hägglund (2008): "Optimal control allocation in vehicle dynamics control for rollover mitigation." In *Proceedings of the 2008 American Control Conference*. Seattle, USA.

2

Tire Modeling

All road vehicles interact with the road surface via tires. More specifically, the tires are responsible for generating those forces which are required to alter the vehicle's speed and course according the driver's inputs.

The physical mechanisms by which tires function are complicated. The approaches taken when modeling tire behaviour may be divided into two groups, empirical methods and physical methods. Empirical methods are concerned with obtaining model structures which may be easily fitted to measurement data, while physical methods are primarily concerned with explaining the mechanisms that determine tire behaviour. The level of detail involved in different tire models varies enormously, and selection of an appropriate model for a particular application is not always easy.

In this chapter, only static tire models are considered. Dynamic models exist, but for the applications in this thesis the considerable added complexity of dynamic models is not warranted. The models presented here comprise a small subset of the models that have been developed. The reader is referred to [Pacejka, 2002; Svendenius, 2007a] for more detailed treatment of tire modeling.

2.1 Slip

In order to generate forces while rolling, the tire must *slip*. Slip occurs in different planes of the tire's motion. Longitudinal slip will be considered first.

Effective Rolling Radius When the tire is rolling freely (no driving or braking force applied), the *effective rolling radius* R_e may be defined as follows:

$$R_e = \frac{v_x}{\omega} \tag{2.1}$$

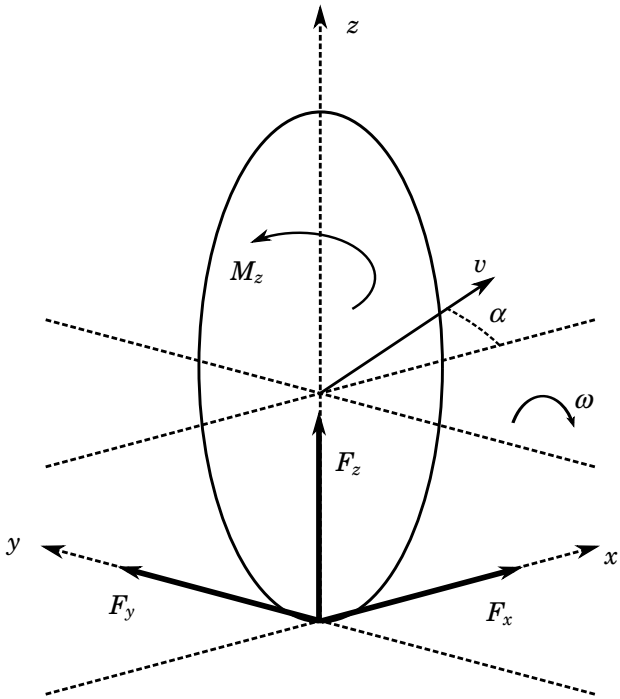

Figure 2.1 Diagram of a wheel showing its local axes, the components of the forces acting at the contact point, the angular velocity ω and the translational velocity v.

where v_x is the longitudinal velocity of the wheel centre, and ω is the wheel's angular velocity. Figure 2.1 illustrates the coordinate system of the wheel.

Longitudinal Slip When a driving or braking torque is applied to the wheel, longitudinal slip develops. The *slip velocity* is the relative velocity of the tire contact patch with the ground, and is given by $v_x - v_c$ where $v_c = R_e\omega$ is the circumferential velocity of the tire. The slip is obtained by normalizing the slip velocity. Two definitions of longitudinal slip are commonly used. The first, denoted λ, is normalized by the longitudinal velocity v_x:

$$\lambda = -\frac{v_x - R_e\omega}{v_x} \tag{2.2}$$

The longitudinal slip λ is defined in such a way that a positive (accelerating) force gives a positive slip value. An alternative definition denoted

as *physical slip* σ_x, is defined as the slip velocity normalized by the circumferential velocity:

$$\begin{aligned}\sigma_x &= \frac{v_x - v_c}{v_c} \\ &= \frac{v_x - R_e\omega}{R_e\omega}\end{aligned} \tag{2.3}$$

Lateral Slip The lateral slip is the ratio of the wheel's velocities in the y and x directions. It can be defined through the lateral slip angle α as follows:

$$\tan\alpha = -\frac{v_y}{v_x} \tag{2.4}$$

The definition is such that positive side forces correspond to positive slip angles. An alternative definition of the lateral slip is:

$$\sigma_y = \frac{v_y}{v_c} \tag{2.5}$$

An additional slip quantity is known as spin, and is caused by rotation of the wheel about the z axis. The wheel camber angle γ, defined as the angle between the wheel's xz plane and the vertical, influences this quantity.

The Tire System

The tire may be regarded as a system having the slip components as inputs, and the forces F_x, F_y and moment M_z as outputs [Pacejka, 2002]:

$$\begin{aligned}F_x &= F_x(\lambda, \alpha, \gamma, F_z) \\ F_y &= F_y(\lambda, \alpha, \gamma, F_z) \\ M_z &= M_z(\lambda, \alpha, \gamma, F_z)\end{aligned}$$

where F_z is the wheel load. Additionally, the camber angle γ will be assumed to be zero, reducing the dependence of the forces to three variables. The effects of nonzero camber angle are investigated in [Svendenius, 2007a].

Pure and Combined Slip

Pure slip is used to denote the case where a given slip quantity appears in isolation, that is, when all other slip quantities are zero. Combined slip is the term used to describe the situation in which multiple slip types occur.

The combined slip case is complex since there are limits on the maximum resultant force available from the tire.

A standard approach to modeling combined slip is to select appropriate models for pure slip and then use the results in a model for combined slip. In this way it is possible to mix empirical and physical models to suit the application. An example of this is given in [Svendenius and Gäfvert, 2004a].

2.2 Empirical Models

Linear Approximations

A simple linear approximation can be made in the pure slip case by examining the gradient of the force-slip characteristic. Due to the shape of the curves, the approximations are valid only for small slip values. The approximations are given by:

$$F_x \approx C_\lambda \lambda \tag{2.6a}$$

$$F_y \approx C_\alpha \alpha \tag{2.6b}$$

$$M_z \approx -C_M \alpha \tag{2.6c}$$

Here, C_λ is the longitudinal (or braking) stiffness, C_α is the lateral (or cornering) stiffness and C_M is known as the aligning stiffness.

The Magic Formula

A more accurate expression for the lateral force F_y is given by the so-called *Magic Formula* [Pacejka, 2002]:

$$F_y = D \sin[C \arctan\{B\alpha - E(B\alpha - \arctan(B\alpha))\}] \tag{2.7}$$

where:

$B = \dfrac{C_\alpha}{CD}$ is the stiffness factor

$D = \mu F_z = F_{y,\text{peak}}$ is the peak factor

$C_\alpha = c_1 \sin\left(2 \arctan\left(\dfrac{F_z}{c_2}\right)\right)$

C, E are shape factors

c_1 is the maximum cornering stiffness

c_2 is the load at maximum cornering stiffness

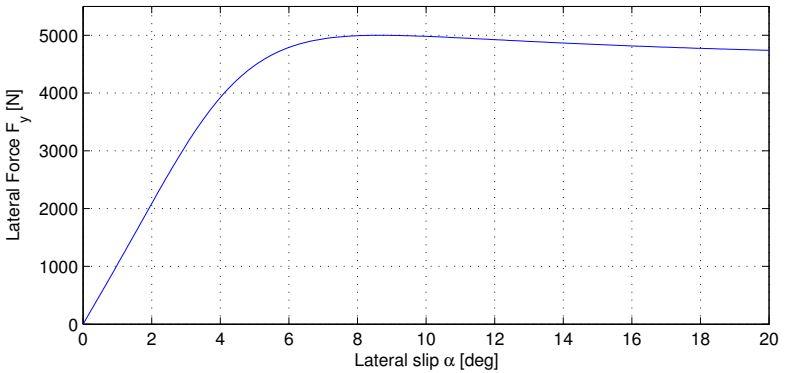

Figure 2.2 The relation between lateral force F_y and slip angle α obtained using the Magic Formula (2.7).

Figure 2.2 illustrates the relation between slip angle and lateral force, for typical parameter values.

2.3 The Friction Ellipse

A common technique used for modelling combined slip is based on the idea of the *friction ellipse*, shown in Figure 2.3. The assumption is that the longitudinal and lateral forces acting on each tire cannot exceed their maximum values $F_{x,max}$ and $F_{y,max}$, and the resultant force thus lies on the ellipse given by:

$$\left(\frac{F_y}{F_{y,max}}\right)^2 + \left(\frac{F_x}{F_{x,max}}\right)^2 = 1 \tag{2.8}$$

The friction ellipse idea can be confirmed empirically by plotting curves of the forces F_x and F_y for different slip values and noting that the envelope of the curves approximates an ellipse [Wong, 1993].

The maximum lateral force $F_{y,max}$ may be taken to be given by the magic formula (2.7). The maximum longitudinal force is taken to be given by the product of the coefficient of friction between the tire and the road surface (μ) and the normal force F_z acting on the tire:

$$F_{x,max} = \mu F_z$$

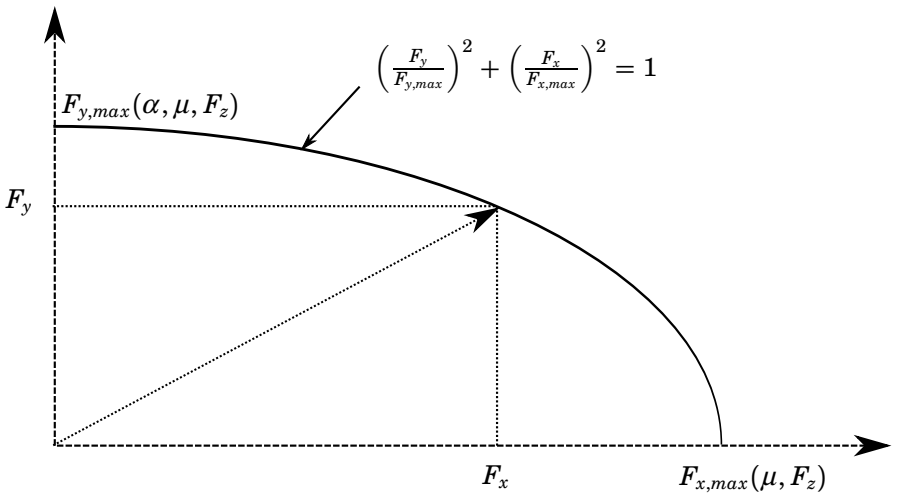

Figure 2.3 The friction ellipse, showing maximum lateral and longitudinal forces, the resultant force and its components.

2.4 The Brush Model

A commonly used physical modeling approach is based on the so-called *brush model*. This approach is attractive since it gives good insight into the operation, as well as providing a very useful model structure.

In this section a brief derivation of the brush model for pure and combined slip is presented. Zero camber angle is assumed. More detailed presentations of the brush model can be found in [Pacejka, 2002] and [Svendenius, 2007a].

Validation of the brush model using measurement data from numerous tire and road surface combinations is performed in [Svendenius *et al.*, 2007], [Svendenius, 2007b] and [Svendenius, 2007c].

The brush model considers the tire to be composed of many small elements or bristles, which deform upon contact with the road surface. The contact patch, illustrated in Figure 2.4 denotes the region of the tire surface which is in contact with the road, and is of length $2a$. Upon entering the contact patch, a bristle is undeformed. The deformation of the bristle increases as it moves within the contact patch. Initially, the end of the bristle in contact with the road is stationary with respect to the road. This situation is maintained until the static friction μ_s can no longer provide the required force to deform the bristle further. At this point the bristle begins to slide along the road surface. The contact patch may therefore be

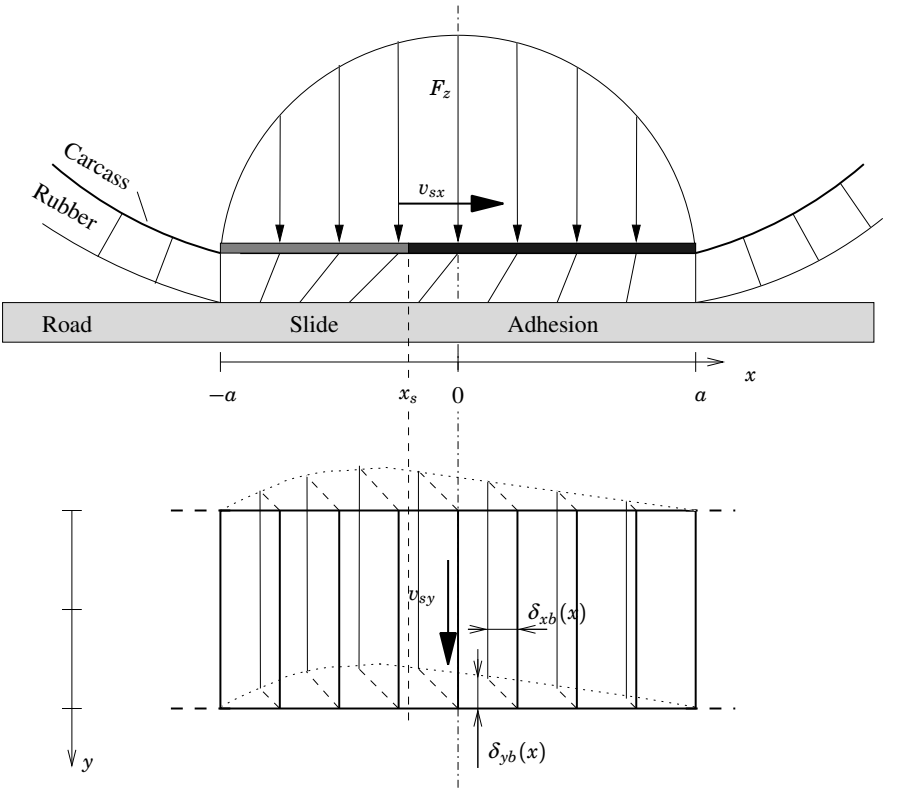

Figure 2.4 Illustration of the brush model, showing the deformation of the bristles in both x and y directions.

divided into two regions, known as the adhesive region and sliding region.

Determining the Size of the Adhesion Region

The sizes of these regions may be determined by considering the limitations imposed by the friction. For an infinitesimally small bristle at position x in the contact patch, let the normal force acting upon it be $dF_z(x)$. The longitudinal and lateral forces acting on the bristle are constrained by the relation:

$$\left(\frac{dF_{ax}(x)}{\mu_{sx} dF_z(x)}\right)^2 + \left(\frac{dF_{ay}(x)}{\mu_{sy} dF_z(x)}\right)^2 \leq 1 \tag{2.9}$$

The forces $dF_{ax}(x)$ and $dF_{ay}(x)$ are given by:

$$\begin{aligned} dF_{ax}(x) &= c_x dx \delta_x(x) \\ dF_{ay}(x) &= c_y dx \delta_y(x) \end{aligned} \tag{2.10}$$

where c_x and c_y are the longitudinal and lateral bristle stiffnesses per unit length, and $\delta_x(x)$ and $\delta_y(x)$ are the bristle deformations at the position x. The deformations are determined by the slip:

$$\begin{aligned} \delta_x(x) &= -\sigma_x(a-x) \\ \delta_y(x) &= -\sigma_y(a-x) \end{aligned} \tag{2.11}$$

By introducing the pressure distribution $q(x)$, the normal force may be written as $dF_z(x) = q(x)dx$. The friction constraint (2.9) can then be written as:

$$\sqrt{\left(\frac{c_x\sigma_x}{\mu_{sx}}\right)^2 + \left(\frac{c_y\sigma_y}{\mu_{sy}}\right)^2}(a-x) \leq q(x) \tag{2.12}$$

The pressure distribution $q(x)$ can be described by a parabolic function:

$$q(x) = \frac{3F_z}{4a}\left(1 - \frac{x^2}{a^2}\right) \tag{2.13}$$

By using (2.12) and (2.13), the point x_s at which adhesion ceases and sliding starts may be calculated from:

$$\sqrt{\left(\frac{c_x\sigma_x}{\mu_{sx}}\right)^2 + \left(\frac{c_y\sigma_y}{\mu_{sy}}\right)^2}(a-x_s) = \frac{3F_z}{4a}(a-x_s)(a+x_s) \tag{2.14}$$

giving:

$$x_s = \frac{4a^3}{3F_z}\sqrt{\left(\frac{c_x\sigma_x}{\mu_{sx}}\right)^2 + \left(\frac{c_y\sigma_y}{\mu_{sy}}\right)^2} - a \tag{2.15}$$

Normalized Slip

Clearly, $x_s \in [-a, a]$. The case $x_s = a$ corresponds to full sliding. For pure slip, limit slips σ_x^o and σ_y^o can be defined, corresponding to the slip value at which full sliding begins. These limit slips can be calculated from (2.15):

$$\begin{aligned} \sigma_x^o &= \frac{3F_z\mu_{sx}}{2a^2c_x} \\ \sigma_y^o &= \frac{3F_z\mu_{sy}}{2a^2c_y} \end{aligned} \tag{2.16}$$

By normalizing the slip components with respect to these limit slips, a normalized slip quantity ψ can be defined:

$$\psi = \sqrt{\left(\frac{\sigma_x}{\sigma_x^o}\right)^2 + \left(\frac{\sigma_y}{\sigma_y^o}\right)^2} \tag{2.17}$$

Values of ψ less than 1 correspond to a combination of adhesion and sliding in the contact patch. When $\psi = 1$, full sliding occurs. The breakaway point can be written in terms of ψ:

$$x_s = (2\psi - 1) - a \tag{2.18}$$

Adhesive Forces

The adhesive forces in the x and y directions are given by the integrals:

$$\begin{aligned} F_{ax} &= -c_x\sigma_x \int_{x_s}^{a} (a - x)dx \\ F_{ay} &= -c_y\sigma_y \int_{x_s}^{a} (a - x)dx \end{aligned} \tag{2.19}$$

Evaluating these integrals using (2.18) gives:

$$\begin{aligned} F_{ax} &= \begin{cases} -2a^2c_x\sigma_x(1-\psi)^2 & \psi < 1 \\ 0 & \psi \geq 1 \end{cases} \\ F_{ay} &= \begin{cases} -2a^2c_y\sigma_y(1-\psi)^2 & \psi < 1 \\ 0 & \psi \geq 1 \end{cases} \end{aligned} \tag{2.20}$$

The conditional statements in (2.20) reflect the fact that the adhesive forces cease to exist when full sliding occurs.

Figure 2.5 illustrates the adhesive force for the case of pure longitudinal slip. The incremental force depends linearly on position within the contact patch, and increases until the maximum available force, determined by the static friction and the pressure distribution, is achieved. This intersection determines the size of the adhesion region. Outside of this region, sliding occurs.

Sliding Forces

The normal force acting on the sliding region is:

$$\begin{aligned} F_{sz} &= \int_{-a}^{x_s} q(x)dx \\ &= F_z\psi^2(3 - 2\psi) \end{aligned} \tag{2.21}$$

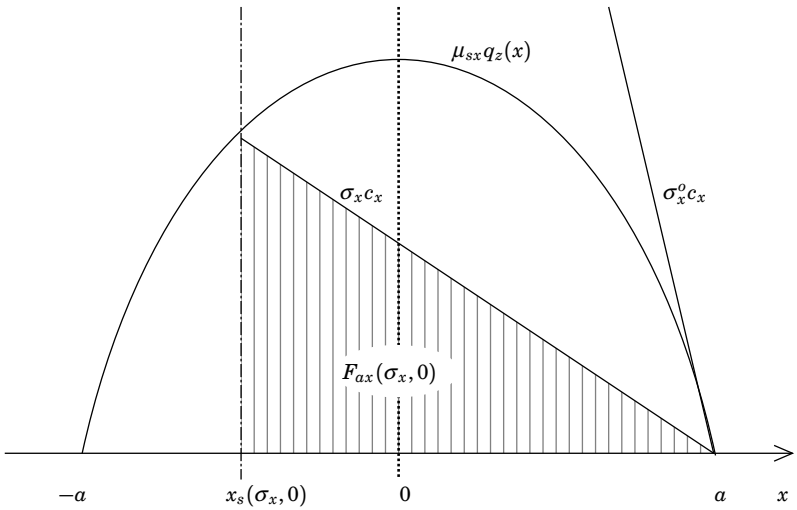

Figure 2.5 Illustration of the adhesive force in the case of pure longitudinal slip. The adhesive force increases linearly with bristle deflection until the maximum force, given by $\mu_{sx} q_z(x)$, is reached, at which point sliding begins.

If the sliding (dynamic) friction μ_d is assumed to be isotropic, the sliding forces in the x and y directions are:

$$\begin{aligned} F_{sx} &= -\cos\beta \mu_d F_{sz} = -\cos\beta \mu_d F_z \psi^2 (3 - 2\psi) \\ F_{sy} &= -\sin\beta \mu_d F_{sz} = -\sin\beta \mu_d F_z \psi^2 (3 - 2\psi) \end{aligned} \tag{2.22}$$

where β is the sideslip angle.

The sliding force for pure longitudinal slip is illustrated in Figure 2.6. Its magnitude is determined by the sliding friction and the size of the adhesion region.

Resultant Force

The resulting tire force is the sum of the adhesive and sliding forces:

$$\begin{aligned} F_x &= F_{ax} + F_{sx} \\ F_y &= F_{ay} + F_{sy} \end{aligned} \tag{2.23}$$

Using the expressions (2.20) and (2.22) gives:

$$\begin{aligned} F_x &= -2a^2 c_x \sigma_x (1 - \psi)^2 - \cos\beta \mu_d F_z \psi^2 (3 - 2\psi) \\ F_y &= -2a^2 c_y \sigma_y (1 - \psi)^2 - \sin\beta \mu_d F_z \psi^2 (3 - 2\psi) \end{aligned} \tag{2.24}$$

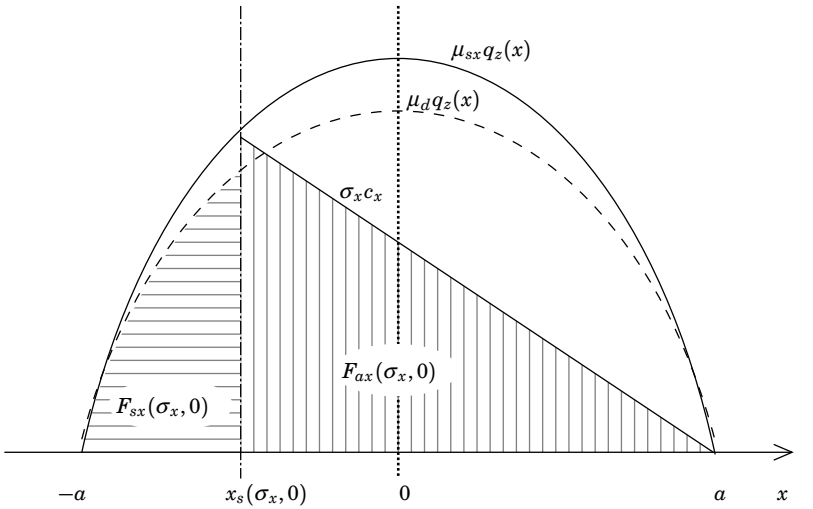

Figure 2.6 Illustration of the effects of static and dynamic friction for pure longitudinal slip. The adhesive force is as shown in Figure 2.5. The sliding force is determined by the $\mu_d q_z(x)$.

Pure Slip

In the case of pure longitudinal slip ($\sigma_y = 0$), the brush model yields the following longitudinal force:

$$F_x = -2a^2 c_x \sigma_x (1-\psi)^2 - \mu_d F_z \psi^2 (3 - 2\psi) \tag{2.25}$$

With $\sigma_y = 0$, the normalized slip becomes $\psi = |\sigma_x|/\sigma_x^o$. Introducing the longitudinal tire stiffness $C_x = 2a^2 c_x$, the longitudinal force becomes:

$$F_x = -C_x \sigma_x + \frac{C_x^2 \sigma_x |\sigma_x|}{3\mu_d F_z} - \frac{C_x^3 \sigma_x^3}{27 \mu_d^2 F_z^2} \tag{2.26}$$

Combined Slip

For the case of combined slip (both σ_x and σ_y nonzero) the expressions become more complex. The sideslip angle β can be replaced by writing:

$$\cos\beta = \frac{\sigma_x}{\sqrt{\sigma_x^2 + \sigma_y^2}} \tag{2.27}$$

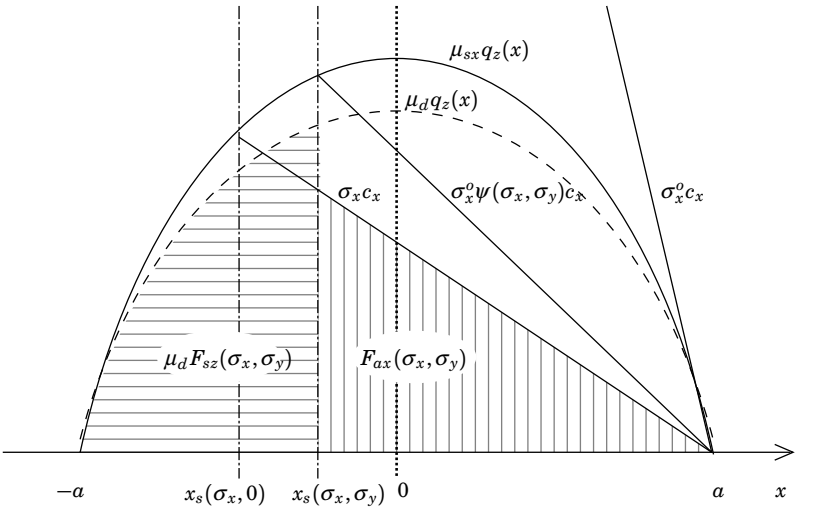

Figure 2.7 Illustration of the effects of combined slip. The adhesive region is reduced in size compared to Figure 2.6 due to the lateral deformation of the bristles.

This results in the following relation for longitudinal force:

$$F_x = -C_x\sigma_x + 2C_x\sigma_x\psi - C_x\sigma_x\psi^2 - \frac{3F_z\psi^2\sigma_x}{\sqrt{\sigma_x^2+\sigma_y^2}} + \frac{2F_z\psi^3\sigma_x}{\sqrt{\sigma_x^2+\sigma_y^2}} \tag{2.28}$$

Figure 2.7 shows the effects of combined slip on the distribution of adhesive and sliding forces. The adhesive region is reduced in size due to the addition of lateral bristle deformation. Figure 2.8 shows a typical brush model characteristic, illustrating the effects of combined slip, as well as the contributions from the adhesive and sliding forces.

2.5 Summary

In this chapter several tire models have been presented. The variation in complexity is large, which is necessary in order to serve the requirements of different applications. The simplest form of model, a linear relationship between slip and force, will be used in the following chapter to derive a very simple, but nevertheless extremely widely used vehicle model. More complex empirical models such as the magic formula can be used when the nonlinear behaviour of the tire must be taken into account. Empirical

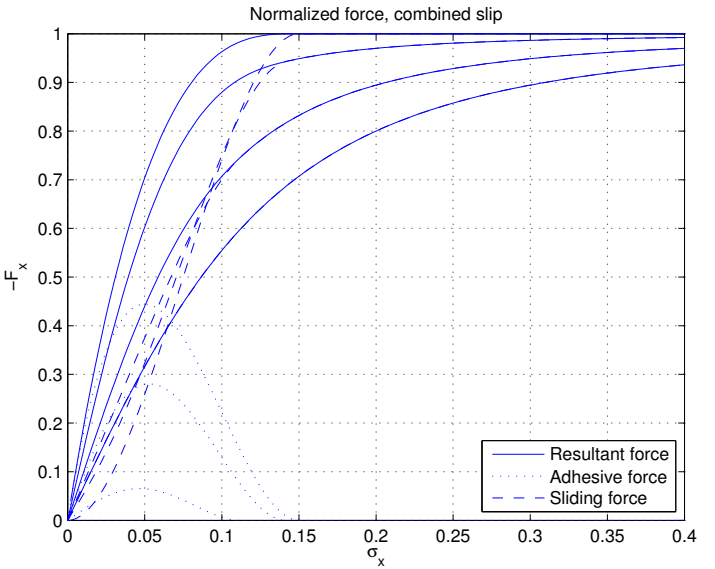

Figure 2.8 Example of a brush model characteristic for combined slip. The plot shows the longitudinal force, normalized with the normal force, for $\sigma_y \in [0, 0.05, 0.1, 0.15]$. The contributions from the adhesive and sliding forces are shown as dotted and dashed lines respectively.

models do however require that a number of (often non-physical) parameters be known.

The friction ellipse approach to combined slip modeling will be used extensively in this thesis, in situations where the relation between lateral and longitudinal forces are critical.

The brush model will be used for friction estimation. It is well suited to this task since there are only a small number of parameters, all of which are physical.

3

Vehicle Modeling

The tire models presented in Chapter 2 describe the forces generated by the interaction of the tire and the road surface. These forces can be seen as the inputs to the dynamics determining the motion of the vehicle. In this chapter, dynamical models of varying complexity are presented. As was the case with tire modeling, different vehicle models are required to suit different applications.

The models presented here are not new, nevertheless a detailed derivation of the more complex two-track models will be carried out. This serves a number of purposes. The derivation of a model is very useful for gaining understanding of the underlying physics, and allows more rigorous motivation of any approximations which may be made later. In addition, models presented in the literature often omit aspects of the derivation.

3.1 Coordinate Systems

In order to facilitate the derivation of the equations of motion, it is useful to define a number of coordinate systems. This allows a more systematic approach to modeling, which is particularly important when dealing with more complex models. Figure 3.1 illustrates a right-hand coordinate system. *Roll*, *pitch* and *yaw* are defined as rotations around the x, y and z axes respectively.

Let S_i denote an earth-fixed inertial right-hand coordinate system, with the z axis oriented upwards relative to the earth. Denote with S_v a coordinate system attached to the vehicle, which rotates with angular velocity $\omega_i^v = \begin{pmatrix} 0 & 0 & \dot{\psi} \end{pmatrix}^T$ about the z axis, and translates in the xy plane with velocity $\begin{pmatrix} v_x & v_y & 0 \end{pmatrix}^T$. In addition, let S_c denote a chassis coordinate system, rotated an angle θ about the y axis of the S_v frame. The x axis of the chassis coordinate system is the roll axis, about which the roll angle ϕ is defined. Finally, denote with S_b a body coordinate system which

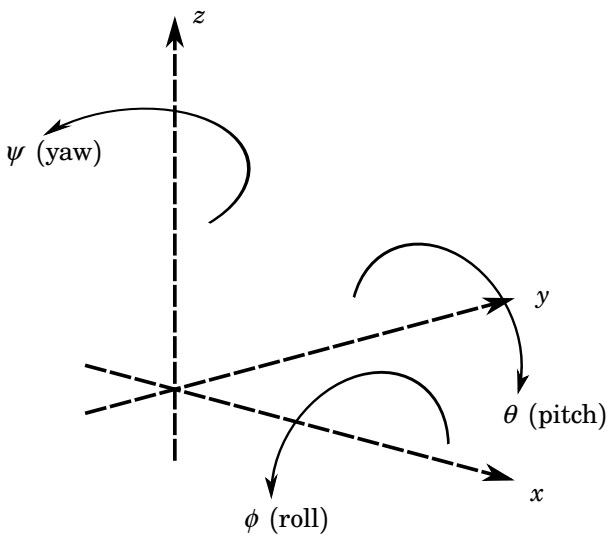

Figure 3.1 The coordinate axes. The x axis corresponds to the longitudinal axis of the vehicle, positive in the forward direction. The y axis corresponds to the lateral axis, positive to the left. Roll, pitch and yaw are defined around the x, y and z axes respectively.

rotates around the x axis of the chassis system with angular velocity $\omega_c^b = (\dot{\phi} \quad 0 \quad 0)^T$. The position of the centre of gravity (CG) in the body frame is given by $P_{CG}^b = (0 \quad 0 \quad h)^T$.

In the derivations of the models it will be necessary to transform between these coordinate systems. To this end, rotation matrices may be defined. The rotation matrix from the inertial system S_i to the vehicle system S_v is defined as:

$$R_i^v(\psi) = \begin{pmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{3.1}$$

The rotation matrix from the vehicle system S_v to the chassis system S_c is given by:

$$R_v^c(\theta) = \begin{pmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{pmatrix} \tag{3.2}$$

The rotation matrix from the chassis system S_c to the body system S_b is given by:

$$R_c^b(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{pmatrix} \tag{3.3}$$

A rotation from the inertial system S_i to the body system S_b can be expressed as the product of the above rotation matrices:

$$R_i^b(\psi, \theta) = R_c^b(\phi) R_v^c(\theta) R_i^v(\psi) \tag{3.4}$$

Since the rotation matrices are skew-symmetric, the relation:

$$RR^T = R^T R = I$$

may be used to obtain the rotation matrices for rotations in the opposite direction. Due to the special structure of the matrices, this may be thought of as rotation by a negative angle:

$$R_i^v(\psi) = R_v^i(-\psi)$$

When dealing with moving coordinate systems, the expressions for velocities and accelerations become more complex due to the need to express these quantities in inertial frames when they are to be used in equations of motion. Consider a point described by the vector P relative to a body-fixed frame. Assume that the origin of the body system is translated by a vector R from the origin of an inertial frame, and that the body frame rotates with angular velocity ω relative to the inertial frame. The expression for the velocity of the point P in the inertial frame is given by:

$$\left.\frac{dP}{dt}\right|_i = \frac{dR}{dt} + \left.\frac{dP}{dt}\right|_b + \omega_i^b \times P \tag{3.5}$$

Similarly, the acceleration of the point P in the inertial frame is given by:

$$\left.\frac{d^2P}{dt^2}\right|_i = \frac{d^2R}{dt^2} + \left.\frac{dP^2}{dt^2}\right|_b + \dot{\omega}_i^b \times P + 2\omega_i^b \times \left.\frac{dP}{dt}\right|_b + \omega_i^b \times (\omega_i^b \times P) \tag{3.6}$$

3.2 Linear Single-Track Model

The simplest vehicle model is the linear single-track model, also known as the bicycle model, which is obtained by approximating the front and

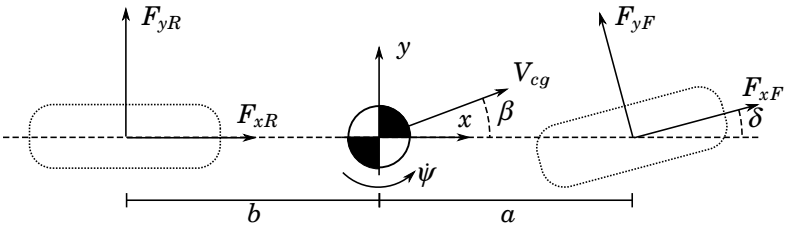

Figure 3.2 Single-track model, showing the combined front and rear tire forces, the steering angle δ, the yaw rate $\dot{\psi}$, and the vehicle sideslip angle β.

rear pairs of wheels as single wheels and linearizing the equations. The model is illustrated in Figure 3.2. Assuming that the steering angle δ is small, the equations of motion are given by [Pacejka, 2002]:

$$\begin{aligned} m(\dot{v}_y + v_x\dot{\psi}) &= F_{yF} + F_{yR} \\ I_{zz}\ddot{\psi} &= aF_{yF} - bF_{yR} \end{aligned}$$

where F_{yF} and F_{yR} are the combined front and rear lateral tire forces, I_{zz} is the moment of inertia around the z axis, a and b are the distances from the front and rear wheels to the center of gravity, and $\dot{\psi}$ is the yaw rate. The slip angles of the front and rear wheels α_F and α_R can be approximated as:

$$\alpha_F \approx \delta - \frac{1}{v_x}(v_y + a\dot{\psi}) \tag{3.7a}$$

$$\alpha_R \approx -\frac{1}{v_x}(v_y - b\dot{\psi}) \tag{3.7b}$$

Linear approximations of the tire forces, as described in (2.6a), may be used to obtain simple expressions for the front and rear lateral tire forces:

$$F_{yF} \approx C_F\alpha_F \tag{3.8a}$$

$$F_{yR} \approx C_R\alpha_R \tag{3.8b}$$

where C_F is the front cornering stiffness and C_R is the rear cornering stiffness.

The model becomes linear and time invariant if it is assumed that the forward velocity v_x is constant. This results in a two degree of freedom model, with the lateral velocity v_y and yaw rate $\dot{\psi}$ as states. The input is the steering angle δ. The system may be written on state space form as:

$$\begin{pmatrix} \dot{v}_y \\ \ddot{\psi} \end{pmatrix} = \begin{pmatrix} -\frac{C_F+C_R}{mv_x} & \frac{bC_R-aC_F}{mv_x} - v_x \\ \frac{bC_R-aC_F}{I_{zz}v_x} & -\frac{a^2C_F+b^2C_R}{I_{zz}v_x} \end{pmatrix} \begin{pmatrix} v_y \\ \dot{\psi} \end{pmatrix} + \begin{pmatrix} \frac{C_F}{m} \\ \frac{aC_F}{I_{zz}} \end{pmatrix} \delta \tag{3.9}$$

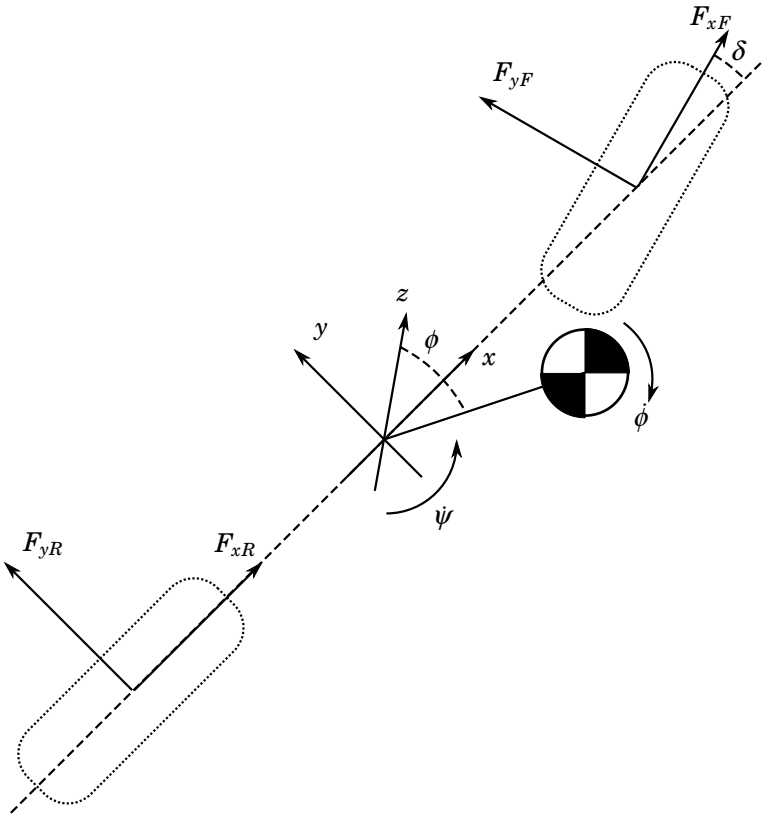

Figure 3.3 Single-track model with roll dynamics.

It should be noted that since the approximations made in the derivation of the model are based on the assumption of small steering angles, it may be necessary to perform some saturation of the outputs, based on the maximum available friction force.

Single-Track Model with Roll Dynamics

Roll dynamics and non-constant longitudinal velocity can be incorporated into a single-track model. The equations of motion are augmented by a

torque balance around the x axis. The equations are:

$$\begin{aligned} m(\dot{v}_y + v_x\dot{\psi}) &= F_{y,F} + F_{y,R} \\ I_{zz}\ddot{\psi} &= aF_{y,F} - bF_{y,R} \\ I_{xx}\ddot{\phi} + K_\phi\dot{\phi} + C_\phi\phi &= mh(\dot{v}_y + v_x\dot{\psi}) \end{aligned}$$

where K_ϕ and C_ϕ are damping and spring coefficients respectively. The slip angles are the same as in (3.7a), and the tire forces are considered to be linearly related to the slip angles as in (3.8a).

This model takes as inputs the steering angle δ and the longitudinal velocity v_x, and has roll angle ϕ, lateral velocity v_y, yaw rate $\dot{\psi}$ and vehicle sideslip angle β as outputs.

3.3 Two-Track Model

In order to incorporate the effects of the individual tire forces, as well as suspension and a more accurate representation of the roll dynamics, a two-track model can be used, shown in Figure 3.4. The suspension is modeled as a torsional spring and damper system acting around the roll axis, illustrated in Figure 3.5. In this way, the pitch dynamics of the vehicle are ignored. The resulting model has four degrees of freedom, namely translational motion along the x and y axis, as well as rotational motion about the x axis (roll) and the z axis (yaw).

In this section two modeling approaches will be used. The Newton-Euler approach involves setting up and solving Newton's equations for translational motion and Euler's equations for rotational motion. This approach will be used for deriving a model based on the assumption of a roll axis coinciding with the vehicle x axis. This gives a model on ordinary differential equation (ODE) form. The Euler-Lagrange method involves choosing a set of configuration coordinates and setting up the Lagrangian of the system. This method will be used for the case where a constant nonzero pitch angle is assumed, giving a model on differential algebraic equation (DAE) form. Model validation is performed with the same vehicle simulation software as will be used to evaluate the controllers in the following chapters.

Derivation of Tire Forces

In the derivation of the model it will be convenient to express the tire forces acting on the vehicle as resultant forces in the x and y directions of the S_v frame as well as a resultant moment about the z axis. These forces and moments will be referred to as *generalized forces*. By considering

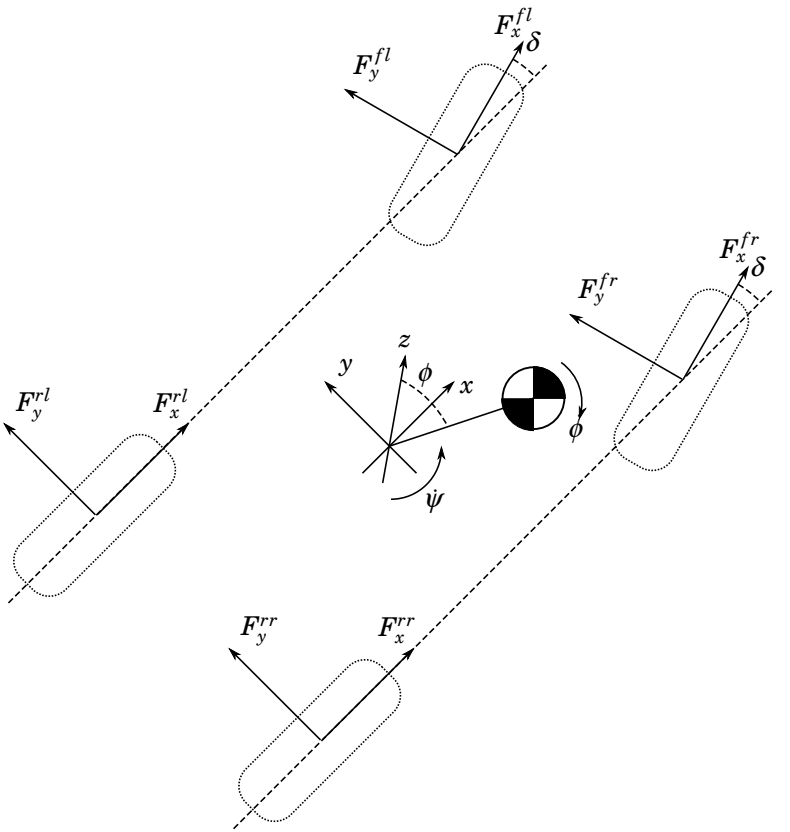

Figure 3.4 Two-track model with roll dynamics.

Figure 3.6, the following expressions relating the individual tire forces to the generalized forces are obtained:

$$F_{xT} = F_x^{rl} + F_x^{rr} + (F_x^{fl} + F_x^{fr})\cos\delta - (F_y^{fl} + F_y^{fr})\sin\delta \tag{3.10a}$$

$$F_{yT} = F_y^{rl} + F_y^{rr} + (F_y^{fl} + F_y^{fr})\cos\delta + (F_x^{fl} + F_x^{fr})\sin\delta \tag{3.10b}$$

$$\begin{aligned} M_T =& (F_y^{fl} + F_y^{fr})a\cos\delta - (F_y^{rl} + F_y^{rr})b + (F_x^{fl} + F_x^{fr})a\sin\delta + \\ & (F_x^{rr} + F_x^{fr}\cos\delta + F_y^{fl}\sin\delta - F_x^{rl} - F_x^{fl}\cos\delta - F_y^{fr}\sin\delta)l \end{aligned} \tag{3.10c}$$

where δ is the steering angle (measured at the wheels).

Newton-Euler Modeling

In this section a four degree of freedom model is derived using Newton-Euler modeling. The external forces acting on the vehicle are expressed

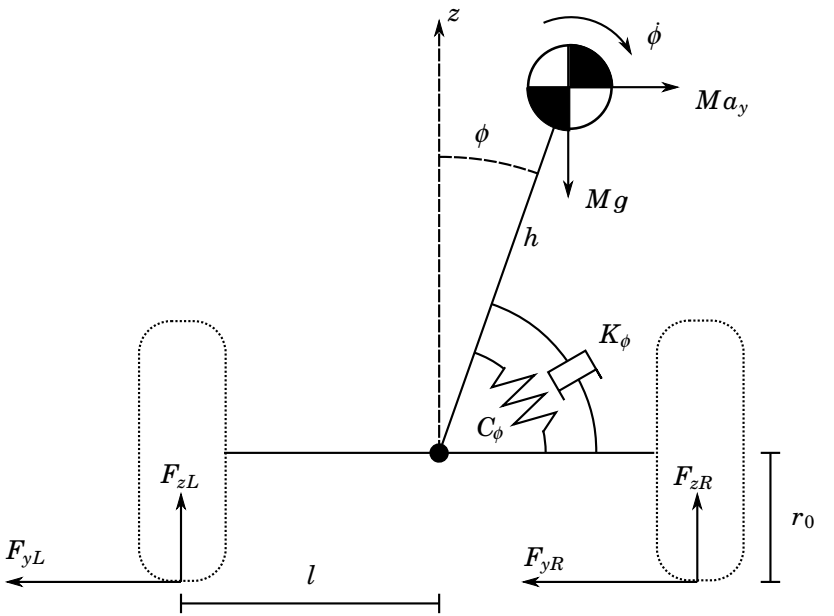

Figure 3.5 The two-track model in the vertical plane, showing suspension modeled as a torsional spring and damper.

in the vehicle coordinate system, so it is natural to write the equations of motion in this system.

Angular Motion Euler's equation states that the sum of the external torque acting on a system is given by the rate of change of angular momentum:

$$\tau = \frac{d(I^v \omega^s)}{dt} \tag{3.11}$$

where τ is the external torque or moment applied to the system, I^v is the inertia tensor relative to the coordinate frame in which the equations are to be derived, and ω^s is the spacial angular velocity.

Since the vehicle frame is rotating, it is not an inertial frame and (3.11) must be modified as in (3.5). The rate of change of angular momentum is given by:

$$\tau = \frac{d}{dt}(I^v \omega^s)\bigg|_v + \omega_i^v \times I^v \omega^s \tag{3.12}$$

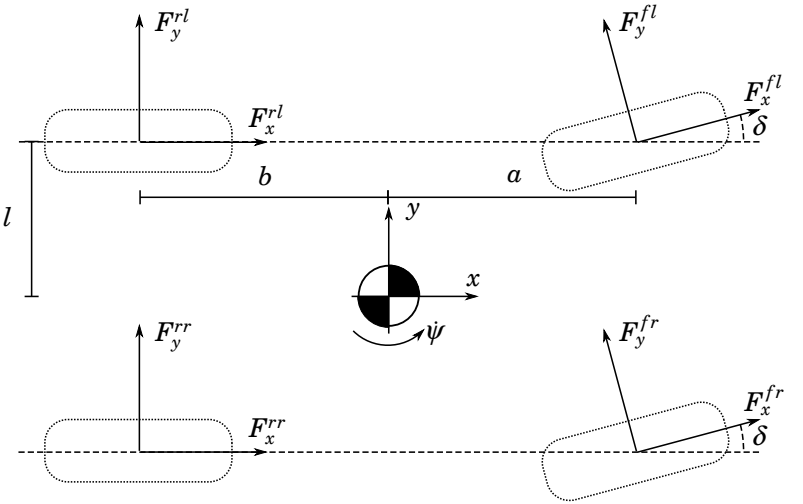

Figure 3.6 Planar chassis model, showing the horizontal components of the tire forces.

where ω_i^v is the angular velocity of the vehicle coordinate system relative to the inertial system. This is given by:

$$\omega_i^v = \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix} \tag{3.13}$$

The inertia tensor in the vehicle frame is given by:

$$I^v = R_b^v(\phi) I^b R_b^{v\,T}(\phi) \tag{3.14}$$

where I^b is the inertia tensor in the body frame:

$$I^b = \begin{pmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{pmatrix} \tag{3.15}$$

This gives:

$$I^v = \begin{pmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy}\cos^2\phi + I_{zz}\sin^2\phi & (I_{yy} - I_{zz})\sin\phi\cos\phi \\ 0 & (I_{yy} - I_{zz})\sin\phi\cos\phi & I_{yy}\sin^2\phi + I_{zz}\cos^2\phi \end{pmatrix} \tag{3.16}$$

The spacial angular velocity ω_s is given by:

$$\omega_s = \begin{pmatrix} \dot{\phi} \\ 0 \\ \dot{\psi} \end{pmatrix} \tag{3.17}$$

The components of the torque vector which are of interest are the x and z components, given by:

$$\tau_x = F_{yT} h \cos\phi + mgh \sin\phi - C_\phi \phi - K_\phi \dot{\phi} \tag{3.18}$$
$$\tau_z = M_T - F_{xT} h \sin\phi \tag{3.19}$$

Using (3.12) the equations of angular motion are found to be:

$$\ddot{\phi} = \frac{F_{yT} h \cos\phi + mgh \sin\phi - C_\phi \phi - K_\phi \dot{\phi} + \dot{\psi}^2 (I_{yy} - I_{zz}) \sin\phi \cos\phi}{I_{xx}} \tag{3.20}$$

$$\ddot{\psi} = \frac{M_T - F_{xT} h \sin\phi - 2(I_{yy} - I_{zz}) \sin\phi \cos\phi \dot{\phi} \dot{\psi}}{I_{yy} \sin^2\phi + I_{zz} \cos^2\phi} \tag{3.21}$$

By assuming that the roll angle ϕ is small and making appropriate approximations, simplified expressions can be obtained:

$$\ddot{\phi} \approx \frac{F_{yT} h + mgh\phi - C_\phi \phi - K_\phi \dot{\phi} + \dot{\psi}^2 (I_{yy} - I_{zz}) \phi}{I_{xx}} \tag{3.22}$$

$$\ddot{\psi} \approx \frac{M_T - F_{xT} h\phi - 2(I_{yy} - I_{zz}) \phi \dot{\phi} \dot{\psi}}{I_{yy} \phi^2 + I_{zz}} \tag{3.23}$$

Translational Motion The equations for translational motion can be obtained from (3.6). The equations are:

$$\dot{v}_x = \frac{F_{xT}}{m} + v_y \dot{\psi} - h \sin\phi \ddot{\psi} - 2h \cos\phi \dot{\phi} \dot{\psi} \tag{3.24}$$

$$\dot{v}_y = \frac{F_{yT}}{m} - v_x \dot{\psi} - h \sin\phi \cos\phi \dot{\psi}^2 + h\ddot{\phi} \tag{3.25}$$

Substituting the expressions for the angular accelerations from (3.20) and

(3.21) into (3.24) and (3.25) gives:

$$\begin{aligned}\dot{v}_x =& \frac{F_{xT}}{m} + v_y\dot{\psi} - 2h\cos\phi\dot{\phi}\dot{\psi} \\ & - h\sin\phi\left(\frac{M_T - F_{xT}h\sin\phi - 2(I_{yy} - I_{zz})\sin\phi\cos\phi\dot{\phi}\dot{\psi}}{I_{yy}\sin^2\phi + I_{zz}\cos^2\phi}\right) \qquad (3.26)\\ \dot{v}_y =& \frac{F_{yT}}{m} - v_x\dot{\psi} - h\sin\phi\cos\phi\dot{\psi}^2 + \frac{h}{I_{xx}}(F_{yT}h\cos\phi + mgh\sin\phi \\ & - C_\phi\phi - K_\phi\dot{\phi} + \dot{\psi}^2(I_{yy} - I_{zz})\sin\phi\cos\phi) \qquad (3.27)\end{aligned}$$

The translational and angular equations of motion are thus given by (3.26), (3.27), (3.20) and (3.21).

Euler-Lagrange Modeling

In this section, the Euler-Lagrange method will be used to obtain the equations of motion for the four degree of freedom vehicle model, including a constant but nonzero pitch angle θ_r. The derivation follows that in [Pacejka, 2002]. The Euler-Lagrange equations state that:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q_i^x \qquad (3.28)$$

where q_i are generalized coordinates and Q_i^x are corresponding external forces. The Lagrangian L is:

$$L = T - U$$

where T is the kinetic energy of the system and U is the potential energy. When U is independent of the generalized velocities, Lagrange's equations become:

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{q}_i} - \frac{\partial T}{\partial q_i} + \frac{\partial U}{\partial q_i} = Q_i^x \qquad (3.29)$$

For a vehicle model with two translational and two rotational degrees of freedom, one choice of generalized coordinates q_i for the system is the Cartesian coordinates X and Y of the ground-fixed (inertial) coordinate system, the yaw angle ψ between the X axis and the (vehicle-fixed) x axis, and the roll angle ϕ, defined around the x axis. The configuration of the vehicle is then described by:

$$q = \begin{pmatrix} X \\ Y \\ \psi \\ \phi \end{pmatrix}$$

However, the first three generalized coordinates in this representation are of little interest, since they represent the position of the vehicle within an arbitrary earth-fixed system. A set of coordinates in a vehicle-fixed frame would be of more use, since these coordinates are more likely to be available as measurements or estimates in an actual vehicle.

Consider the coordinate change $q_i^* = q_i^*(q_i)$, where:

$$q^* = \begin{pmatrix} v_x \\ v_y \\ \dot{\psi} \\ \phi \end{pmatrix}$$

This coordinate change is obtained from:

$$\begin{aligned} v_x &= \dot{X}\cos\psi + \dot{Y}\sin\psi \\ v_y &= -\dot{X}\sin\psi + \dot{Y}\cos\psi \end{aligned}$$

The new configuration coordinates q^* are now the longitudinal and lateral velocities v_x and v_y, as well as the yaw rate $\dot{\psi}$ and roll angle ϕ. These coordinates are independent of the choice of earth-fixed frame, and are therefore of more practical use.

With this coordinate transformation, Lagrange's equations (3.29) must also be transformed. In particular, the original generalized velocities can be written as:

$$\begin{aligned} \dot{X} &= \dot{X}(v_x, v_y, \psi) \\ \dot{Y} &= \dot{Y}(v_x, v_y, \psi) \end{aligned}$$

The partial derivatives of the kinetic energy T are then:

$$\begin{aligned} \frac{\partial T}{\partial \dot{X}} &= \frac{\partial T}{\partial v_x}\frac{\partial v_x}{\partial \dot{X}} + \frac{\partial T}{\partial v_y}\frac{\partial v_y}{\partial \dot{X}} + \frac{\partial T}{\partial \psi}\frac{\partial \psi}{\partial \dot{X}} = \frac{\partial T}{\partial v_x}\cos\psi - \frac{\partial T}{\partial v_y}\sin\psi \\ \frac{\partial T}{\partial \dot{Y}} &= \frac{\partial T}{\partial v_x}\frac{\partial v_x}{\partial \dot{Y}} + \frac{\partial T}{\partial v_y}\frac{\partial v_y}{\partial \dot{Y}} + \frac{\partial T}{\partial \psi}\frac{\partial \psi}{\partial \dot{Y}} = \frac{\partial T}{\partial v_x}\sin\psi + \frac{\partial T}{\partial v_y}\cos\psi \\ \frac{\partial T}{\partial \psi} &= \frac{\partial T}{\partial v_x}\frac{\partial v_x}{\partial \psi} + dT v_y \frac{\partial v}{\partial \psi} = \frac{\partial T}{\partial v_x} v_y - \frac{\partial T}{\partial v_y} v_x \end{aligned}$$

The resulting equations of motion are found to be:

$$\frac{d}{dt}\frac{\partial T}{\partial v_x} - \dot{\psi}\frac{\partial T}{\partial v_y} = Q_{v_x} \tag{3.30a}$$

$$\frac{d}{dt}\frac{\partial T}{\partial v_y} + \dot{\psi}\frac{\partial T}{\partial v_x} = Q_{v_y} \tag{3.30b}$$

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{\psi}} - v_y\frac{\partial T}{\partial v_x} + v_x\frac{\partial U}{\partial v_y} = Q_{\psi} \tag{3.30c}$$

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{\phi}} - \frac{\partial T}{\partial \phi} + \frac{\partial U}{\partial \phi} = Q_{\phi} \tag{3.30d}$$

The generalized forces Q_i can be obtained from the expression for virtual work:

$$\delta W = \sum_{i=1}^{4} Q_i \delta q_i$$

Here, δq_i are infinitesimal displacements in the v_x, v_y, ϕ and ψ directions. The virtual work is given by:

$$\delta W = \sum F_x \delta x + \sum F_y \delta y + \sum M_z \delta \psi + \sum M_\phi \delta \phi$$

The generalized forces are given by (3.10):

$$\sum F_x = F_{xT} \tag{3.31a}$$

$$\sum F_y = F_{yT} \tag{3.31b}$$

$$\sum M_z = M_T \tag{3.31c}$$

$$\sum M_\phi = -K_\phi \dot{\phi} \tag{3.31d}$$

In the last equation K_ϕ is the total damping coefficient of the suspension.

The kinetic and potential energies must now be derived. The kinetic energy is composed of two parts, the translational part and the rotational part. The velocities of the centre of gravity in the x and y directions can be found from (3.5) and are given by:

$$v_x^{CG} = v_x + h \sin\phi\dot{\psi}$$
$$v_y^{CG} = v_y - h\dot{\phi}$$

where h is the distance of the centre of gravity from the roll axis. Here, the angle θ_r between the roll axis and the x axis is neglected. The translational

kinetic energy is therefore given by (with θ_r neglected and ϕ assumed small):

$$T_{trans} = \frac{1}{2}m\big((v_x + h\phi\dot{\psi})^2 + (v_y - h\dot{\phi})^2\big)$$

The rotational kinetic energy of a system is given by [Spiegel, 1967]:

$$T_{rot} = \frac{1}{2}\Big(I_{xx}\omega_x^2 + I_{yy}\omega_y^2 + I_{zz}\omega_z^2 + 2I_{xy}\omega_x\omega_y + 2I_{xz}\omega_x\omega_z + 2I_{yz}\omega_y\omega_z\Big)$$

where I_{ii} are the moments of inertia around the axis i and the quantities I_{ij} are products of inertia.

The rotational kinetic energy must now be determined. A troublesome issue arising in rigid body rotations is that the results depend on the order in which the rotations are performed. According to [Pacejka, 2002], the rotational kinetic energy of the system (with θ_r accounted for but assumed small, and ϕ also assumed small) is given by:

$$T_{rot} = \frac{1}{2}\Big(I_{xx}\dot{\phi}^2 + I_{yy}(\phi\dot{\psi})^2 + I_{zz}(\dot{\psi}^2 - \phi^2\dot{\psi}^2 + 2\theta_r\dot{\psi}\dot{\phi}) - 2I_{xz}\dot{\psi}\dot{\phi}\Big)$$

The total kinetic energy is obtained by summing the two parts:

$$\begin{aligned} T = {} & \frac{1}{2}m\big((v_x + h\phi\dot{\psi})^2 + (v_y - h\dot{\phi})^2\big) + \\ & + \frac{1}{2}\Big(I_{xx}\dot{\phi}^2 + I_{yy}(\phi\dot{\psi})^2 + I_{zz}(\dot{\psi}^2 - \phi^2\dot{\psi}^2 + 2\theta_r\dot{\psi}\dot{\phi}) - 2I_{xz}\dot{\psi}\dot{\phi}\Big) \end{aligned} \tag{3.32}$$

The potential energy of the system is stored in the suspension springs and the height of the centre of gravity. It is given by:

$$U = \frac{1}{2}C_\phi\phi^2 - mg(h - h\cos\phi)$$

If a small angle approximation is used for the second term on the right hand side, the term will disappear. Instead, write:

$$\begin{aligned} mgh(1 - \cos\phi) &= 2mgh\sin^2\left(\frac{\phi}{2}\right) \\ &\approx \frac{1}{2}mgh\phi^2 \end{aligned}$$

The potential energy is then given by:

$$U = \frac{1}{2}C_\phi\phi^2 - \frac{1}{2}mgh\phi^2$$

The equations of motion may now be obtained by evaluating the Lagrangian equations. To evaluate (3.30a) the following partial derivatives must first be calculated:

$$\frac{\partial T}{\partial v_x} = m(v_x + h\phi\dot{\psi}) \qquad \frac{\partial T}{\partial v_y} = m(v_y - h\dot{\phi})$$

$$\frac{d}{dt}\frac{\partial T}{\partial v_x} = m\dot{v}_x + mh(\dot{\phi}\dot{\psi} + \phi\ddot{\psi})$$

The first equation of motion is then given by:

$$m\left[\dot{v}_x - \dot{\psi}v_y + h(2\dot{\phi}\dot{\psi} + \phi\ddot{\psi})\right] = F_{xT} \tag{3.33}$$

To derive the next equation the derivative:

$$\frac{d}{dt}\frac{\partial T}{\partial v_y} = m\dot{v}_y - mh\ddot{\phi}$$

must be calculated. The second equation of motion is then:

$$m\left[\dot{v}_y + \dot{\psi}v_x - h\ddot{\phi} + h\dot{\psi}^2\phi\right] = F_{yT} \tag{3.34}$$

For the third equation the derivatives:

$$\frac{\partial T}{\partial \dot{\psi}} = mhv_x\phi + mh^2\phi^2\dot{\psi} + I_{yy}\phi^2\dot{\psi} + I_{zz}(\dot{\psi} - \dot{\psi}\phi^2 + \theta_r\dot{\phi}) - I_{xz}\dot{\phi}$$

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{\psi}} = mh\dot{v}_x\phi + mhv_x\dot{\phi} + (mh^2 + I_{yy})(\ddot{\psi}\phi^2 + 2\dot{\psi}\phi\dot{\phi}) +$$
$$+ I_{zz}(\ddot{\psi} - \ddot{\psi}\phi^2 - 2\dot{\psi}\phi\dot{\phi} + \theta_r\ddot{\phi}) - I_{xz}\ddot{\phi}$$

are calculated.The third equation of motion is given by:

$$(I_{zz} + (mh^2 + I_{yy} - I_{zz})\phi^2)\ddot{\psi} + (I_{zz}\theta_r - I_{xz})\ddot{\phi} + mh\phi(\dot{v}_x - v_y\dot{\psi})$$
$$+ 2\phi\dot{\phi}\dot{\psi}(mh^2 + I_{yy} - I_{zz}) = M_T \tag{3.35}$$

For the fourth equation of motion the partial derivatives with respect to ϕ must be calculated:

$$\frac{\partial T}{\partial \dot{\phi}} = -mh(v_y - h\dot{\phi}) + I_{xx}\dot{\phi} + I_{zz}\theta_r\dot{\psi} - I_{xz}\dot{\psi}$$

$$\frac{\partial T}{\partial \phi} = mh\dot{\psi}(v_x + h\phi\dot{\psi}) + I_{yy}\dot{\psi}^2\phi - I_{zz}\dot{\psi}^2\phi$$

$$\frac{\partial U}{\partial \phi} = C_\phi\phi - mgh\phi$$

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{\phi}} = -mh(\dot{v}_y - h\ddot{\phi}) + I_{xx}\ddot{\phi} + I_{zz}\theta_r\ddot{\psi} - I_{xz}\ddot{\psi}$$

The final equation of motion is then given by:

$$(I_{xx} + mh^2)\ddot{\phi} - mh(\dot{v}_y + \dot{\psi}v_x) + (I_{zz}\theta_r - I_{xz})\ddot{\psi} - (mh^2 + I_{yy} - I_{zz})\phi\dot{\psi}^2 + (C_\phi - mgh)\phi + K_\phi\dot{\phi} = 0 \tag{3.36}$$

The model can be written on the form:

$$\begin{pmatrix} m & 0 & mh\phi & 0 & 0 \\ 0 & m & 0 & -mh & 0 \\ mh\phi & 0 & I_{zz} + (mh^2 + I_{yy} - I_{zz})\phi^2 & I_{zz}\theta_r - I_{xz} & 0 \\ 0 & -mh & I_{zz}\theta_r - I_{xz} & I_{xx} + mh^2 & K_\phi \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{v}_x \\ \dot{v}_y \\ \ddot{\psi} \\ \ddot{\phi} \\ \dot{\phi} \end{pmatrix}$$

$$= \begin{pmatrix} F_{xT} + m\dot{\psi}v_y - 2mh\dot{\phi}\dot{\psi} \\ F_{yT} - m\dot{\psi}v_x - mh\dot{\psi}^2\phi \\ M_T + mhv_y\dot{\psi}\phi - 2\phi\dot{\phi}\dot{\psi}(mh^2 + I_{yy} - I_{zz}) \\ mhv_x\dot{\psi} + (mh^2 + I_{yy} - I_{zz})\dot{\psi}^2\phi - (C_\phi - mgh)\phi \\ \dot{\phi} \end{pmatrix} \tag{3.37}$$

The parameters of the two-track models are summarized in Table 3.1.

Model Validation

In order to evaluate the models, the results are compared with simulation data from Daimler's CASCaDE vehicle simulation software. The total external forces and moments are taken from a (closed loop) simulation with CASCaDE and used as inputs to the derived models in open loop simulations performed using Dymola.

No adjustments were made to the vehicle parameters during the validation. Figures 3.7 to 3.11 show comparisons between the vehicle states from the CASCaDE simulation (denoted as 'measured') and the models obtained by Newton's and Lagrange's methods. It is clear that the longitudinal velocity as well as the rotational states agree very closely, while there is some discrepancy with the lateral velocity. Since the rotational dynamics are of primary interest, it can be concluded that the models are sufficiently accurate.

Since no parameter adjustment was performed, it is difficult to determine whether the observed discrepancies stem from model errors or parameters. The complex structure of the models renders systematic parameter optimization difficult.

Table 3.1 Parameters of the two–track model

Symbol	Description
m	Vehicle mass
h	Height of CG above roll axis
I_{xx}	Moment of inertia about x–axis
I_{yy}	Moment of inertia about y–axis
I_{zz}	Moment of inertia about z–axis
I_{xz}	Product of inertia for x and z axes
a	Distance from front axle to CG position (along x–axis)
b	Distance from rear axle to CG position (along x–axis)
l	Half track width
C_ϕ	Total roll stiffness
K_ϕ	Total roll damping
θ_r	Angle between roll axis and x–axis

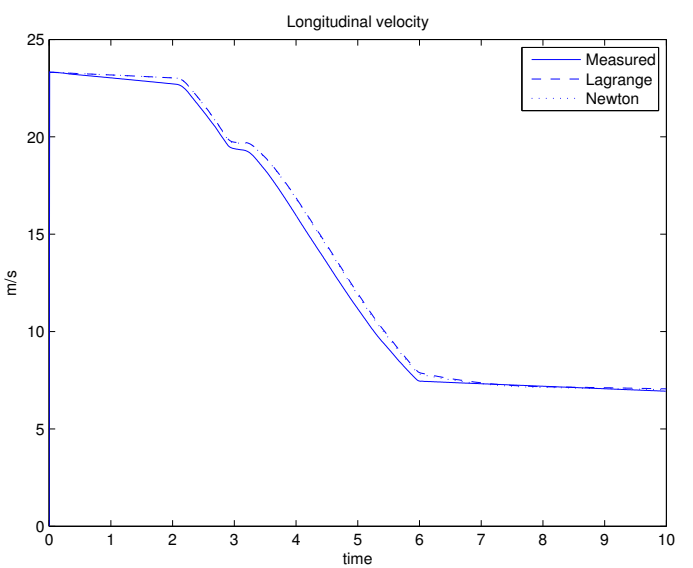

Figure 3.7 The longitudinal velocity v_x. The models agree closely with the simulation results.

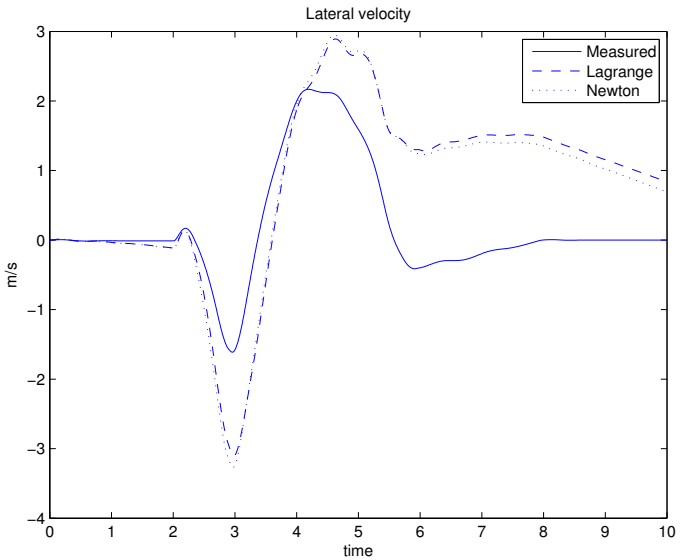

Figure 3.8 The lateral velocity v_y. The models agree with each other, but there is some discrepancy between them and the simulation results.

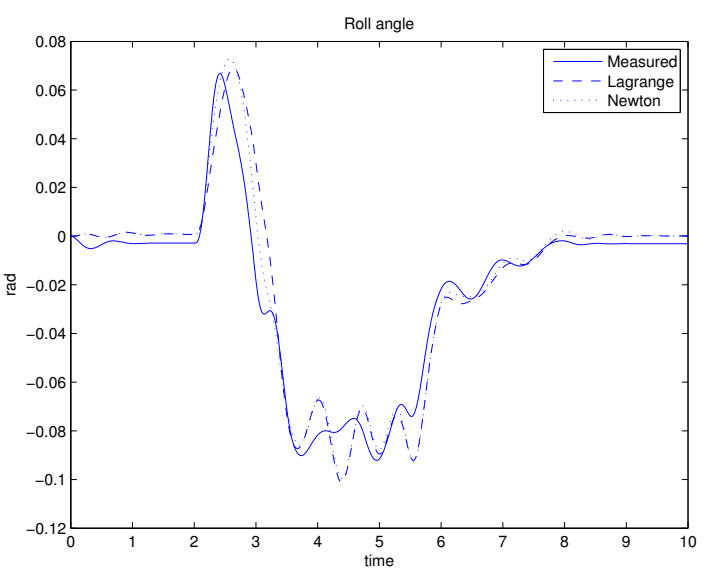

Figure 3.9 The roll angle ϕ. The agreement is relatively good.

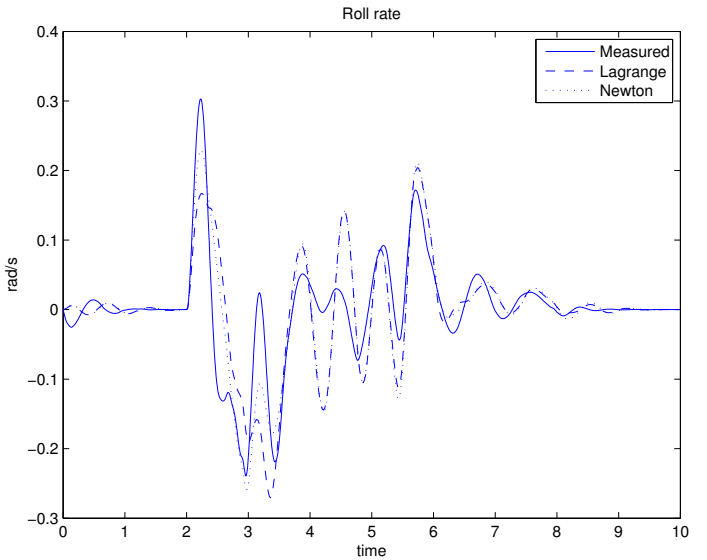

Figure 3.10 The roll rate $\dot{\phi}$. There are larger discrepancies between the models and the measured data, but the behaviours are similar.

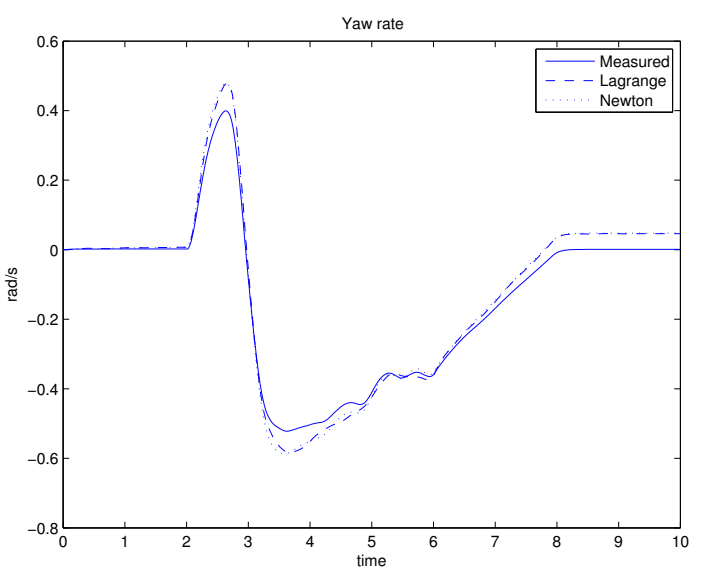

Figure 3.11 The yaw rate ψ. The models agree closely with the simulation results.

Application of Models

Reference Trajectory Generation The linear bicycle model described in Section 3.2 is not sufficient for control design since it lacks roll behavior. However, it may be used for the generation of reference trajectories, in particular the generation of a yaw rate reference, given the driver's steering command δ and the vehicle's longitudinal velocity.

Models for Control Design A linear model including roll dynamics could be used for roll control, and linear models are indeed extensively used in the literature. However, linear models use a number of assumptions and approximations which are unlikely to be valid during extreme maneuvering. These include:

- Constant longitudinal velocity
- Small steering angles
- Linear tire forces
- Simple approximations of tire slip values (α)

These approximations imply that although linear models may be useful for designing control systems intended for use under 'normal' driving conditions, they may be of limited use for the case of extreme maneuvering, where nonlinearities in tire characteristics and vehicle dynamics must be taken into account. In addition, the load transfer which occurs during extreme maneuvering cannot be modeled with a single-track model. For these reasons, the two-track nonlinear model given in (3.20), (3.21), (3.24) and (3.25) will be used as a basis for control design.

4

Rollover Analysis and Detection

In this chapter, analysis of vehicle rollover is performed. The aim of the analysis is to give insight into the mechanisms that cause rollover, and to determine how detection of an imminent rollover event might be performed. Methods of rollover detection are discussed, including a review of previous work on the subject.

4.1 Static Rollover Analysis

The underlying cause of untripped vehicle rollover accidents is the rotational motion occurring when a vehicle makes a turn. Figure 4.1 illustrates a vehicle performing a turn with a radius of curvature ρ. In order to maintain the curved trajectory, a force directed towards the centre of rotation must act upon the centre of gravity (CG) of the vehicle. Another way of considering this is to use the method of D'Alembert [Spiegel, 1967], in which accelerations are represented by pseudo-forces. D'Alembert's method allows dynamics problems to be viewed as statics problems. Figure 4.2 shows the pseudo-force ma_y acting on the centre of gravity of a vehicle performing a turn. Note that the pseudo-force acts in the opposite direction to the acceleration that it replaces, that is, it is directed radially outwards from the centre of rotation. The external forces acting on the vehicle act at the road-tire contact point, not the centre of gravity, meaning that a resulting moment acts on the vehicle. The magnitude of the resulting moment depends on the height of the centre of gravity above the road. A higher centre of gravity gives a larger moment. This moment is counteracted by a moment due to the reaction (normal) forces acting on the tires on the outside of the turn. This moment depends

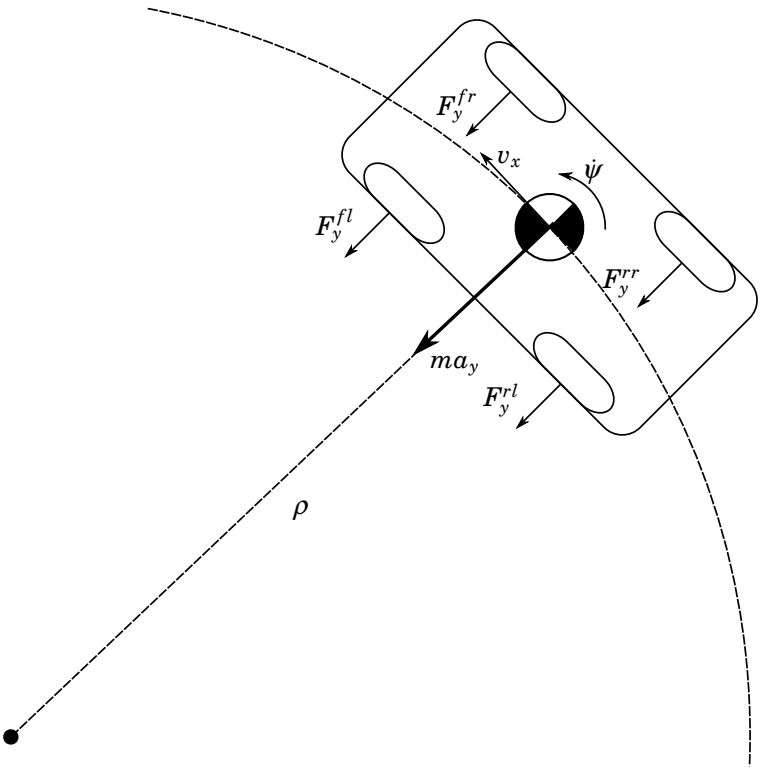

Figure 4.1 Illustration of a vehicle driving along a curved trajectory with radius of curvature ρ.

on the track width of the vehicle (the distance between inner and outer wheels). Clearly, if the moment due to the rotational motion of the vehicle exceeds the moment due to the the normal forces on the tires, then the vehicle will begin to roll.

A static condition for rollover can be derived from consideration of the resultant force vector acting on the center of gravity. If the line of action of the force lies outside the contact point of the outside wheels, then rollover will occur. Figure 4.3 illustrates the situation in the case of a vehicle without suspension. In this case, the condition for rollover to occur is:

$$ma_y h_T > mgl$$
$$a_y > \frac{gl}{h_T} \tag{4.1}$$

It is easy to see from (4.1) that the ratio of the height of the centre of

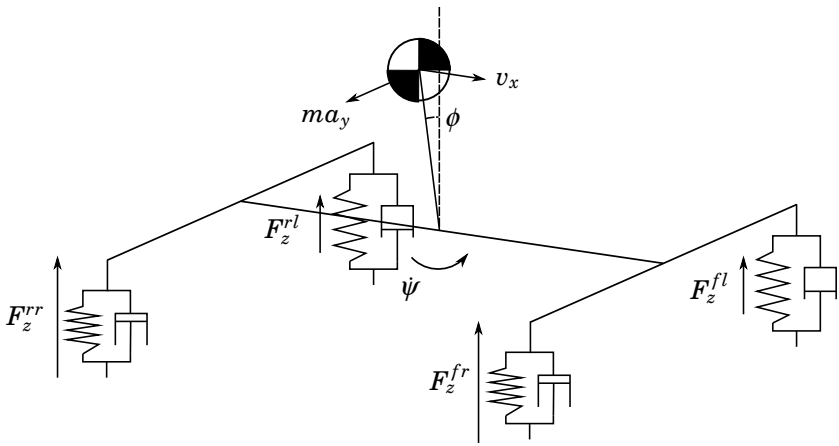

Figure 4.2 Illustration of the pseudo-force ma_y acting on the vehicle's centre of gravity.

gravity h_T to the half track width l determines the lateral acceleration necessary for rollover to occur. It is also worthy of note that the vehicle mass m does not appear in the condition. Only the geometry of the vehicle is important. Figure 4.4 illustrates the slightly more complicated case of a vehicle in which suspension kinematics are taken into account. In this case the condition for rollover to occur is given by:

$$a_y > \frac{g(l - h_T \sin\phi)}{h_T \cos\phi} \tag{4.2}$$

Static Stability

When the lateral acceleration threshold (4.1) obtained from analysis of Figure 4.3 is exceeded, roll motion of the vehicle ensues. Figure 4.5 illustrates the simple case of a vehicle without suspension after the onset of rollover. To gain insight into the nature of rollover accidents, it is interesting to perform some stability analysis for this simplified model. By resolving the weight mg and the pseudo-force ma_y into components in the vehicle-fixed y and z directions, the following dynamics are obtained:

$$m(a_y l \sin\phi + a_y h \cos\phi + gh \sin\phi - gl \cos\phi) = I\ddot{\phi} \tag{4.3}$$

It is clear from (4.3) that the resultant roll moment becomes larger as the roll angle ϕ increases. Figure 4.6 shows the variation of the overturning

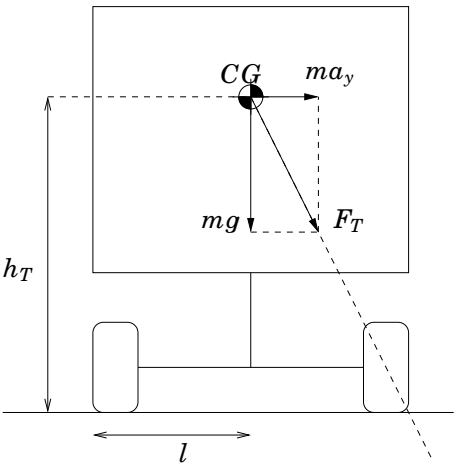

Figure 4.3 Illustration of the rollover limit for a vehicle with suspension elements neglected. The line of action of the resultant force acting on the CG passes through the contact point of the tires on the outside of the turn.

moment with roll angle for a number of different values of the lateral acceleration. The parameters used in the plots were $m = 3500$kg, $h = 1.4$m and $l = 0.9$m.

Although this analysis is based on a very simple model, ignoring the dynamic effects of the suspension, it brings to light some important points. Primarily, it can be seen that large sustained lateral accelerations are not necessary to cause rollover. Once rollover has begun, the magnitude of the lateral acceleration required to sustain it decreases. The effect of the lateral acceleration is replaced by the component of the weight acting along the vehicle's y axis.

4.2 Load Transfer

An important phenomenon in the study of rollover is load transfer. Load transfer refers to the shift in distribution of the vehicle's weight between the wheels. This has an important effect on the forces acting on the vehicle, due to the fact that the maximum achievable friction force for each tire depends on the normal force acting on the tire.

Lateral Load Transfer

Lateral load transfer is the change in normal force acting on the tires due to both the acceleration of the centre of gravity, and the shifting of posi-

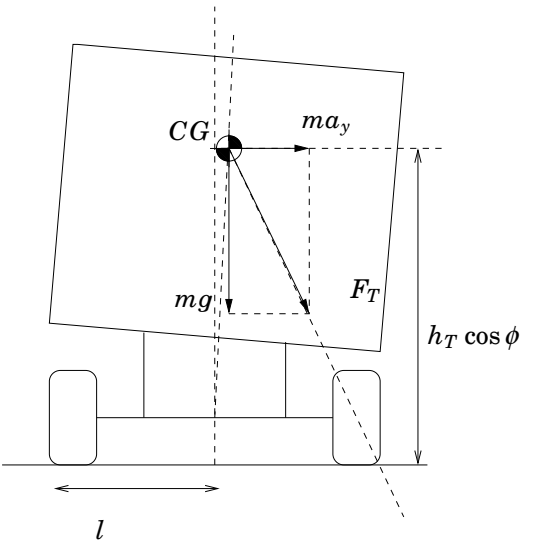

Figure 4.4 Illustration of the rollover limit for a vehicle with suspension kinematics taken into account. The changing position of the CG implies that the lateral acceleration required to produce rollover becomes smaller for larger roll angles.

tion of the CG in the y direction due to the movement of the suspension. Figure 4.7 illustrates lateral load transfer in the vertical plane. Figure 4.2 shows the effect of load transfer on the suspension.

Longitudinal Load Transfer

In addition to lateral load transfer, longitudinal load transfer can occur, due to acceleration in the longitudinal direction. Longitudinal load transfer occurs between the front and rear axles of the vehicle. The total resultant load transfer for each tire is the sum of the lateral and longitudinal load transfer. Figure 4.8 illustrates longitudinal load transfer. The simultaneous effect of lateral and longitudinal load transfer is illustrated in Figure 4.9.

Load Transfer Ratio

The load transfer ratio (LTR) can be defined as the difference between the normal forces on the right and left hand sides of the vehicle divided by their sum:

$$\mathrm{R} = \frac{F_{zR} - F_{zL}}{F_{zR} + F_{zL}} \tag{4.4}$$

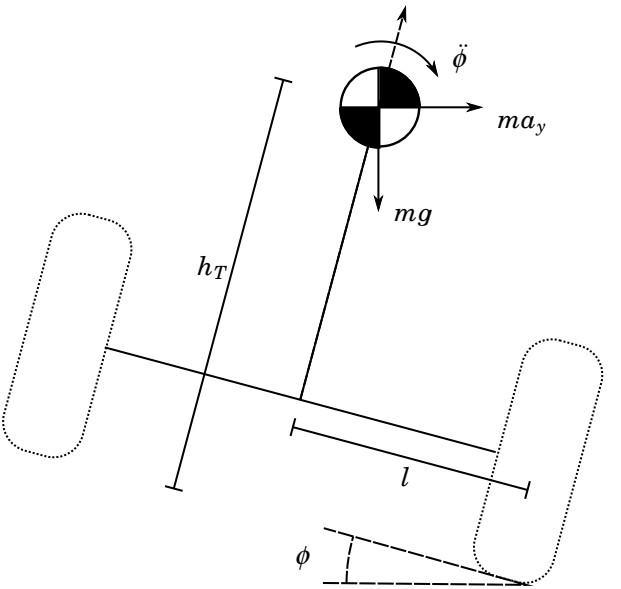

Figure 4.5 Illustration of vehicle after the onset of rollover.

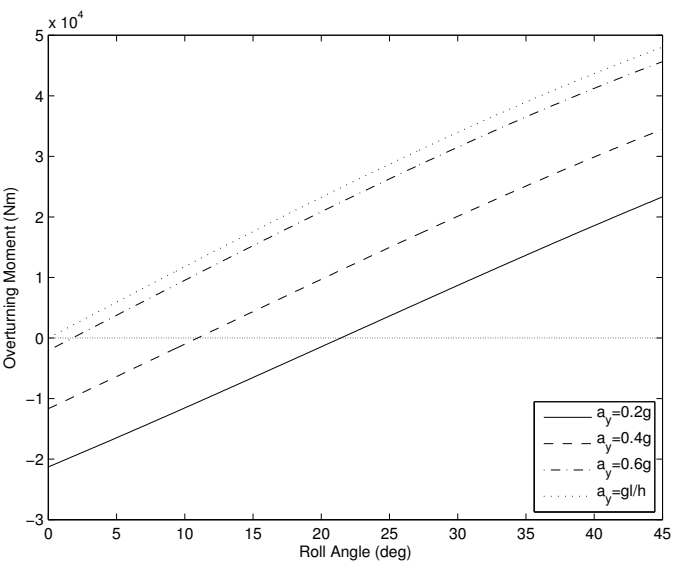

Figure 4.6 Plot showing the variation of the resulting moment acting on the vehicle with roll angle for different values of the lateral acceleration a_y. Positive values imply a net overturning moment, and negative values a restoring moment.

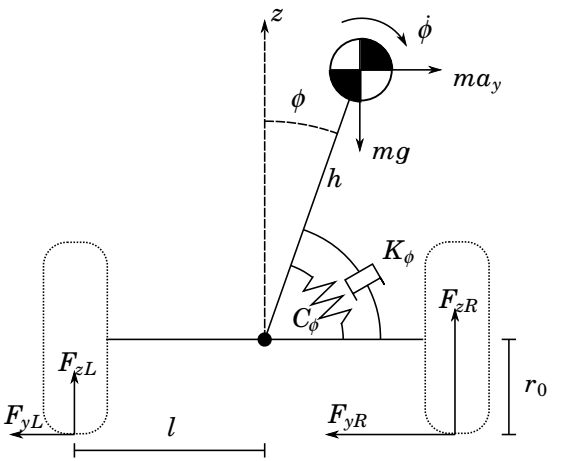

Figure 4.7 Lateral load transfer illustrated in the vertical plane.

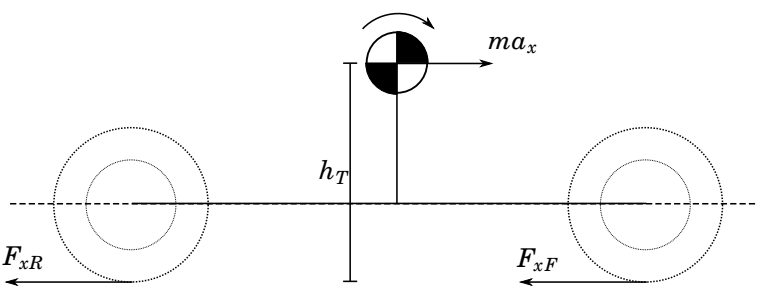

Figure 4.8 Illustration of longitudinal load transfer during braking.

where R denotes the LTR. Assuming no vertical motion, the sum of the normal forces equals the weight of the vehicle:

$$\mathrm{R} = \frac{F_{zR} - F_{zL}}{mg} \tag{4.5}$$

The LTR is defined for the entire vehicle. However, it is important to note that the longitudinal load transfer has the effect of decreasing the normal forces at the rear wheels during cornering (this is true even if braking is not performed, since the lateral forces have a component in the negative x direction). This effect implies that the rear wheel on the inside of the turn will be the first to lose contact with the road during a rollover.

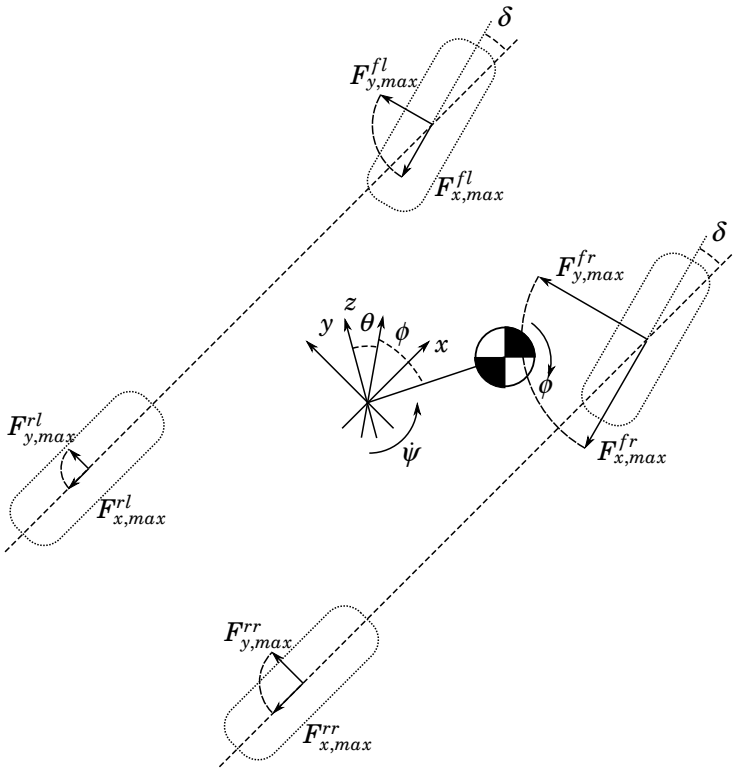

Figure 4.9 Illustration of the effects of load transfer on attainable tire forces during simultaneous cornering and braking.

4.3 Methods for Rollover Detection

Vehicle dynamics controllers do not operate continuously. They are typically activated only for certain situations. This switching requires some form of algorithm capable of detecting when the vehicle has entered, or is about to enter, a state in which the controller should be active. For VDC systems dealing with rollover prevention, the switching algorithm must be capable of detecting when a rollover is imminent. As was shown in the preceding section, the vehicle becomes increasingly unstable during rollover. It is therefore desirable that the detection algorithm has some form of predictive action, so that the controller may be activated as rapidly as possible.

Numerous methods for rollover detection have been suggested. For large trucks, several methods are studied in detail in [Dahlberg, 2001], in-

cluding load transfer, lateral acceleration, energy, and worst-case dynamic methods. In [Odenthal *et al.*, 1999], the load transfer ratio is used in a dynamic law to activate an emergency steering and braking controller. If a roll rate gyro is available, activation can be performed relatively easily. An alternative approach to rollover detection is taken in [Chen and Peng, 1999], where a vehicle model is simulated faster than real-time. If the simulated vehicle experiences rollover, an estimate of the time remaining before rollover occurs is available. This method is used in [Chen and Peng, 2000] and [Chen and Peng, 2001] to activate a rollover mitigation controller.

Roll Angle and Roll Rate Measurement

If the vehicle is equipped with sensors capable of measuring the roll angle ϕ and roll rate $\dot{\phi}$, rollover detection can be performed simply by analysing these measurements. The simplest approach would be to define a threshold value of the roll angle ϕ_{max} and to switch on the controller when $|\phi| > \phi_{max}$. In order to introduce predictive action to the algorithm, the roll rate measurement may be used. For instance, the controller may be switched on when $|\phi| > \phi_{max}$ and $\dot{\phi}\,\mathrm{sign}\,\phi > 0$. An obvious disadvantage using roll angle and roll rate measurements is that extra sensors are required.

Load Transfer-based Methods

Since load transfer occurs in connection with rollover events, the load transfer ratio is often used for rollover detection as well as control. The case of $R = \pm 1$, corresponding to the point at which one wheel begins to lose contact with the road surface, is often used as a 'critical situation' which should be avoided in order to prevent rollover [Odenthal *et al.*, 1999; Johansson and Gäfvert, 2004]. The load transfer ratio R is used as the switching variable in [Odenthal *et al.*, 1999], in which emergency braking is used to mitigate rollover. Two switching algorithms are proposed. One is based on a threshold value of R, denoted $\hat{R}$, and results in the control law:

$$F_{xT} = \begin{cases} 0 & \text{if } |R| \leq \hat{R} \\ -ma_{x,max} & \text{if } |R| > \hat{R} \end{cases} \tag{4.6}$$

where F_{xT} is the total braking force and $a_{x,max}$ is the maximum allowed deceleration. A dynamic switching strategy is also proposed, based on the derivative of the load transfer ratio. The idea is that the controller should be fully active when R is greater than $\hat{R}$ and is increasing ($\dot{R}\,\mathrm{sign}\,R > 0$) , but should be gradually switched off when R is decreasing ($\dot{R}\,\mathrm{sign}\,R < 0$).

This results in the control law:

$$F_{xT} = \begin{cases} 0 & \text{if } |R| \leq \hat{R} \\ -ma_{x,max} & \text{if } |R| > \hat{R} \text{ and } \dot{R}\,\text{sign}\,R > 0 \\ -\dfrac{|R| - \hat{R}}{R_{max} - \hat{R}} ma_{x,max} & \text{if } |R| > \hat{R} \text{ and } \dot{R}\,\text{sign}\,R < 0 \end{cases} \tag{4.7}$$

Energy-based Methods

The detection of a rollover event, and the subsequent activation of the controller, can be performed by considering the roll energy of the vehicle. The 'critical situation' is defined as when two wheels begin to lift from the ground, that is, the normal forces for these wheels become zero. In this situation, complete load transfer occurs, meaning that the total normal force on the side of the vehicle remaining in contact with the ground is equal to the vehicle's weight mg. It is assumed that the total lateral tyre force F_{yT} for the wheels retaining contact are at their maximum levels, so that $F_{yT} = \mu F_{zT} = \mu mg$.

The roll energy associated with the vehicle is composed of a potential energy part involving the energy stored in the suspension springs as well as the height of the center of gravity, and a kinetic energy part. The roll energy is given by:

$$E = \frac{1}{2} C_\phi \phi^2 - mgh(1 - \cos\phi) + \frac{1}{2}(I_{xx} + mh^2)\dot{\phi}^2 \tag{4.8}$$

A critical value of the roll energy, $E_{critical}$ can now be found, which represents the minimum possible roll energy in the critical situation of wheel lift-off.

For rollover to occur, the total moment around the roll axis due to the motion of the center of gravity must be greater than the moment due to the normal force from the tyres still in contact. The critical situation can then be defined by an inequality involving moments acting on the vehicle.

$$F_{zT} l < F_{yT} r_0 + C_\phi \phi + K_\phi \dot{\phi} \tag{4.9}$$

The critical roll energy $E_{critical}$ is found by minimizing (4.8) over ϕ and $\dot{\phi}$, subject to (4.9) (with the inequality replaced by an equality).

In [Johansson and Gäfvert, 2004], a normalized rollover warning measure, denoted ROW, is introduced, defined as:

$$ROW = \frac{E_{critical} - E}{E_{critical}}$$

The critical situation can thus be reached for $ROW \leq 0$. Switching is then accomplished by setting a threshold value of ROW (greater than zero),which can be found experimentally, and switching on the controller when $ROW \leq ROW_{threshold}$.

Lateral Acceleration-based Methods

As has been indicated in this chapter, the root cause of rollover can be seen as the D'Alembert pseudo-force ma_y acting at the centre of gravity and giving rise to a roll moment. Since the lateral acceleration a_y is usually measured in vehicles equipped with ESP systems, its use in rollover detection is attractive. The lateral acceleration limits (4.1) and (4.2) could be used as a starting point for obtaining a switching algorithm based on the lateral acceleration measurement. However, these limits are based on greatly simplified models. The limit in (4.2), obtained from a slightly more complex model than the limit in (4.1), gives a lower value for the maximum allowable lateral acceleration. Increasing the complexity of the model will in general result in increasingly smaller values. In addition, the situation becomes more complex when the suspension dynamics are accounted for. In this case, the state of the vehicle suspension system determines the rollover threshold. This is easily explained in physical terms; when the vehicle begins a turn at a time when the suspension is in steady state, energy is stored in the springs and more lateral acceleration is required for rollover. On the other hand, if energy is stored in the springs, and the vehicle is then subject to an acceleration in the opposite direction, the subsequent release of energy from the suspension will result in a lower value of lateral acceleration required for rollover. The lateral acceleration could be used in either a threshold-based switching strategy or a dynamic one. A threshold-based strategy could take the form:

$$F_{xT} = \begin{cases} 0 & \text{if } |a_y| \leq a_{y,threshold} \\ -ma_{x,max} & \text{if } |a_y| > a_{y,threshold} \end{cases} \tag{4.10}$$

which is similar to (4.6). A dynamic switching strategy could take a form similar to (4.7). However, care should be taken when implementing the time derivative of measured signals. The presence of noise means that filtering must be performed. To approach the problem more rigorously, the detection algorithm can be considered in the context of PD control.

PD Control A proportional-plus-derivative (PD) controller can be described by:

$$u(t) = K\left(e(t) + T_d \frac{de(t)}{dt}\right) \tag{4.11}$$

where $u(t)$ is the control signal and $e(t)$ is the tracking error. The derivative time constant T_d can be interpreted as the prediction horizon of the controller. In practice, (4.11) cannot be implemented directly because the derivative action greatly amplifies the noise. A common modification is to introduce a low-pass element into the transfer function for the derivative part. This may be seen as approximating the original transfer function:

$$sT_d \approx \frac{sT_d}{1 + sT_d/N} \tag{4.12}$$

where N is a parameter used to limit the high frequency gain.

A PD structure may be used to obtain a switching condition. Let $\hat{a}_y(t)$ denote the signal on which the switching is performed. Using the modified PD law, $\hat{a}_y(t)$ may be obtained using the filter:

$$\hat{A}_y(s) = K\left(A_y(s) + \frac{sT_d}{1 + sT_d/N}A_y(s)\right) \tag{4.13}$$

where $A_y(s)$ and $\hat{A}_y(s)$ are the Laplace transforms of $a_y(t)$ and $\hat{a}_y(t)$ respectively. This can be seen as a predicted value of a_y in T_d seconds in the future when $K = 1$. A switching law could then take the form:

$$F_{xT} = \begin{cases} 0 & \text{if } |\hat{a}_y(t)| \leq a_{y,threshold} \\ -ma_{x,max} & \text{if } |\hat{a}_y(t)| > a_{y,threshold} \end{cases} \tag{4.14}$$

Dynamic or predictive switching of this type may be useful for compensating for actuator dynamics. For example, braking systems typically have rate constraints which means that desired braking forces are not reached until after a certain rise time. The derivative time constant T_d may be used to compensate for these dynamics.

Summary

In this section a number of methods for rollover detection have been reviewed. In selecting an appropriate method it is necessary to take into account not only the potential effectiveness of the method but also practical considerations such as the availability of the information required by the algorithms. Algorithms based on energy and load transfer, for instance, require either measurements or good estimates of a number of variables, including roll angle and roll rate. In addition,a number of vehicle parameters are required for energy-based methods. Apart from these considerations, it is possible to group the methods into 'cause-based' and 'effect-based' methods. Energy considerations and load transfer can be

seen as effects of a rollover event, while lateral acceleration may be seen as the cause. For these reasons, the methods of detection based on lateral acceleration are chosen for use in the switching algorithm.

Such an approach does however depend on the quality of the lateral acceleration measurement. If the sensor signal is noisy or contains significant scale factor errors, this approach may not be ideal. Another important point is that the suspension dynamics are not taken into account. However, accounting for the dynamics introduces a host of difficult problems. Using a worst-case approach would almost certainly lead to an overly conservative switching strategy. This is highly undesirable since this implies that the controller would be activated in cases where it is in fact not required. One solution could be to use a switching strategy 'tuned' to the natural frequency of the suspension dynamics. This would introduce dependency on vehicle parameters, in particular the loading conditions (mass, moments of inertia and CG position), which may necessitate an adaptive solution.

In order to avoid overly conservative strategies, the particular solution of the dynamics is of primary interest. For this reason the method of [Chen and Peng, 1999] is interesting, despite its computational complexity and parameter dependence.

In summary, although the approach based on lateral acceleration has been chosen for use in this thesis, there is more work to be done on this topic. It is likely that a combination of some or all of the methods discussed in this chapter could be used in unison.

5

Control Design for Rollover Mitigation

As was seen in Chapters 2 and 3, the equations governing the vehicle dynamics and tire characteristics are complex. The plant can be considered as a nonlinear multiple-input multiple-output (MIMO) plant with a number of important constraints arising from the tire characteristics, in particular the friction ellipse.

The aim of the controller is to prevent vehicle rollover. However, it is not possible to decouple other aspects of vehicle motion. In particular, the yaw rate and sideslip must simultaneously be controlled.

There is currently a great deal of activity in the development of vehicle actuators. Modern braking systems allow individual assignment of braking force to each wheel, known as electronic brake force distribution (EBD). In addition to the brake system, active steering systems are being developed. These may take the form of steering assistance systems, which allow some modification of the driver's steering commands, and steer-by-wire systems, in which full control authority is available. Examples of the former can be found in certain production vehicles, while the latter are not yet in production. Active suspension systems allow online modification of the suspension dynamics. However, the hardware for such systems is complex and expensive.

In this work, a braking system with electronic brake force distribution is taken as the actuator. This is the most realistic choice of actuator with respect to implementation, since the controller can be used on any vehicle with a braking system supporting EBD. The use of active steering and active suspension components would allow more control freedom, but such systems are both expensive and currently not widespread in production vehicles. Nevertheless, the control architecture adopted to solve the problem is modular and rather general, so the introduction of additional actuators could be relatively easily accommodated.

5.1 Controller Architecture

The dynamic models derived in Chapter 3 describe the motion of the vehicle subject to certain generalized forces and moments. These forces and moments are (when aerodynamic effects are neglected) created by the forces generated by the tires. As was seen in Chapter 2, the characteristics of the tire are complex, particularly in the case of combined slip, where the longitudinal and lateral forces are interrelated. The longitudinal tire forces are effectively the control signals when EBD is used as the actuator, which introduces the input constraint that only braking forces are allowed. The nonlinear relation between longitudinal and lateral forces described by the friction ellipse further complicates the situation.

Similar model structures, in which the dynamics are expressed with generalized forces and moments as inputs, can be found in aerospace and marine applications. In the case of aircraft, the generalized forces and moments may be attained by combinations of actuator inputs, which depend on the configuration of the airframe.

Control Allocation

One approach that has received considerable attention recently, particularly in aerospace applications, is known as *control allocation*. The underlying idea is to split the controller into two parts. Typically, the two parts consist of a dynamic controller, which uses the generalized forces and moments as control variables, and a *control allocator*, which maps (either statically or dynamically) the generalized force and moment commands to actuator commands. The division of the controller into a dynamic part and a control allocator can be thought of along similar lines as the division of the plant into a dynamics part and an actuator. This is illustrated in Figure 5.1. The generalized force and moment commands produced by the dynamic controller will be referred to as *virtual controls* v and the actuator commands produced by the control allocator will be referred to as *actual controls* u.

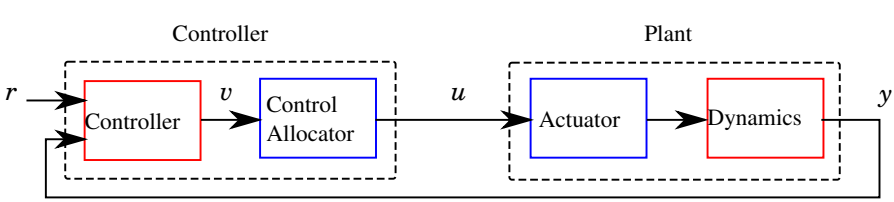

Figure 5.1 Control system structure with control allocation. The plant is split up into an actuator part and a dynamics part. The controller is similarly split into a controller which generates the virtual controls v, and a control allocator, which maps the virtual controls to actual controls u.

Motivation Although the mapping from virtual to actual controls (generalized forces and moments to actuator commands) is often static, there are a number of reasons for not using the actual control signals directly in the control design. Control allocation is typically used for over-actuated systems, in which there are more actuator inputs than virtual control inputs. Returning to the aircraft example, the virtual controls or generalized forces consist of three forces and three moments. However, depending on the aircraft configuration, there may be a large number of different actuators which may be used to obtain these resultant forces and moments. These could include elevators, ailerons, canard foreplanes, thrust vectoring and more. Similarly, for a road vehicle, resultant forces and moments may be achieved by using electronic brake force distribution, active steering, and active suspension. This is illustrated in Figure 5.2. Additional complications arise when constraints are imposed on the actuators. In this case the control allocator must choose a combination of actuator inputs which give the desired result while satisfying the constraints. Attempting to solve a multivariable design problem, possibly with nonlinear dynamics, while respecting constraints is generally difficult.

Advantages By dividing the design problem into two smaller problems, the control design task becomes more manageable. By dealing with constraints and nonlinearities in the actuator characteristics within the control allocator, standard control design methods may be more easily applied for the design of the dynamic controller. The modularity of the method in itself is an advantage, since once a controller has been designed, it may be used in conjunction with different control allocators depending on vehicle configuration, without necessitating a total redesign. Similarly, the control allocator provides a natural method for dealing with faults in actuators. Depending on the design of the allocator, a faulty actuator may simply be removed from the allocation problem. The control allocator will then work to fulfill the virtual control commands using the remaining actuators.

Previous Work Control allocation is an area of extensive research in aerospace applications. The application of control allocation to flight control design for a fighter aircraft is well documented in [Härkegård, 2003]. Other examples of aircraft applications include [Bordignon, 1996], [Davidson *et al.*, 2001], [Poonamallee *et al.*, 2004] and [Wise *et al.*, 1999]. Control allocation for road vehicles is discussed in [Andreasson, 2006], [Tøndel and Johansen, 2005] and [Plumlee *et al.*, 2004]. In [Fredriksson *et al.*, 14 17 Dec 2004], control allocation is used for vehicle handling control of hybrid vehicles. This application is particularly interesting since hybrid drivetrains, depending on their configuration, offer additional aspects of

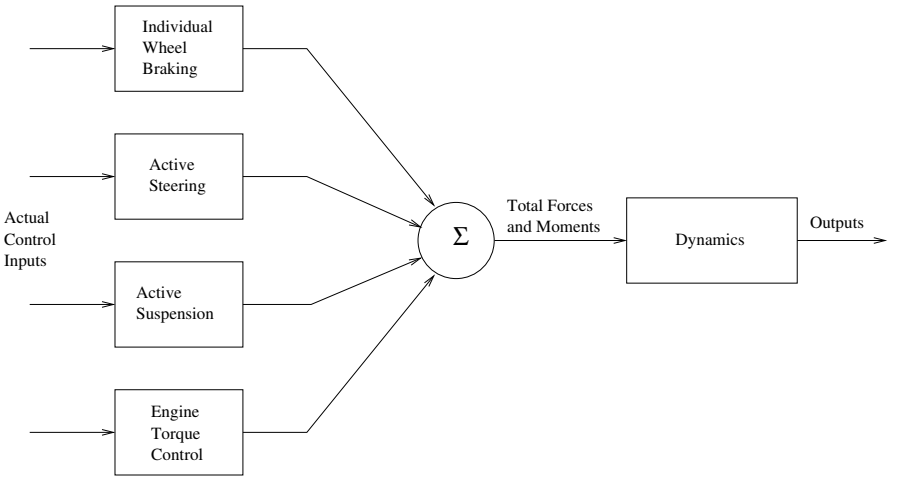

Figure 5.2 Block diagram illustrating the separation of actuators and dynamics for a road vehicle. One advantage of control allocation is modularity; the same dynamic controller can be used in conjunction with different control allocators depending on the types of actuator present in a given vehicle configuration.

control authority compared to vehicles with standard drivetrains. For example, the use of individual electric motors allows control of driving force, in addition to braking. This additional control freedom would certainly improve the performance of many VDC systems.

Marine applications, dealing with ships with multiple actuators, are also well suited to control allocation. Such an example is discussed in [Tjønnås and Johansen, 2008].

5.2 Control Objectives

The control task consists of two parts. Primarily, vehicle rollover must be prevented. Secondly, the yaw rate must be stabilized, and should track a reference. This secondary control objective is important, since the extreme maneuvering giving rise to a potential rollover may be necessary to avoid an obstacle, or remain on the road. Restriction of the vehicle sideslip angle β (the angle between the vehicle-fixed x-axis and the velocity vector) is also important, but this can be accomplished through appropriate yaw rate control [Tøndel and Johansen, 2005].

For the design, the generalized forces and moments, or virtual controls $v = (\, F_{xT} \quad F_{yT} \quad M_T \,)^T$ will be used. The task of obtaining the individual braking forces from given virtual controls is performed by the control

allocator, described in the following chapter.

Many previous approaches to rollover prevention are based on prevention of wheel lift-off [Johansson and Gäfvert, 2004]. While this is sufficient to prevent rollover, it may not be necessary, since roll stability may be maintained even after the loss of contact of a wheel. This approach may lead to conservative controllers which restrict maneuvering performance more than necessary. This is clearly undesirable in situations where extreme maneuvering is required to avoid an obstacle.

The strategy for roll control adopted here is to define a maximum allowable roll angle ϕ_{max} and design a controller to ensure that this limit is never exceeded. The choice of ϕ_{max} could come from an analysis of the dynamics of rollover such as that presented in the previous chapter, or could be chosen via experiments. Once a value of ϕ_{max} is decided, a corresponding limit on the total lateral force F_{yT} may be determined. From the friction ellipse, it can be seen that F_{yT} can be influenced by varying the total longitudinal (braking) force F_{xT}. The choice of F_{xT} constitutes the first part of the control design task. Yaw motion must then be controlled via the total moment M_T.

5.3 Roll Control

A bound on the roll angle may be translated into a bound on F_{yT} in the following way. From (3.22), the roll dynamics can be described by:

$$I_{xx}\ddot{\phi} + (C_\phi - mgh + (I_{zz} - I_{yy})\dot{\psi}^2)\phi + K_\phi\dot{\phi} = F_{yT}h$$

for small values of the roll angle ϕ. For reasonable values of the yaw rate $\dot{\psi}$ and the moments of inertia I_{zz} and I_{yy} the term $(I_{zz} - I_{yy})\dot{\psi}^2$ is much smaller than the term $C_\phi - mgh$ and may be neglected. The physical interpretation of this approximation is that the contribution to the roll angle of the motion of the vehicle about the z axis is much smaller than that of the rotation around the centre of curvature of the trajectory.

When this approximation is made, a scalar linear system is obtained. The transfer function from lateral force F_{yT} to roll angle ϕ is given by:

$$G_{roll}(s) = \frac{h}{I_{xx}s^2 + K_\phi s + C_\phi - mgh} \tag{5.1}$$

with corresponding impulse response $g_{roll}(t)$. The roll angle ϕ is given by:

$$\phi(t) = \int_0^\infty F_{yT}(t-\tau)g_{roll}(\tau)d\tau \tag{5.2}$$

$$\leq ||F_{yT}|| \underbrace{\int_0^\infty |g_{roll}(\tau)|d\tau}_{||G_{roll}||_{L_1}} \tag{5.3}$$

If the maximum allowable roll angle is given by ϕ_{max}, then the following inequality is obtained:

$$||F_{yT}|| \leq \frac{\phi_{max}}{||G_{roll}||_{L_1}} \tag{5.4}$$

This may be regarded as a dynamic lateral acceleration limit for a given maximum roll angle. Depending on the choice of ϕ_{max}, this relation will give a lower limit for the lateral acceleration (or lateral force) than the static limits obtained in the previous chapter, which ignore the suspension dynamics.

Recalling the idea of the friction ellipse, it is possible to limit F_{yT} by choosing F_{xT} sufficiently large. By considering a friction ellipse for the entire vehicle, and ignoring the effects of steering, the following approximate relation is obtained:

$$F_{xT} \approx F_{xT,max}\sqrt{1-\left(\frac{F_{yT}}{F_{yT,max}}\right)^2} \tag{5.5}$$

where $F_{xT,max}$ and $F_{yT,max}$ are the maximum attainable generalized forces. Substituting the condition (5.4) into (5.5) gives:

$$|F_{xT}| \geq F_{xT,max}\sqrt{1-\left(\frac{\phi_{max}}{F_{yT,max}||G_{roll}||_{L_1}}\right)^2} \tag{5.6}$$

Proportional Control with Threshold-based Switching

A proportional controller may now be dimensioned using (5.6) which ensures that the maximum allowable lateral force is never exceeded. It should be noted that some tuning may be required if the bound (5.4) is too conservative. The total lateral force itself cannot be used for feedback as it is not measurable, but the lateral acceleration is measured and may be used. Assuming the sensor is positioned at the origin of the vehicle coordinate system, the lateral force is given by $F_{yT} = ma_y$. The total

longitudinal force F_{xT} is regarded as the virtual control signal, and the control law is given by:

$$F_{xT} = -K_x m|a_y| \tag{5.7}$$

The gain K_x can be chosen so that the maximum allowable lateral force results in a control action satisfying (5.6). The control law may be combined with one of switching strategies presented in the previous chapter. As the control law is based on the lateral acceleration, it is natural to choose a switching method based on the same variable. The simplest algorithm is given by (4.10). A smoothing function is required to provide smooth transitions between on and off modes. Using this strategy, the control law becomes:

$$F_{xT} = \begin{cases} -K_x \Upsilon m|a_y| & |a_y| \geq a_{y,threshold} \\ 0 & |a_y| < a_{y,threshold} \end{cases} \tag{5.8}$$

where Υ is a suitable smoothing function.

Constant Control with Dynamic Switching

An alternative strategy, similar to that proposed in [Odenthal *et al.*, 1999], is to use a constant value for F_{xT}, which satisfies (5.6). This could for example be combined with the PD-type switching strategy (4.13). To prevent chattering, hysteresis may be used. The resulting control law could take the form:

$$F_{xT} = \begin{cases} -m|a_x^d| & \text{if controller on} \\ 0 & \text{if controller off} \end{cases} \tag{5.9}$$

where a_x^d is the desired longitudinal acceleration, possibly determined from a condition such as (5.6). The controller is switched on when $\hat{a}_y \geq \hat{a}_y^{on}$ and off when $\hat{a}_y \leq \hat{a}_y^{off}$, where $\hat{a}_y^{on} > \hat{a}_y^{off}$ are predefined thresholds.

5.4 Yaw Control

Attention may now be directed at controlling the yaw rate $\dot{\psi}$. From (3.20), it can be seen that the yaw rate can be influenced by both the total moment M_T and the total longitudinal force F_{xT}. Since the desired value of F_{xT} is given by the roll controller derived in the previous section, this can be assumed fixed, and M_T can be seen as the control variable for the yaw dynamics.

The purpose of the controller is to track a yaw rate reference signal, denoted $\dot{\psi}_{ref}$. The generation of this reference will be discussed later in the chapter. Since the yaw dynamics are nonlinear, it is appropriate to use a nonlinear control design methodology. A powerful tool often used in nonlinear control design is Lyapunov stability theory.

Lyapunov Theory

Lyapunov stability theory is a commonly used tool for analysis of nonlinear systems, as well as for the design of controllers. The theory allows the stability of particular solutions of a nonlinear system to be analyzed without solving the differential equations. In this section the main ideas of Lyapunov stability theory will be briefly presented. The reader is referred to [Slotine and Li, 1991] for an introduction to Lyapunov theory, and to [Khalil, 2002] for a more detailed treatment. The use of Lyapunov theory for control design is presented in [Kristić *et al.*, 1995].

Autonomous Systems Consider the system:

$$\dot{x} = f(x) \tag{5.10}$$

Assume without loss of generality that the system has an equilibrium point at $x = 0$.

DEFINITION 5.1
The equilibrium point $x = 0$ of (5.10) is:

- stable, if for each $\epsilon > 0$ there exists $\delta = \delta(\epsilon) > 0$ such that:

$$||x(0)|| < \delta \implies ||x(t)|| < \epsilon, \quad \forall t \geq 0$$

- unstable if not stable
- asymptotically stable if it is stable and δ can be chosen such that:

$$||x(0)|| < \delta \implies \lim_{t \to \infty} x(t) = 0$$

□

Lyapunov's method relies on the analysis of the properties of a scalar function $V(x)$, known as a Lyapunov function, to determine the properties of the solution of the system. This is commonly referred to as Lyapunov's direct method, or second method, and is summarised by the following theorem.

THEOREM 5.1
Let $x = 0$ be an equilibrium point of (5.10). Let $V : R^n \to R$ be a continuously differentiable function such that:

- $V(0) = 0$
- $V(x) > 0, \quad \forall x \neq 0$
- $||x(0)|| \to \infty \implies V(x) \to \infty$
- $\dot{V}(x) < 0, \quad \forall x \neq 0$

then $x = 0$ is globally asymptotically stable.
Proof:
For a proof see [Khalil, 2002, ch. 4]. □

Theorem 5.1 applies to the case of autonomous systems. For the case of nonautonomous systems, this cannot be used directly, but corresponding theorems exist.

Nonautonomous Systems Consider the nonautonomous system:

$$\dot{x} = f(t, x) \tag{5.11}$$

The system has an equilibrium point at $x = 0$ at time $t = 0$ if:

$$f(t, 0) = 0, \quad \forall t \geq 0$$

DEFINITION 5.2
The equilibrium point $x = 0$ of (5.11) is:

- stable, if for each $\epsilon > 0$ there exists $\delta = \delta(\epsilon, t_0) > 0$ such that:

$$||x(t_0)|| < \delta \implies ||x(t)|| < \epsilon, \quad \forall t \geq t_0 \geq 0 \tag{5.12}$$

- uniformly stable if, for each $\epsilon > 0$ there exists $\delta = \delta(\epsilon) > 0$ independent of t_0 such that (5.12) is satisfied
- unstable if not stable
- asymptotically stable if it is stable and there exists a positive constant $c = c(t_0)$ such that $x(t) \to 0$ as $t \to \infty$ for all $||x(t_0)|| < c$
- uniformly asymptotically stable if it is uniformly stable and there exists a positive constant c, independent of t_0, such that for all $||x(t_0)|| < c$, $x(t) \to 0$ as $t \to \infty$ uniformly in t_0; that is, for each $\eta > 0$, there exists a $T = T(\eta) > 0$ such that:

$$||x(t)|| < \eta, \quad \forall t \geq t_0 + T(\eta), \ \forall ||x(t_0)|| < c$$

□

THEOREM 5.2
Let $x = 0$ be an equilibrium point for (5.11) and $D \subset R^n$ be a domain containing $x = 0$. Let $V : [0, \infty) \times D \to R$ be a continuously differentiable function such that:

$$W_1(x) \leq V(t,x) \leq W_2(x) \tag{5.13}$$

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(t,x) \leq 0 \tag{5.14}$$

$\forall t \geq 0$ and $\forall x \in D$, where $W_1(x)$ and $W_2(x)$ are continuous positive definite functions on D. Then $x = 0$ is uniformly stable. If the inequality (5.14) is strengthened to:

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(t,x) \leq -W_3(x) \tag{5.15}$$

where $W_3(x)$ is a continuous positive definite function on D, then $x = 0$ is uniformly asymptotically stable.
Proof:
For a proof see [Khalil, 2002, ch. 4]. □

An important condition in Theorem 5.2 is that the Lyapunov function $V(t,x)$ must be *decrescent*, that is, it must be upper bounded by some function $W_2(x)$.

Lyapunov-based Design for Yaw Rate Control

The task of the yaw rate controller is to track a given yaw rate reference trajectory $\dot{\psi}_{ref}$. The aim of the control design is therefore to render $\dot{\psi} - \dot{\psi}_{ref} = 0$ a stable solution of the yaw dynamics. Since the yaw dynamics are given by a scalar system, a suitable Lyapunov function candidate is:

$$V = \frac{1}{2}(\dot{\psi} - \dot{\psi}_{ref})^2 \tag{5.16}$$

The yaw rate reference $\dot{\psi}_{ref} = \dot{\psi}_{ref}(t)$ is a function of time, so the Lyapunov function is time-dependent. The time derivative of V is given by:

$$\dot{V} = \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \dot{\psi}} \frac{d\dot{\psi}}{dt} \tag{5.17}$$

$$= -(\dot{\psi} - \dot{\psi}_{ref}) \frac{d\dot{\psi}_{ref}}{dt} + (\dot{\psi} - \dot{\psi}_{ref}) \frac{d\dot{\psi}}{dt} \tag{5.18}$$

Introducing the yaw dynamics from (3.20) gives:

$$\dot{V} = (\dot{\psi} - \dot{\psi}_{ref}) \left(\frac{M_T - F_{xT} h \sin\phi - 2\dot{\phi}\dot{\psi}(I_{yy} - I_{zz}) \sin\phi \cos\phi}{I_{yy} \sin^2\phi + I_{zz} \cos^2\phi} \right) - (\dot{\psi} - \dot{\psi}_{ref}) \ddot{\psi}_{ref} \tag{5.19}$$

The virtual controls F_{xT} and M_T may now be chosen such that the Lyapunov derivative becomes:

$$\dot{V} = -K_r(\dot{\psi} - \dot{\psi}_{ref})^2 \tag{5.20}$$

where K_r is a positive constant. By Theorem 5.2, the equilibrium $\dot{\psi} = \dot{\psi}_{ref}$ will then be uniformly asymptotically stable. Since F_{xT} is given by the roll control law (5.8), M_T can be found from:

$$\begin{aligned} M_T =& (-K_r(\dot{\psi} - \dot{\psi}_{ref}) + \ddot{\psi}_{ref})(I_{yy}\sin^2\phi + I_{zz}\cos^2\phi) + F_{xT}h\sin\phi \\ &+ 2\dot{\phi}\dot{\psi}(I_{yy} - I_{zz})\sin\phi\cos\phi \end{aligned} \tag{5.21}$$

Yaw Rate Reference Generation

A yaw rate reference $\dot{\psi}_{ref}$ is required for the yaw rate controller designed in the previous section. There are a number of ways of generating this. One approach is to use a simple vehicle model, such as a linearized bicycle model as in (3.9). This approach is well suited to control applications used during steady state driving conditions, since the model is based on a number of approximations valid for small steering and slip angles. It is however of limited use in more extreme maneuvering.

Another simple approach, described in [Kiencke and Nielsen, 2000] is to define a maximum allowable yaw rate $\dot{\psi}_{max}$, and to obtain the yaw rate reference as:

$$\dot{\psi}_{ref} = \begin{cases} \dot{\psi} & |\dot{\psi}| \leq |\dot{\psi}_{max}| \\ \pm\dot{\psi}_{max} & |\dot{\psi}| > |\dot{\psi}_{max}| \end{cases}$$

This approach is appropriate for controlling yaw rate alone, since the controller is inactive when the yaw rate is within the allowed interval. However, it is less attractive in the context of the proposed control strategy, where the controller is switched on in order to prevent rollover. Another approach to reference generation is to calculate a desired radius of curvature, based on the current vehicle speed, and then to calculate a yaw rate reference based on this. Ignoring the effects of vehicle sideslip, the minimum achievable radius of curvature ρ_{min} is given by:

$$\rho_{min} = \frac{u^2}{a_{y,max}}$$

where u is the longitudinal velocity and $a_{y,max}$ is the maximum achievable (or allowed) lateral acceleration. The value of ρ_{min} can be calculated each

time the controller is activated, and can then be used to generate a yaw rate reference:

$$\dot{\psi}_{ref} = \frac{u}{\rho_{min}} \tag{5.22}$$

This method of reference generation thus provides a yaw rate reference which corresponds to a minimum radius of curvature, and it therefore well suited to extreme maneuvers which may be caused by the need to avoid obstacles.

5.5 Summary

In this chapter, control design methodologies for obtaining the virtual control signals F_{xT} and M_T have been developed. These laws may be combined with the detection and switching methods presented in the previous chapter. In the following chapter, control allocation methods for conversion of these virtual controls into the actual control signals will be presented. Clearly, a large number of possible combinations of switching algorithms, control designs and control allocation methods exist. In the remainder of this thesis, the emphasis will be placed on those methods deemed most promising. In particular, the roll control strategy outlined in Section 5.3, along with the yaw rate control law (5.21) will be used. The resulting control law is outlined in Algorithm 5.1.

Algorithm 5.1: Control algorithm for generation of the virtual controls

```
Choose a_y^on, a_y^off, K_r, a_x^d ;
while driving do
    Filter a_y to obtain â_y using (4.13);
    if controller off & â_y ≥ â_y^on then
        controller on ;
        u_start := u;
        Calculate ρ_ref from u_start;
        Calculate ψ̇_ref from ρ_ref ;
        Calculate F_xT = −m a_x^d;
        Calculate M_T from (5.21);
        else if (controller on & â_y ≤ â_y^off) | (controller off & â_y < â_y^on)
        then
            controller off;
        else
            Calculate ψ̇_ref from ρ_ref ;
            Calculate F_xT = −m a_x^d;
            Calculate M_T from (5.21);
        end
    end
end
```

6

Control Allocation

In Chapter 5, control design was carried out using generalized forces and moments, or *virtual controls*. These virtual controls must be converted to actual control signals. Control allocation may be thought of as a systematic approach to performing this conversion.

Control allocation is a natural choice of approach for overactuated systems, particularly where constraints are present. It allows for an intuitive separation of the controller into two parts, corresponding to a separation of the plant into dynamics and actuator parts. Such a separation facilitates the application of standard control design methods, in the sense that constraints and actuator characteristics are handled by the control allocator.

The design of a control allocator involves two distinct stages. The first is the definition of the allocation problem. This is typically accomplished by establishing an optimization problem based on the minimization of the allocation error, with respect to relevant constraints. In the majority of cases this approach yields a static optimization problem which is to be solved at each sample. Because of this it is important to define optimization problems for which calculation of the solution is tractable in the time available.

The second aspect of the design is to determine how the optimization problem should be solved. Until recently, the focus here has been on obtaining approximate solutions to the optimization problem. Examples of such methods used within the context of aircraft control can be found in [Bordignon, 1996]. In recent years, exact solution of control allocation problems has become tractable, due to a combination of appropriate formulation of the optimization problem and choice of numerical method for the solution. Such methods are introduced in [Härkegård, 2003], again within the context of aircraft flight control.

6.1 Problem Formulation

The role of the control allocator is to obtain *actual controls* which will give rise to the desired *virtual controls*. In general, the relationship is $v(t) = g(u(t))$ where $v(t) \in \mathbb{R}^k$ are the virtual controls, $u(t) \in \mathbb{R}^m$ are the actual controls and $g : \mathbb{R}^m \to \mathbb{R}^k$ is the mapping from actual to virtual controls, where $m > k$. The majority of the literature deals with the linear case, where the actual and virtual controls are related by a *control effectiveness matrix* B:

$$v(t) = Bu(t) \tag{6.1}$$

In fact, it will be seen that any relation $v(t) = f(t, u, \theta)$, where θ is a parameter vector and f is linear in the actual controls u, can be handled using linear methods. The control allocation problem is an underdetermined, and often constrained problem. A common approach is to formulate an optimization problem in which the magnitude of the allocation error:

$$\epsilon = ||Bu(t) - v(t)||_p, \quad p = 1, 2, \ldots$$

is minimized, subject to actuator constraints and possibly additional costs on actuator use.

An important requirement imposed on the control allocation algorithm is that it must be implementable in a real-time environment. This is particularly important in automotive contexts, where sample times are typically of the order of 10ms. Algorithms with high levels of computational complexity are therefore not well suited to the application.

6.2 Convex Optimization

In order to use optimization for control allocation, it is natural to construct convex optimization problems. Such problems posses many attractive properties, and efficient solvers exist for a wide range of problem formulations. Additionally, a very large number of problems can be posed as convex optimization problems. In this section, a number of different problem formulations will be outlined. In the following section, methods for solving these types of problems will be reviewed.

The general form of a convex optimization problem is:

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \qquad i = 1, \ldots, m \end{array}$$

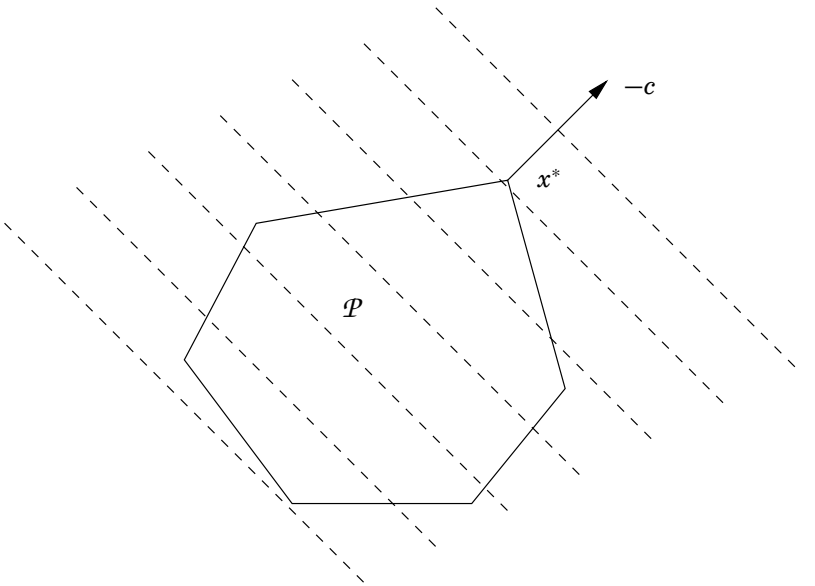

Figure 6.1 Interpretation of the solution of an LP problem, showing the active set $\mathcal{P}$ and the level curves of the objective function, which are hyperplanes orthogonal to c.

in which the objective function $f_0(x)$ and the constraints $f_i(x) \leq b_i$ are convex functions. The *feasible set* $\mathcal{P}$ of the optimization problem is the region in which the constraints are satisfied. The optimum x^* is the point in the feasible set where the objective function (also called cost function) is minimized. A number of sub–classes of convex problems exists, a number of which are outlined below.

Linear Programs

Linear Programs (LP) are convex problems in which both the objective and constraint functions are affine. They have the form:

$$\begin{aligned} \text{minimize} \quad & c^T x + d \\ \text{subject to} \quad & Gx \leq h \\ & Ax = b \end{aligned}$$

The feasible set of an LP is a polyhedron, and since the objective function is linear, the level curves are given by hyperplanes orthogonal to c. The feasible set and level curves of a general LP are illustrated in Figure 6.1.

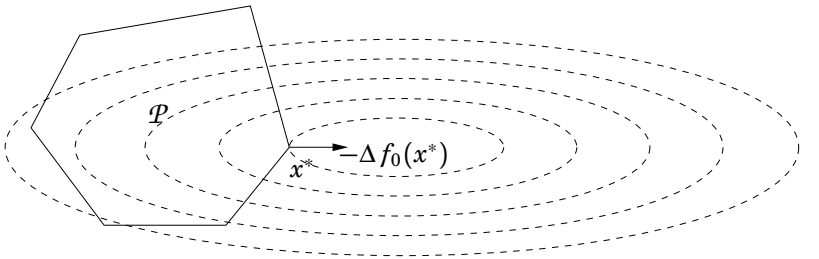

Figure 6.2 Interpretation of the solution of an QP problem, showing the active set $\mathcal{P}$ and the level curves of the objective function.

Quadratic Programs with Linear Constraints

In Quadratic Programs (QP), the objective function is convex quadratic. When linear constraints are present, the problem has the form:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}x^T P x + q^T x + r \\ \text{subject to} & Gx \leq h \\ & Ax = b \end{array}$$

In Figure 6.2 a graphical interpretation of a linearly constrained quadratic programming (LCQP) problem is shown.

Quadratic Programs with Quadratic Constraints

Another class of QPs are those in which the both the objective function and the constraints are convex quadratic:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} & \frac{1}{2}x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

This type of problem is referred to as a quadratically constrained quadratic programming (QCQP) problem.

6.3 Solving Convex Optimization Problems

Posing control allocation problems as convex optimization problems is attractive since there is a wide variety of efficient solvers for different types

of problems. Two classes of methods are of particular interest to the optimization problems considered in this chapter, *Interior Point* and *Active Set* methods.

Interior point methods can be used for convex problems that include inequality constraints, such as QCQP problems. A detailed presentation of interior point methods can be found in [Boyd and Vandenberghe, 2004].

Active set methods are based on finding the active constraints and solving simpler equality-constrained problems. These methods are particularly suitable for control allocation, for several reasons. Primarily, active set algorithms have the appealing property that a feasible solution is available after each iteration. For applications in a real-time environment this is particularly useful, since it means that if the algorithm must be interrupted, a feasible (albeit suboptimal) solution will always be available. In addition, active set methods become much more efficient when a good estimate of the active set is available. For control allocation purposes, a good estimate of the active set is usually given by the active set from the previous sample. In order to maximize efficiency, the structure of the problem must be utilized to select the most effective solver. Active set methods for control allocation are discussed in more detail in [Härkegård, 2003].

The numerical methods described above are used to solve the optimization problem online, at each sample. An alternative approach is to calculate the solutions offline, and select the appropriate one when the controller is active. An example of such a method is multi-parametric quadratic programming (MPQP) [Tøndel and Johansen, 2005]. MPQP is computationally efficient, but requires a considerable amount of memory to store the solution. A drawback of all methods which compute a solution offline is that it is difficult to handle changes in the control allocation problem. An example of such a change could be a fault in an actuator. With online computation, this situation is resolved trivially by simply removing the affected actuator from the allocation problem. With offline solutions, more complex approaches are required.

Another approach to solving control allocation problems is to use a dynamic approach, in which the allocation error converges to zero. Such an approach is taken in [Tjønnås and Johansen, 2008], and represents an elegant and computationally efficient method for control allocation. However, extension of the results to multivariable systems would be required for the application in this thesis.

6.4 Control Allocator Design

The control laws derived in the previous chapter use the generalized forces F_{xT}, F_{yT} and M_T as virtual controls. The aim of the control allocator is to obtain actual control signals which give rise to the desired virtual controls, while respecting certain constraints. The actual control signals in this case are taken to be the longitudinal tire forces. In reality the control commands are the brake pressures, but a simple relationship exists between these quantities. In this section, two methodologies for control allocator design are presented. The first is based on [Johansson and Gäfvert, 2004], and uses the friction ellipse, introduced in Chapter 2, in a direct way, yielding nonlinear constraints. The second approach uses certain assumptions and simplification allowing for the formulation of a simpler optimization problem.

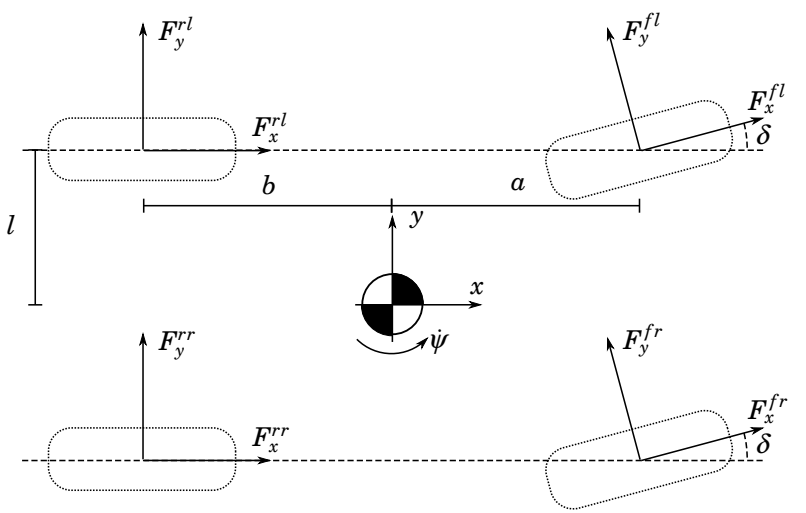

Figure 6.3 Planar chassis model, showing the horizontal components of the tire forces.

In Chapter 3, the relations between individual tire forces and resultant forces and moments were derived by considering Figure 6.3. These

relations are:

$$
\begin{aligned}
F_{xT} =& F_x^{rl} + F_x^{rr} + (F_x^{fl} + F_x^{fr})\cos\delta - (F_y^{fl} + F_y^{fr})\sin\delta \\
F_{yT} =& F_y^{rl} + F_y^{rr} + (F_y^{fl} + F_y^{fr})\cos\delta + (F_x^{fl} + F_x^{fr})\sin\delta \\
M_T =& (F_y^{fl} + F_y^{fr})a\cos\delta + (F_x^{fl} + F_x^{fr})a\sin\delta \\
& - (F_y^{rl} + F_y^{rr})b + (F_x^{rr} + F_x^{fr}\cos\delta + F_y^{fl}\sin\delta \\
& - F_x^{rl} - F_x^{fl}\cos\delta - F_y^{fr}\sin\delta)l
\end{aligned}
$$

Method 1: Formulation of a QCQP Allocation Problem

In this approach, both the longitudinal and lateral tire forces F_{xi} and F_{yi} are regarded as actual controls, with:

$$
u = \begin{pmatrix} F_x^{fl} & F_x^{fr} & F_x^{rl} & F_x^{rr} & F_y^{fl} & F_y^{fr} & F_y^{rl} & F_y^{rr} \end{pmatrix}^T
$$

Define the vector of virtual controls as:

$$
v = \begin{pmatrix} F_{xT} \\ F_{yT} \\ M_T \end{pmatrix} \tag{6.3}
$$

The virtual and actual controls are related by the control effectiveness matrix:

$$
Bu = v
$$

Control Effectiveness Matrix Derivation The control effectiveness matrix is obtained by considering the relations between individual tire forces (the actual controls) and generalized forces and moments (the virtual controls), which are outlined above. This gives:

$$
B^T = \begin{pmatrix}
\cos\delta & \sin\delta & (a\sin\delta + l\cos\delta) \\
\cos\delta & \sin\delta & (a\sin\delta - l\cos\delta) \\
1 & 0 & l \\
1 & 0 & -l \\
-\sin\delta & \cos\delta & (a\cos\delta - l\sin\delta) \\
-\sin\delta & \cos\delta & (a\cos\delta + l\sin\delta) \\
0 & 1 & -b \\
0 & 1 & -b
\end{pmatrix} \tag{6.4}
$$

It is clear that the matrix does in fact vary with δ, but since a new optimization problem is solved at each sample, B can be considered constant.

Constraints A number of constraints are present. The control signals are purely braking forces, giving rise to the constraint:

$$F_{xi} \leq 0 \tag{6.5}$$

The maximum allowable braking force is determined by the coefficient of friction μ between the tire and the road, as well as the vertical wheel load F_z, which can be expressed as:

$$F_{xi} \geq -|\mu F_{zi}| \tag{6.6}$$

The lateral force must act in the correct direction (the same direction as the maximum force $F_{yi,max}$). This can be expressed as:

$$F_{yi} F_{yi,max} \geq 0 \tag{6.7}$$

Finally, there are the constraints arising from the friction ellipse:

$$\left(\frac{F_y}{F_{y,max}}\right)^2 + \left(\frac{F_x}{F_{x,max}}\right)^2 \leq 1 \tag{6.8}$$

These constraints can be expressed as norm constraints, on the form:

$$||W_i u|| \leq 1 \tag{6.9}$$

where the matrices W_i have the form:

$$W_{FL} = \begin{pmatrix} \frac{1}{\mu F_{zFL}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{F_{yFL,max}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.10}$$

If a convex optimization problem is posed using the equality constraint (6.1), the presence of the other constraints may make the problem infeasible (no solution exists which satisfies all the constraints). To avoid this,

a slack variable γ is introduced. Replacing the equality constraint (6.1) with the inequality:

$$||Bu - v|| \leq \gamma \tag{6.11}$$

allows the optimization problem can be written as:

$$\begin{aligned} \text{minimize} \quad & \gamma \\ \text{subject to} \quad & ||Bu - v|| \leq \gamma \\ & ||W_i u|| \leq 1 \\ & F_{yi} F_{yi,max} \geq 0 \\ & F_{xi} \leq 0 \\ & F_{xi} \geq -|\mu F_{zi}| \end{aligned} \tag{6.12}$$

This is a quadratically-constrained quadratic program (QCQP). This particular case of a linear objective function together with quadratic inequality constraints is known as a second order cone programming (SOCP) problem.

Method 2: Formulation of a LCQP Allocation Problem

The second order cone problem posed in (6.12) can be solved efficiently, but it is unlikely that the solution can be found sufficiently quickly to allow real-time implementation. It is clear that some simplification of the problem may be advantageous. Regarding the computational complexity of the algorithm, a key issue is the type of constraints present. The constraints derived from the friction ellipse are quadratic, which increases the complexity of the problem and requires greater computation time for calculating the solution. In addition, the sideslip angles of the wheels α_i must be known in order to calculate the values of $F_{yi,max}$ used in the constraints in (6.12). This represents additional information which cannot be measured and be estimated. It is therefore proposed to make approximations which both simplify the constraints and reduce the amount of extra information required.

Since the controller will be operating exclusively in the limits of the vehicle's driving regime, it is reasonable to make approximations which are valid during these conditions. The first approximation is that the slip angles of all of the wheels are large enough such that the maximum lateral tire forces saturate, and are thus given by $F_{yi,max} = \mu F_{zi}$. This is attractive since the slip angles are no longer required in order to compute the maximum lateral forces. The resultant force on each wheel can now be seen as a function of the applied braking force and the normal force. However, the function is still nonlinear (the friction ellipse becomes a

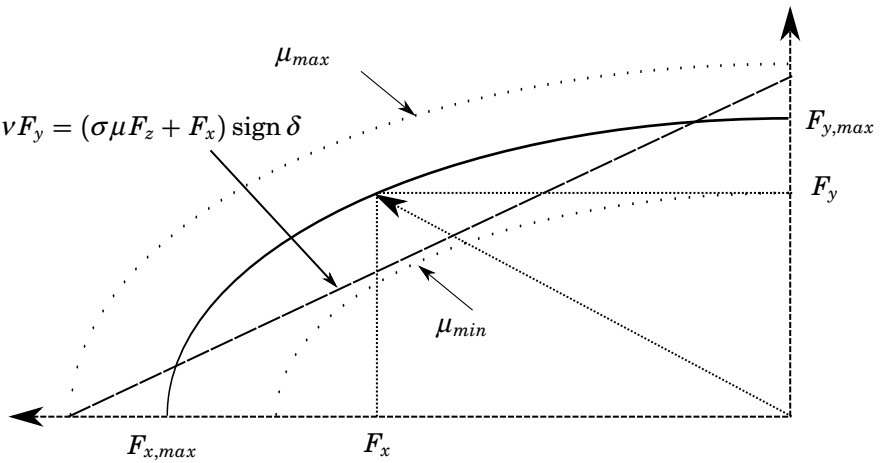

Figure 6.4 The friction ellipse with linear approximation, showing the uncertainty regions arising from the uncertainty of μ. Note that the linear approximation is only valid for $-\sigma\mu F_z \leq F_x \leq 0$. The approximation must therefore be used with constraints to obtain reasonable results.

circle when $F_{yi,max} = F_{xi,max}$), so a further approximation is suggested to simplify the constraints. The friction ellipse can be approximated in each quadrant by a linear function, as in Figure 6.4. This approximation can be justified by considering that there will be a large amount of uncertainty in the radius of the friction circle. In particular, μ is highly uncertain. The linear approximation can be thought of as lying within circles defined by upper and lower bounds of the radius μF_z. The approximation may be refined by introducing tuning parameters to alter the gradient and position of the linear approximations, giving a relationship on the form:

$$\nu F_y = (\sigma\mu F_z + F_x)\,\mathrm{sign}\,\delta \tag{6.13}$$

where ν and σ are the tuning parameters. The $\mathrm{sign}\,\delta$ factor is required to ensure that the resultant force acts in the correct direction. This approximation has the attractive property that the constraints are convex. In the formulation (6.12), it is assumed that the resultant force lies within the ellipse, rather than on the boundary, in order to obtain convex constraints. Using these simplifications, a new control allocation problem can now be formulated.

Replacing F_y with the linear approximation (6.13), and defining $\Delta = \mathrm{sign}\,\delta$, the relationships between braking forces and the generalized forces

become:

$$
\begin{aligned}
F_{xT} =& (\cos\delta - \frac{\Delta}{\nu}\sin\delta)(F_x^{fl} + F_x^{fr}) - \frac{\sigma\mu\Delta\sin\delta}{\nu}(F_z^{fl} + F_z^{fr}) \\
&+ F_x^{rl} + F_x^{rr} && (6.14)\\
F_{yT} =& (\frac{\Delta}{\nu}\cos\delta + \sin\delta)(F_x^{fl} + F_x^{fr}) + \sigma\mu\Delta(F_x^{rl} + F_x^{rr}) \\
&+ \frac{\sigma\mu\Delta\cos\delta}{\nu}(F_z^{fl} + F_z^{fr}) + \frac{\sigma\mu\Delta}{\nu}(F_z^{rl} + F_z^{rr}) && (6.15)\\
M_T =& (\frac{\Delta}{\nu}(a\cos\delta + l\sin\delta) + a\sin\delta - l\cos\delta)F_x^{fl} \\
&+ (\frac{\Delta}{\nu}(a\cos\delta - l\sin\delta) + a\sin\delta + l\cos\delta)F_x^{fr} \\
&+ \frac{\sigma\mu\Delta}{\nu}(a\cos\delta + l\sin\delta)F_z^{fl} + \frac{\sigma\mu\Delta}{\nu}(a\cos\delta - l\sin\delta)F_z^{fr} \\
&+ F_x^{rl}(-l - b\frac{\Delta}{\nu}) + F_x^{rr}(l - b\frac{\Delta}{\nu}) - \frac{b\sigma\mu\Delta}{\nu}(F_z^{rl} + F_z^{rr}) && (6.16)
\end{aligned}
$$

This may be written in vector form as:

$$
\begin{aligned}
F_{xT} =& \begin{pmatrix} (\cos\delta - \frac{\Delta}{\nu}\sin\delta) & (\cos\delta - \frac{\Delta}{\nu}\sin\delta) & 1 & 1 \end{pmatrix} u \\
&- \frac{\sigma\mu\Delta\sin\delta}{\nu}(F_z^{fl} + F_z^{fr}) && (6.17)\\
F_{yT} =& \begin{pmatrix} (\frac{\Delta}{\nu}\cos\delta + \sin\delta) & (\frac{\Delta}{\nu}\cos\delta + \sin\delta) & \frac{\Delta}{\nu} & \frac{\Delta}{\nu} \end{pmatrix} u \\
&+ \frac{\sigma\mu\Delta\cos\delta}{\nu}(F_z^{fl} + F_z^{fr}) + \frac{\sigma\mu\Delta}{\nu}(F_z^{rl} + F_z^{rr}) && (6.18)\\
M_T =& \begin{pmatrix} b_M^1 & b_M^2 & (-l - b\frac{\Delta}{\nu}) & (l - b\frac{\Delta}{\nu}) \end{pmatrix} u \\
&+ \frac{\sigma\mu\Delta}{\nu}\left(a\cos\delta(F_z^{fl} + F_z^{fr}) + l\sin\delta(F_z^{fl} - F_z^{fr}) - b(F_z^{rl} + F_z^{rr})\right) \\
& && (6.19)
\end{aligned}
$$

with $u = \begin{pmatrix} F_x^{fl} & F_x^{fr} & F_x^{rl} & F_x^{rr} \end{pmatrix}^T$, and where:

$$
\begin{aligned}
b_M^1 &= (\frac{\Delta}{\nu}(a\cos\delta + l\sin\delta) + a\sin\delta - l\cos\delta) \\
b_M^2 &= (\frac{\Delta}{\nu}(a\cos\delta - l\sin\delta) + a\sin\delta + l\cos\delta)
\end{aligned}
$$

The virtual controls v can now be expressed as:

$$v = \underbrace{\begin{pmatrix} (\cos\delta - \frac{\Delta}{v}\sin\delta) & (\cos\delta - \frac{\Delta}{v}\sin\delta) & 1 & 1 \\ (\frac{\Delta}{v}\cos\delta + \sin\delta) & (\frac{\Delta}{v}\cos\delta + \sin\delta) & \frac{\Delta}{v} & \frac{\Delta}{v} \\ b_M^1 & b_M^2 & (-l - b\frac{\Delta}{v}) & (l - b\frac{\Delta}{v}) \end{pmatrix}}_{B} u$$
$$+ \underbrace{\begin{pmatrix} -\frac{\sigma\mu\Delta\sin\delta}{v}(F_z^{fl} + F_z^{fr}) \\ \frac{\sigma\mu\Delta\cos\delta}{v}(F_z^{fl} + F_z^{fr}) + \frac{\sigma\mu\Delta}{v}(F_z^{rl} + F_z^{rr}) \\ \frac{\sigma\mu\Delta}{v}\left(a\cos\delta(F_z^{fl} + F_z^{fr}) + l\sin\delta(F_z^{fl} - F_z^{fr}) - b(F_z^{rl} + F_z^{rr})\right) \end{pmatrix}}_{d}$$

This gives the desired linear relationship between actual and virtual controls:

$$v(t) = B(\delta)u(t) + d \tag{6.20}$$

This can be transformed into the required form in (6.1) by defining new virtual controls $v'(t) = v(t) - d$. The vector d depends only on the normal forces F_{zi}, and is constant at each sample time. In addition, although the control effectiveness matrix depends on the steering angle, this may be assumed constant at each sample.

The constraints are now given by:

$$-|\sigma\mu F_{zi}| \leq F_{xi} \leq 0 \tag{6.21}$$

These constraints have the form of 'box constraints':

$$\underline{u} \leq u \leq \overline{u} \tag{6.22}$$

This type of constraint allows the formulation of an optimization problem with a special structure, which allows rapid computation.

Optimization Problem A linearly-constrained quadratic programming problem may be formulated. Such problems may take the form:

$$\begin{aligned} u &= \arg\min_{u\in\Omega} ||W_u(u - u_d)||_2 \\ \Omega &= \arg\min_{\underline{u}\leq u\leq\overline{u}} ||W_v(Bu - v')||_2 \end{aligned} \tag{6.23}$$

where W_u and W_u are diagonal weighting matrices, u_d is a desired actual control value, and $\underline{u}$ and $\overline{u}$ are constraints on the actual controls. This

type of problem is known as Sequential Least-Squares (SLS), since the solution is computed in two steps. First, the weighted allocation error $||W_v(Bu-v')||$ is minimized. If feasible solutions are found, then the 'best' solution is obtained by minimizing $||W_u(u-u_d)||$.

A faster algorithm can be obtained by approximating the SLS formulation as a Weighted Least-Squares (WLS) problem:

$$u = \arg\min_{\underline{u}\leq u\leq \overline{u}} \left(||W_u(u-u_d)||_2^2 + \gamma||W_v(Bu-v')||_2^2\right) \tag{6.24}$$

Here, the solution is calculated in a single step. The parameter γ is typically chosen to be very large in order to emphasize the importance of minimizing the allocation error.

Calculating the Solution Active set methods for the solution of the optimization problems (6.23) and (6.24) are presented in [Härkegård, 2003]. These methods will be briefly reviewed here.

Consider the least squares problem:

$$\min_u ||Au-b|| \tag{6.25a}$$

$$Bu = v \tag{6.25b}$$

$$\underbrace{\begin{pmatrix} I \\ -I \end{pmatrix}}_{C} u \geq \underbrace{\begin{pmatrix} \underline{u} \\ -\overline{u} \end{pmatrix}}_{U} \tag{6.25c}$$

The principal idea of active set methods is that in each step, some of the inequality constraints are taken to be equality constraints, while the remainder are ignored. Denote with $\mathcal{W}$ the *working set*, which contains all of the active constraints. Algorithm 6.1 outlines an active set method for finding the solution of the least squares problem above.

The weighted least squares problem may now be solved using Algorithm 6.2.

Parameter Selection Since only F_{xT} and M_T are used as virtual controls, F_{yT} may effectively be removed from the allocation problem by making the corresponding weight in the matrix W_v small relative to the other weights. The relative magnitudes of the remaining weights can be used to determine which virtual control is given priority, in cases where both cannot be satisfied simultaneously. For example, F_{xT} is the most critical virtual control for prevention of rollover, so the corresponding entry in W_v can be chosen to be larger than the others.

Algorithm 6.1: Active set algorithm

Let u^0 be a feasible starting point, satisfying (6.25c) ;
for $i = 0, 1, 2, \ldots$ **do**
Given suboptimal iterate u^i, find the optimal perturbation p, considering the inequality constraints in $\mathcal{W}$ as equality constraints and ignoring the remainder. This is done by solving:

$$\begin{aligned} &\min_p ||A(u^i + p) - b|| \\ &Bp = 0 \\ &p_i = 0, \quad i \in \mathcal{W} \end{aligned}$$

if $u^i + p$ *feasible* **then**
Set $u^{i+1} = u^i + p$;
Compute Lagrange multipliers as:

$$A^T(Au - b) = \begin{pmatrix} B^T & C_0^T \end{pmatrix} \begin{pmatrix} \mu \\ \lambda \end{pmatrix}$$

where C_0 consists of the rows of C corresponding to the constraints in the active set ;
if $\lambda \geq 0$ **then**
u^{i+1} is optimal solution;
Return $u = u^{i+1}$
else
Remove constraint corresponding to most negative λ from the working set $\mathcal{W}$;
end

The desired actual control vector u_d can be chosen in a number of ways. One possibility is to choose it as the actuator position that would be obtained in the absence of constraints:

$$u_d = B^\dagger v'$$

where $B^\dagger$ is the pseudo-inverse of B.

The weighting matrix W_u can be used to influence the distribution of the control actions among the actuators. It is most useful in cases where different types of actuators are present and preference should be given to a particular type. In this thesis the actuators are all of the same type so the choice of W_u is not critical.

Algorithm 6.2: Solution of the WLS control allocation problem (6.24)

Obtain the initial working set $\mathcal{W}$ from the active set from the previous sampling interval ;
Obtain the starting point u^0 from the optimal point from the previous sampling interval ;
Rewrite the cost function as:

$$||W_u(u-u_d)||_2^2 + \gamma||W_v(Bu-v')||_2^2 = \left\| \underbrace{\begin{pmatrix} \gamma^{\frac{1}{2}} W_v B \\ W_u \end{pmatrix}}_{A} u - \underbrace{\begin{pmatrix} \gamma^{\frac{1}{2}} W_v v \\ W_u u_d \end{pmatrix}}_{b} \right\|^2$$

Use Algorithm 6.1 to solve:

$$\begin{aligned} u =& \arg\min_u ||Au-b|| \\ & \underline{u} \leq u \leq \overline{u} \end{aligned}$$

Rate Constraints

Rate constraints in the actuators (in this case the braking system) may be taken into account in the control allocation problem by modifying the constraints at each sample time. Let the rate constraints be given by:

$$r_{min} \leq \dot{u}(t) \leq r_{max} \tag{6.26}$$

Approximating the derivative with the backward difference method:

$$\dot{u}(t) \approx \frac{u(t) - u(t-T_s)}{T_s}$$

where T_s is the sampling period allows the rate constraints to be rewritten as position constraints. The maximum allowable deviations of the positions from one sample time to another are given by:

$$\begin{aligned} \Delta_{min} &= r_{min} T_s \\ \Delta_{max} &= r_{max} T_s \end{aligned}$$

The new constraints are given by:

$$\underline{u}^*(t) = \max\{\underline{u}, u(t-Ts) + \Delta_{min}\} \tag{6.27}$$

$$\overline{u}^*(t) = \min\{\overline{u}, u(t-Ts) + \Delta_{max}\} \tag{6.28}$$

Table 6.1 Brake pressure rate constraints

Max. rising slew rate	200 bar/s
Max. falling slew rate	1000 bar/s

The rate constraints present in the control problem are the brake pressure rising and falling slew rates. These are summarized in Table 6.1. Since the braking forces are negative, the maximum rising slew rate corresponds to the minimum rate of change r_{min}, and the maximum falling slew rate corresponds to r_{max}. The brake pressure slew rates are converted into force rates of change using the appropriate scaling factors.

Hotstarting

The active set algorithm provides both the solution to the QP problem, as well as the current active set. The active set is the set of constraints which are currently active for the given solution. In control allocation, where the QP problem is solved at each sample time, it is very common that the active set obtained at a given sample time will also be the optimal active set at the next sample time. This implies that the previous active set and optimum point found in one sample time may be used as the starting point for the optimization problem in the following sample time, thereby reducing the number of iterations required to find the new solution. The use of the previous sample's solution and active set as the starting point for the next optimization is sometimes called 'hotstarting', and may significantly reduce the average computation time required to solve the control allocation problem.

The use of the previous optimum and active set is included in Algorithm 6.2. However, problems can occur in the case of time varying constraints. In the control allocation problem presented here, the constraints depend on the normal forces F_{zi}, which are time dependent. Time variations of the constraints also occur when rate constraints are incorporated. This time variation presents two problems:

- The previously calculated optimum $u(t-1)$ may no longer be feasible at time t and can thus no longer be used as a starting point for the algorithm
- Elements of the optimum $u(t-1)$ that were saturated (meaning that their corresponding constraints were active) may no longer be saturated at time t. The active set at time $t-1$ may therefore no longer be valid at time t and may require updating.

To avoid these problems, the algorithm for the solution of the WLS problem may be modified to include a feasibility check for the starting point of

the algorithm as well as updating of the active set. Algorithm 6.3 includes these modifications.

Algorithm 6.3: Solution of the WLS control allocation problem (6.24)

Let u^0 be the optimal point obtained at time $t-1$, and $\mathcal{W}^0$ be the corresponding active set ;
if $\underline{u}(t) < u^0 < \overline{u}(t)$ **then**
  Remove any active constraints from $\mathcal{W}^0$;
else
  Saturate the infeasible elements of u^0 and update the initial working set $\mathcal{W}^0$;
Rewrite the cost function as:

$$||W_u(u-u_d)||_2^2 + \gamma||W_v(Bu-v')||_2^2 = \left\| \underbrace{\begin{pmatrix} \gamma^{\frac{1}{2}} W_v B \\ W_u \end{pmatrix}}_{A} u - \underbrace{\begin{pmatrix} \gamma^{\frac{1}{2}} W_v v \\ W_u u_d \end{pmatrix}}_{b} \right\|^2$$

Use Algorithm 6.1 to solve:

$$\begin{aligned} u =& \arg\min_u ||Au-b|| \\ & \underline{u} \leq u \leq \overline{u} \end{aligned}$$

Implementational Aspects

In this chapter the actual control signals are taken to be the braking forces at each wheel. In reality, force demands cannot be assigned directly but must be passed to an ABS system or slip controller. In the current implementation, force commands are translated to brake pressure commands based on the parameters of the braking system, and given values of the road-tire friction and the friction between the brake pads and rotors. The commands required by the ABS system depend on the type of controller used. Modern ABS systems control wheel slip to specified reference values.

It is clear that the road-tire friction coefficient is required. Friction estimation is treated in Chapter 10, and is in general a difficult problem, but in the case of rollover mitigation there are certain factors which simplify the situation. Firstly, in order for rollover to occur, high friction is required. The activation of the rollover controller therefore implies a certain minimum friction level. In addition, once braking begins, full uti-

lization of the friction is virtually guaranteed, allowing for straightforward estimation. This implies that a nominal, high friction value can be used when the controller is activated, which may then be rapidly updated as braking begins.

As regards the commands required by the ABS system, certain uncertainties must be taken into account. For ABS systems using force or pressure references, the value of the friction coefficient between the pads and the rotor is important. This value may vary considerably, which clearly affects the force transmitted from the brake calipers to the wheel. For ABS systems requiring reference slips, there is the question of how the references should be generated. This is discussed briefly in Chapter 10.

6.5 Summary

The control allocation methods presented in this chapter are used to convert the virtual control signals into actuator inputs. Methods based on convex optimization have been investigated. It was seen that a direct approach, using the existing expressions for actuator constraints, yields a convex optimization problem with convex constraints. While efficient methods for solving problems of this type exist, it is nevertheless slightly too computationally intensive to be used in a real-time application with fast sampling rates.

By making some approximations, a linearly constrained quadratic programming problem has been posed. It was seen that existing active set algorithms are capable of solving such problems very efficiently. Nevertheless, these algorithms are more general than is required by the specific problem at hand; in particular, they allow the presence of equality constraints, which do not appear in the control allocation formulation. In addition, it was seen that certain issues arise when the algorithm is used with hotstarting, in the presence of time-varying constraints. A workaround for this has been presented, but there is room for improvement.

In the following chapter, an active set algorithm tailored to the specific optimization problem at hand is developed.

7

Active Set Methods for Control Allocation

In Chapter 6 it was shown how careful formulation of an optimization problem for control allocation yielded a problem for which the solution may be calculated in real time. The solution was obtained using existing active set methods for a general class of least squares problems with linear equality and inequality constraints, as described in [Björck, 1996].

Although these numerical methods are efficient, it was noted in Chapter 6 that the optimization problem in question contained no equality constraints. In general, control allocation problems based on quadratic cost functions and simple bound constraints on the actuators do not contain any equality constraints. These types of problems may be denoted as bound-constrained quadratic programming problems (BCQP), having the form:

$$\begin{aligned} \min \quad & q(u) \\ \text{subject to} \quad & \underline{u} \leq u \leq \overline{u} \end{aligned} \tag{7.1}$$

where $\underline{u}$ and $\overline{u}$ are the lower and upper bounds on the control input u respectively, and $q(u)$ is a convex quadratic cost function.

Although this class of problem appears limited, it is in fact suitable for a large number of control allocation applications. Rate constraints also lead to simple bound constraints (which are different from sample to sample) and can therefore also be handled, as described in Chapter 6.

Since the active set algorithm used in Chapter 6 is more general than required, it may be asked whether a more specific numerical method may be developed for the solution of the type of problem of interest here. Development of such a numerical method is described in this chapter.

7.1 Problem Formulation

Classical active set algorithms typically only make one change to the working set each iteration, which can result in a large number of iterations if there are many variables, or if the initial working set was very different from the optimal active set [Björck, 1996]. For large scale problems this is a major issue and considerable research has been done to find algorithms that are capable of identifying the optimal active set more quickly. This is often done using gradient projection methods [Moré and Toraldo, 1989]. For small scale systems, this problem has received less attention, since efficient algorithms already exist. For offline applications, existing algorithms are sufficient. The use of these algorithms for the solution of control allocation problems in real-time does however raise the issue of computation time even for small scale systems.

The performance of classical active set algorithms in real-time settings such as model predictive control and control allocation is greatly improved by ability to re-use the optimal solution and active set obtained in the previous sampling instant as starting points. This is known as hotstarting. However, when the allocation problem is time-varying, the use of hotstarting can cause problems, as discussed in Chapter 6. This is particularly true when time-varying constraints (such as rate constraints) are present. Such problems may be avoided by performing certain checks before running the active set algorithm in each sample. In particular, the feasibility of the starting point must be checked. If the constraints are different, the solution from the previous sample may not be feasible for the new problem. In addition, previously active constraints may no longer be active. If such conditions are violated, decisions must be taken as to how the starting point should be chosen, which will naturally affect the operation of the algorithm. Such a solution is not particularly elegant.

Another issue with the use of active set methods in real-time applications is the relatively large variability in the number of iterations required to find the optimum. This has been noted in model predictive control, where interior point methods are sometimes preferred for this reason.

As previously mentioned, the active set methods used thus far are intended for problems including equality constraints, which are not present in the control allocation problem structure developed in Chapter 6. This, in addition to the more practical problems discussed above, motivate the investigation of new types of numerical methods, tailored to the specific optimization problem at hand, and with a focus on performance in real time applications.

Recall from Chapter 6 that the control allocation problem could be

formulated as a Weighted Least-Squares (WLS) problem:

$$u = \arg\min_{\underline{u} \leq u \leq \overline{u}} \left(||W_u(u - u_d)||_2^2 + \gamma ||W_v(Bu - v)||_2^2 \right) \tag{7.2}$$

The cost function may be rewritten as:

$$\begin{aligned} &||W_u(u - u_d)||_2^2 + \gamma ||W_v(Bu - v)||_2^2 \\ &= \left\| \underbrace{\begin{pmatrix} \gamma^{\frac{1}{2}} W_v B \\ W_u \end{pmatrix}}_{A} u - \underbrace{\begin{pmatrix} \gamma^{\frac{1}{2}} W_v v \\ W_u u_d \end{pmatrix}}_{b} \right\|_2^2 \end{aligned} \tag{7.3}$$

which allows the minimization problem to be written as:

$$\begin{aligned} \min \quad & ||Au - b||_2^2 \\ \text{subject to} \quad & \underline{u} \leq u \leq \overline{u} \end{aligned} \tag{7.4}$$

This is exactly the form in (7.1), with $q(u) = ||Au - b||_2^2$.

7.2 Active Set Algorithms

In Chapter 6 it was shown how a typical control allocation formulation can take the form of a bound-constrained quadratic programming problem. In this section, the solution of such problems using active set methods will be examined. First, a general description of active set methods is given. A more specific algorithm, summarized in [Björck, 1996] and used in the context of control allocation in [Härkegård, 2003], is then examined.

Classical Primal Active Set Algorithm

In this section a general description of active set algorithms is given, following the presentation in [Moré and Toraldo, 1989]. More details can be found in [Björck, 1996] and [Hager *et al.*, 1996]. This type of algorithm is sometimes referred to as the classical primal active set algorithm (CPASA).

The principal of operation of all active set methods is that at each iteration, the active inequality constraints are regarded as equality constraints, and the remaining constraints are disregarded. In the case of bound-constrained problems, an active constraint corresponds to a variable holding a constant value, which simplifies the solution since variables corresponding to the active constraints are simply removed from the resulting optimization problem. Once a minimum to this problem is found,

the Karush-Kuhn-Tucker (KKT) optimality conditions are checked. If they are fulfilled, then the algorithm stops, otherwise, one of the active constraints breaking the KKT conditions is removed. For bound-constrained quadratic programming problems, the KKT conditions are equivalent to:

$$\begin{aligned} \frac{\partial q(u_j)}{\partial u_j} &= 0, \quad \underline{u}_j < u_j < \overline{u}_j \\ \frac{\partial q(u_j)}{\partial u_j} &\geq 0, \quad u_j = \underline{u}_j \\ \frac{\partial q(u_j)}{\partial u_j} &\leq 0, \quad u_j = \overline{u}_j \end{aligned} \tag{7.5}$$

Several definitions may be made. The *working set* $\mathcal{W}_k$ at iteration k is a subset of the *active set* $\mathcal{A}(u^k)$:

$$\mathcal{A}(u^k) = \{i : u_i = \underline{u}_i \text{ or } u_i = \overline{u}_i\}$$

Variables in $\mathcal{W}_k$ are known as *bound* variables. Variables not in $\mathcal{W}_k$ are known as *free* variables. It is also useful to define the *binding set* $\mathcal{B}(u)$ as:

$$\begin{aligned} \mathcal{B}(u) = \{i : u_i = \underline{u}_i \text{ and } \partial_i q(u) \geq 0 \text{ or } u_i = \overline{u}_i \\ \text{and } \partial_i q(u) \leq 0\} \end{aligned} \tag{7.6}$$

The starting point u^0 of the algorithm is assumed to be feasible (such a starting point is trivial to find in the case of bound-constrained problems) and $\mathcal{W}_0 \subset \mathcal{A}(u^0)$. The next iterate u^{k+1} may be found by solving:

$$\begin{aligned} \min_p \quad & q(u^k + p) \\ \text{subject to} \quad & p_i = 0, i \in \mathcal{W}_k \end{aligned} \tag{7.7}$$

This corresponds to an unconstrained minimization problem in the free variables. Given the optimal perturbation p the next iterate u^{k+1} may be found. The standard method is to calculate an α:

$$\alpha = \max\{\alpha \in [0, 1] : \underline{u} \leq u^k + \alpha p \leq \overline{u}\} \tag{7.8}$$

from which the next iterate is found:

$$u^{k+1} = u^k + \alpha p$$

If $\alpha = 1$, the optimality conditions are checked. Otherwise, the new active constraint is added to the working set.

The algorithm used in Chapter 6 includes both equality and inequality constraints, and is presented in [Björck, 1996]. This algorithm will be used as a benchmark for comparison with the modified algorithm developed in this chapter. The problem is on the form:

$$\min_{u} \quad ||Au - b||_2 \tag{7.9a}$$

$$\text{subject to} \quad Bu = v \tag{7.9b}$$

$$\underbrace{\begin{pmatrix} I \\ -I \end{pmatrix}}_{C} u \geq \underbrace{\begin{pmatrix} \underline{u} \\ -\overline{u} \end{pmatrix}}_{U} \tag{7.9c}$$

The details of the active set algorithm used to solve the problem are given in Algorithm 7.1.

7.3 Modifications

The method of finding the next iterate described in (7.8) has several attractive features. Using this method guarantees that the cost function decreases at each iteration. Additionally, this method also works when equality constraints are present. Since equality constraints do not appear in the control allocation formulation however, there exists more freedom in the choice of the next iterate.

The modified algorithms for large-scale problems outlined in [Moré and Toraldo, 1989] are concerned with identifying the optimal active set quickly. In order to obtain a similar effect in the case of small scale problems, an alternative to (7.8) is proposed here.

The method in (7.8) can be interpreted as moving along a line between the current iterate and the unconstrained minimum until the boundary of the feasible set is encountered. This point becomes the next iterate, and the active constraint at this boundary is added to the working set. It is not obvious that this is the 'best' course of action in any sense. An alternative choice would be to saturate all free variables whose values at the unconstrained maximum lie outside the feasible set. This can be expressed as finding a matrix Γ:

$$\Gamma = \begin{pmatrix} \alpha^1 & 0 & \dots & 0 \\ 0 & \alpha^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \alpha^n \end{pmatrix} \tag{7.10}$$

Algorithm 7.1: Classical primal active set algorithm

Let u^0 be a feasible starting point, satisfying (7.9c) ;
for $i = 0, 1, 2, \ldots$ **do**
Given suboptimal iterate u^i, find the optimal perturbation p, considering the inequality constraints in $\mathcal{W}$ as equality constraints and ignoring the remainder. This is done by solving:

$$\begin{aligned} &\min_p ||A(u^i + p) - b||_2 \\ &Bp = 0 \\ &p_i = 0, \quad i \in \mathcal{W} \end{aligned}$$

if $u^i + p$ *feasible* **then**
Set $u^{i+1} = u^i + p$;
Compute Lagrange multipliers as:

$$A^T(Au - b) = \begin{pmatrix} B^T & C_0^T \end{pmatrix} \begin{pmatrix} \mu \\ \lambda \end{pmatrix}$$

where C_0 consists of the rows of C corresponding to the constraints in the active set;
if $\lambda \geq 0$ **then**
u^{i+1} is optimal solution;
Return $u = u^{i+1}$
else
Remove constraint corresponding to most negative λ from the working set $\mathcal{W}$;
else
Find $\alpha = \max\{\alpha \in [0, 1] : \underline{u} \leq u^i + \alpha p \leq \overline{u}\}$ and set $u^{i+1} = u^i + \alpha p$. Add bounding constraint to working set.
end

such that:

$$\alpha^i = \max\{\alpha^i \in [0, 1] : \underline{u} \leq u + \alpha^i p \leq \overline{u}\} \tag{7.11}$$

The update is then:

$$u^{k+1} = u^k + \Gamma p \tag{7.12}$$

An alternative interpretation is that this method picks the point in the feasible set which is the closest in the Euclidean sense to the unconstrained minimum. Let P^f denote the unconstrained minimum of the

problem in the free variables. Solve, for $u^f = \{u_i : i \notin \mathcal{W}_i\}$:

$$\begin{aligned} \min \quad & ||u^f - P^f||_2 \\ \text{subject to} \quad & \underline{u} \leq u^f \leq \overline{u} \end{aligned} \tag{7.13}$$

Denote the solution to this problem u^*. If P^f is feasible, then clearly $u^* = P^f$. Otherwise, one or more constraints will be active at u^*. The next iterate is obtained by setting $u_i^{k+1} = u_i^*$ for $i \notin \mathcal{W}_i$. The working set is then expanded by adding those active constraints for which the KKT conditions (7.5) are satisfied. This method of updating the iterate may be directly substituted in the standard CPASA algorithm. The problem to be solved is:

$$\min_u \quad ||Au - b||_2 \tag{7.14a}$$

$$\text{subject to} \quad \underbrace{\begin{pmatrix} I \\ -I \end{pmatrix}}_{C} u \geq \underbrace{\begin{pmatrix} \underline{u} \\ -\overline{u} \end{pmatrix}}_{U} \tag{7.14b}$$

This modified algorithm is outlined in Algorithm 7.2.

Convergence Properties

It will now be shown that this method of updating the iterate has attractive properties, including convergence to the optimum in a maximum of $2n - 1$ steps, when starting with an empty working set. To prove this, the following lemma is needed.

LEMMA 7.1
Let u^* denote the solution to (7.13) for a given set of free variables u^f associated with a problem of the form (7.4). Then, at least one of the constraints active at u^* will also be active at the solution of (7.4).
Proof: Assume without loss of generality that $P_a^f \succeq \overline{u}_a$ for some set of indices a. This implies that $u_a^* = \overline{u}_a$. Assume now that no constraints active at u^* are active at the solution to (7.4), denoted u^{opt}. This implies $u_a^{opt} \succ \overline{u}_a$. Since u^{opt} must lie on the boundary of the feasible set, there must exist a separating hyperplane, passing through u^{opt}, which separates the feasible set from a level set $\mathcal{C}$, containing u^{opt}, of the original objective function in (7.4). Since P^f lies within $\mathcal{C}$, the line $P^f - u^{opt}$ must also lie in $\mathcal{C}$, by convexity. But since $P_a^f \succeq \overline{u}_a \succ u_a^{opt}$ there must exist a δ such that $u_a^{opt} + \delta(P_a^f - u_a^{opt})$ lies in both $\mathcal{C}$ and the feasible set. Thus no separating hyperplane exists at u^{opt}. If one or more of the constraints active at u^* were active at u^{opt}, then it is no longer generally true that $u_a^{opt} + \delta(P_a^f - u_a^{opt})$ lies within the feasible set, and a separating hyperplane may then exist. □

Algorithm 7.2: Modified active set algorithm

Let u^0 be a feasible starting point, satisfying (7.14b) ;
for $i = 0, 1, 2, \ldots$ **do**
  Given suboptimal iterate u^i, find the optimal perturbation p, considering the inequality constraints in $\mathcal{W}$ as equality constraints and ignoring the remainder. This is done by solving:

$$\min_p ||A(u^i + p) - b||_2$$
$$p_i = 0, \quad i \in \mathcal{W}$$

  if $u^i + p$ *feasible* **then**
    Set $u^{i+1} = u^i + p$;
    Compute Lagrange multipliers as:

$$A^T(Au - b) = C_0^T \lambda$$

    where C_0 consists of the rows of C corresponding to the constraints in the active set;
    if $\lambda \geq 0$ **then**
      u^{i+1} is optimal solution;
      Return $u = u^{i+1}$
    else
      Remove constraint corresponding to most negative λ from the working set $\mathcal{W}$;
  else
    Find $\alpha^i = \max\{\alpha^i \in [0, 1] : \underline{u} \leq u + \alpha^i p \leq \overline{u}\}$ and set $u^{k+1} = u^k + \Gamma p$;
    Compute Lagrange multipliers for the active constraints;
    Add constraints satisfying KKT conditions to working set.
end

This lemma formalizes the idea that the proposed updating method acts to find the optimal active set quickly. It can be thought of as identifying the constraints which are 'closest' to the unconstrained minimum of the original problem.

PROPOSITION 7.1
The active set algorithm with updating as in (7.13) and with the initial working set empty, converges in a maximum of $2n - 1$ iterations where n is the number of optimization variables.
Proof: Lemma 1 shows that in each iteration, at least one of the constraints

that become active will be active at the optimum of the original problem. Only those constraints which satisfy the KKT conditions (7.5) at u^* are added to the working set. The constraints satisfying the KKT conditions at u^* may not necessarily be optimal at u^{opt} however. Consider the case where n constraints are active, of which only one which remains active at u^{opt}. The worst case occurs when the $n-1$ constraints which will *not* be active at u^{opt} fulfill the KKT conditions at u^* and are added to the working set, while the remaining constraint does not. In this case $n-1$ future iterations will be required to remove these constraints from the working set, since only one constraint may be removed at each iteration. An additional $n-1$ iterations are required to locate the optimum, giving a total of $2n-1$ iterations in the worst case. □

Properties relevant to Real-Time Applications

As previously mentioned, standard active set algorithms used in real-time settings benefit significantly from the use of hotstarting, where the results from the previous sampling instant are used as a starting point for the next sampling instant. Complications arise, however, when the constraints vary with time. Time varying constraints can arise when rate constraints are present [Härkegård, 2003; Schofield, 2006], or when the constraints depend on time varying parameters. When constraints vary from sample to sample, using the solution from the previous sample is not straightforward. For instance, the solution may no longer be feasible with respect to the new constraints, or certain constraints active at the previous solution may no longer be active. Such situations can violate the starting point assumptions of the active set algorithm and lead to incorrect behaviour. While logical checks could be carried out within the algorithm to ensure that starting conditions are met, this would influence the behaviour of the resulting algorithm. The proposed algorithm avoids this problem by removing the need for hotstarting. In terms of computational complexity, the modified algorithm adds only an addition check of the KKT conditions. The work done per iteration is comparable to the original algorithm. On the other hand, faster identification of the optimal active set means that fewer iterations are required in virtually all cases.

Two Dimensional Example

In order to visualize the operation of the proposed modification it is useful

to examine a two-dimensional example. Consider the problem defined by:

$$
\begin{aligned}
A &= \begin{pmatrix} 1 & 3 \\ 5 & 7 \end{pmatrix}, \quad b = \begin{pmatrix} 50 \\ 50 \end{pmatrix} \\
\underline{u} &= \begin{pmatrix} -10 \\ -10 \end{pmatrix}, \quad \overline{u} = \begin{pmatrix} 10 \\ 10 \end{pmatrix} \\
u_d &= \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \gamma = 1000
\end{aligned}
\tag{7.15}
$$

The problem was solved using Algorithm 7.1, implemented in Matlab code in the Quadratic Control Allocation Toolbox (QCAT) [Härkegård, 2004], as well as with Algorithm 7.2.

The operation of the standard algorithm is illustrated in Figure 7.1. From an initial starting point, the next iterate is found by following the line connecting the starting point and the unconstrained minimum until the boundary of the feasible set is encountered. At this point a new constraint, $u_1 = \underline{u}_1 = -10$ is added to the working set. In the next iteration, u_2 is the only free variable, and the minimum of the unconstrained cost function in this subproblem is also outside the feasible set, so the constraint $u_2 = \overline{u}_2 = 10$ is added to the working set. In the following iteration, there are no free variables, so the constraint with the most negative Lagrange multiplier ($u_1 = \underline{u}_1$) is removed from the working set. In the final iteration, the unconstrained minimizer is feasible. The KKT conditions are checked and found to be fulfilled, and the algorithm terminates.

Figure 7.2 illustrates the operation of the modified algorithm, starting from the same initial point. In the first iteration, the point in the feasible set closest to the unconstrained minimum is found. Although two constraints are active at this point ($u_1 = \underline{u}_1$ and $u_2 = \overline{u}_2$), only the constraint on u_2 fulfills the KKT conditions, and is added to the active set. In the following iteration the minimum is found to be feasible, and the KKT conditions are satisfied. This simple example illustrates the advantages of the modified algorithm over the standard algorithm, principally that fewer iterations are required. Although the number of iterations required in both cases depends on the starting point, the modified algorithm nevertheless requires the same or fewer iterations. The only situation in which the original algorithm takes fewer iterations is when the initial point lies on the line connecting the unconstrained minimum with the optimal solution. In this example the standard algorithm would find the optimum in one iteration while the modified algorithm would require two.

Rollover Mitigation Controller

The new algorithm has also been tested with the rollover mitigation controller developed in Chapters 5 and 6. A detailed presentation of the sim-

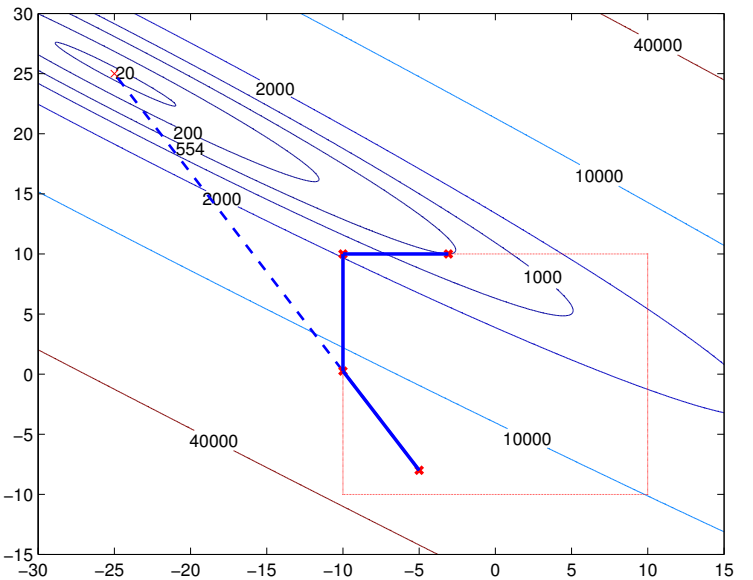

Figure 7.1 The two-dimensional example (7.15), solved using the standard algorithm. The path taken by the iterates is rather indirect.

ulation results will be given in Chapter 8, but a comparison between the new algorithm and the standard one will be presented here. A 10 second simulation of a standard 'Fishhook' maneuver was used for evaluation.

Both the standard active set algorithm and the modified version were used to solve the control allocation problem. In both cases the control performance is exactly the same, since the only difference is the method used to solve the allocation problem. Using hotstarting in this application leads to problems due to the time-varying constraints, as discussed in Chapter 6. Figure 7.3 illustrates the distribution of the number of iterations required by the respective algorithms, with and without hotstarting. The modified algorithm most often requires fewer iterations than the standard algorithm, and never required more than six iterations, one less than the theoretical worst case of seven. The average number of iterations required by the standard algorithm without hotstarting was 4.9, while the average for the modified algorithm was 3.4. With hotstarting, and the additional logical checks described above, the average number of iterations required were 2.9 for the standard algorithm and 2.4 for the proposed algorithm.

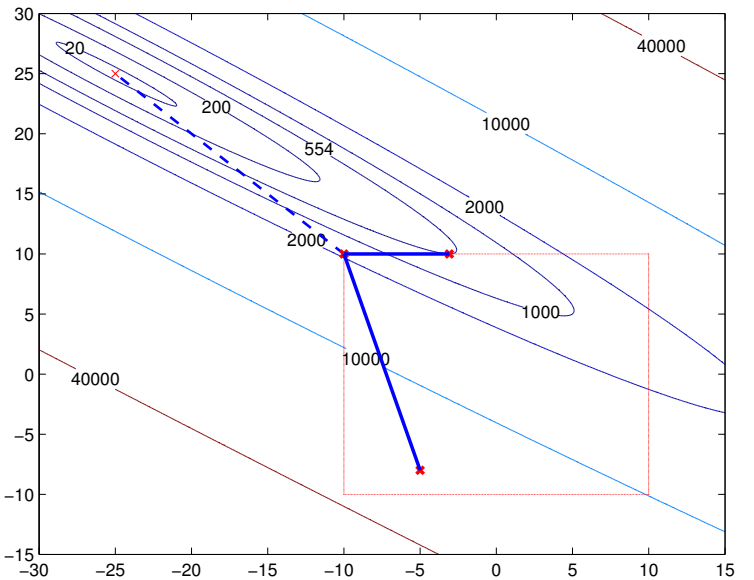

Figure 7.2 The two-dimensional example (7.15), solved using the modified algorithm. This example illustrates the advantage of moving to the point in the feasible set closest to the unconstrained maximum, which is the salient feature of the modified algorithm. This avoids interim steps on the boundary of the feasible set as in Figure 7.1.

From the perspective of real-time implementation, the modified algorithm is clearly preferable. Although the optimization problems are small and can be solved quickly even by the standard algorithm, applications such as vehicle dynamics control typically use sample times of 10ms, meaning that even small improvements can be important. In particular, the variance in the number of iterations required is smaller for the new algorithm, and the maximum number of iterations required is lower.

Discussion

The results clearly indicate the improved performance of the proposed algorithm. Good performance is achieved without the need for hotstarting, which is important in this application where user checks are required to prevent problems with hotstarting using the standard algorithm. It can be seen that the particular hotstarting method used here improved performance, but in general it is desirable to avoid hotstarting in such applications.

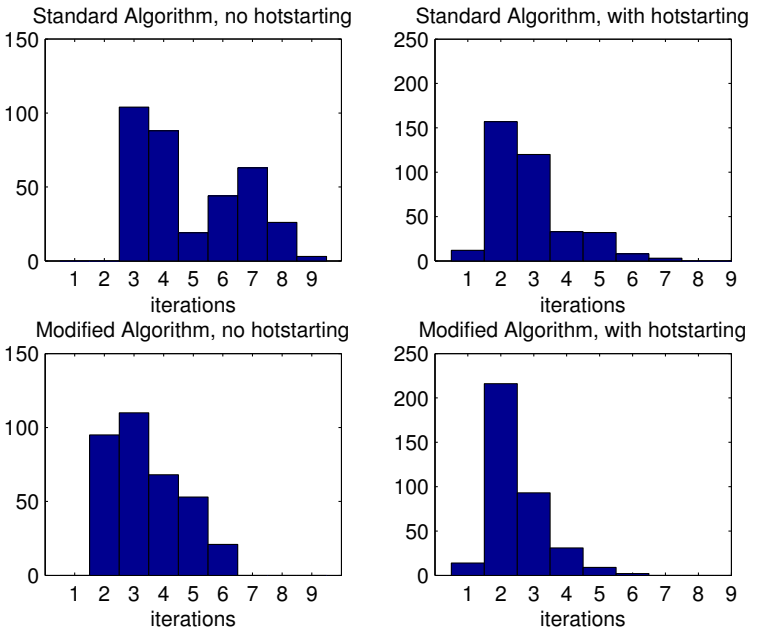

Figure 7.3 Histogram showing the number of iterations required by the standard and modified algorithms during the rollover mitigation controller simulation. The theoretical maximum number of iterations for the modified algorithm, 7 in this case, is never required. Note that controller performance is exactly equal in all cases, that is, the algorithms obtain the correct solutions to the allocation problems. Controller performance could, however, be affected if the computation time exceeds the sample time.

7.4 Conclusions

A new active set algorithm for solving bound-constrained least squares problems has been presented in this chapter. Unlike existing algorithms, the proposed algorithm does not require hotstarting to obtain good performance. This is particularly important in time-varying allocation problems, where the use of hotstarting can cause problems. The key feature in the proposed algorithm is the method for updating the iterates, which allows the addition of multiple constraints to the active set at each iteration. The algorithm differs from classical active set algorithms in that the next iterate is found by minimizing the distance from the feasible set to the unconstrained minimum of the current subproblem, rather than by moving along the line between the current iterate and the unconstrained mini-

mum. The aim of this modification is to allow faster identification of the optimal active set, and allow the addition of multiple constraints to the working set at each iteration. It is shown that the algorithm finds the optimum in at most $2n-1$ iterations, where n is the number of optimization parameters, given that the algorithm starts with an empty working set.

8

Simulation Results For Rollover Mitigation

To fully evaluate the performance of any vehicle control algorithm, experiments with real vehicles are required. Initial design evaluations are however typically carried out with the help of simulations. Modern computing tools allow the use of very advanced vehicle simulations, which accurately reproduce the behaviour of actual vehicles. Testing via simulation is therefore an important step in the development of new algorithms. In this chapter, simulation results of the proposed strategies are presented.

The simulations presented in this chapter were carried out in Matlab-Simulink using DaimlerChrysler's CASCaDE (Computer Aided Simulation of Car, Driver and Environment) software. CASCaDE is an advanced vehicle simulator, incorporating accurate tire models and full degree-of-freedom chassis models. In addition to the vehicle dynamics, the simulator includes other control systems such as Anti-lock Braking Systems (ABS). This is extremely important since these subsystems affect the control system outputs, in this case the braking forces. The vehicle used in the simulations was a commercial van with a gross weight of roughly 3500kg. The standard vehicle setup used in the simulations was a vehicle with an additional load of 420kg. The vehicle parameters used are summarized in Table 8.1. The controller parameters used in all the simulations are summarized in Table 8.2. A number of test maneuvers were simulated. These will be described in the next section.

8.1 Test Maneuvers

Due to the dangerous nature of rollover accidents, a number of vehicle safety organisations have evaluated the performance of production vehi-

Table 8.1 Vehicle parameters used in the simulations, with the default load of 420kg.

Symbol	Description	Value	Unit
m_e	Vehicle mass (empty)	2800	kg
h_e	CG height (empty)	0.79	m
m_b	Load mass	420	kg
h_b	Load height	1	m
a_b	Load distance from front axle	4.2	m
I_{xx}	MI about x–axis	2275	kgm^2
I_{yy}	MI about y–axis	13400	kgm^2
I_{zz}	MI about z–axis (with load)	16088	kgm^2
a	CG distance from front axle	1.58	m
b	CG distance from rear axle	1.97	m
l	Half track width	0.8126	m
C_ϕ	Roll stiffness	221060	Nm/rad
K_ϕ	Roll damping	12160	Nms/rad

cles to assess their safety. In order to obtain a common measure, a number of standardized maneuvers have been developed. The National Highway Traffic Safety Administration (NHTSA) has developed various standard maneuvers, including the so-called fishhook and J-turn maneuvers, which are described here.

Fishhook

The fishhook maneuver is an important test maneuver in the context of rollover. It attempts to maximize the roll angle under transient conditions and is performed as follows, with a start speed of 80km/h[1]:

- The steering wheel angle is increased at a rate of 720deg/sec up to $6.5\delta_{stat}$, where δ_{stat} is the steering angle which is necessary to achieve 0.3g stationary lateral acceleration at 80km/h
- This value is held for 250ms
- The steering wheel is turned in the opposite direction at a rate of 720 deg/sec until it reaches -$6.5\delta_{stat}$

[1]The original specification from the NHTSA is given in imperial units and states a start speed of 50 mph. Metric conversions will be used in this thesis.

Table 8.2 Controller parameters used in the simulations

Symbol	Description	Value
K_r	Yaw rate controller gain	1
a_x^d	Desired longitudinal deceleration	$0.4g$
$\hat{a}_x^{\text{on}}$	Threshold for controller switching	7
$\hat{a}_x^{\text{off}}$	Threshold for controller switching	5
σ	Tuning parameter in (6.13)	1
ν	Tuning parameter in (6.13)	1
W_v	Weighting matrix in (6.24)	$[1 \quad 100 \quad 30]$
W_u	Weighting matrix in (6.24)	$[1 \quad 1 \quad 1 \quad 1]$
γ	Parameter in (6.24)	10^6

Figure 8.1 illustrates the driver input during a Fishhook maneuver. No brake or accelerator commands are given during the maneuver.

J-Turn

The J-turn is a simple step in the steering wheel angle driving the vehicle towards the physical limits. This maneuver can cause a roll over of vehicles with critical load. The speed of the vehicle just before the step input to the steering wheel angle is 60 mph (approximately 96 km/h). After releasing the accelerator pedal the steering wheel angle is increased at a rate of 1000deg/sec until it reaches 8 times the value δ_{stat}. The steering input for this maneuver is illustrated in Figure 8.2.

Roll Rate Feedback Fishhook

A modified version of the fishhook maneuver, known as the roll rate feedback fishhook maneuver, can also be specified. This maneuver is optimized for producing maximum vehicle roll. The sequence of events is similar to the standard fishhook, but the second steering angle change is performed only after the roll rate becomes small (that is, when the roll angle reaches its maximum). The maneuver is performed as follows, with a start speed of 80km/h:

- The steering wheel angle is increased at a rate of 720deg/sec up to 6.5 times δ_{stat}
- This value is held until the roll rate drops below 1.5 deg/sec

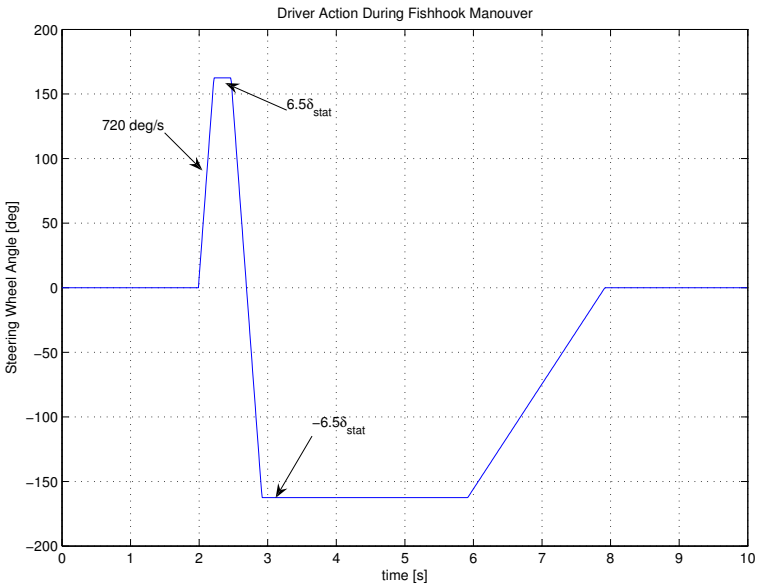

Figure 8.1 Driver input during the fishhook maneuver.

- The steering wheel is turned in the opposite direction at a rate of 720 deg/sec until it reaches -6.5δ_{stat}

8.2 Simulation Results

In this section simulation results relating to the fishhook and J-turn maneuvers are presented. The control strategies were found to work well for a large number of different test maneuvers, but priority will be given here to the fishhook and J-turn, as they are standard maneuvers, specifically intended for the investigation of both roll and yaw stability.

The results presented here use the control strategy outlined in Algorithm 5.1, and the control allocation strategy in Algorithm 6.3. The control allocation problem was solved using a C implementation of the active set methods described in Chapter 6. The implementation is based on the Quadratic Control Allocation Toolbox (QCAT) for Matlab written by Ola Härkegård, which contains implementations of the algorithms described in [Härkegård, 2003]. SVD methods are used to solve the least squares problems. The SVD code is based on the algorithms presented in [Wilkin-

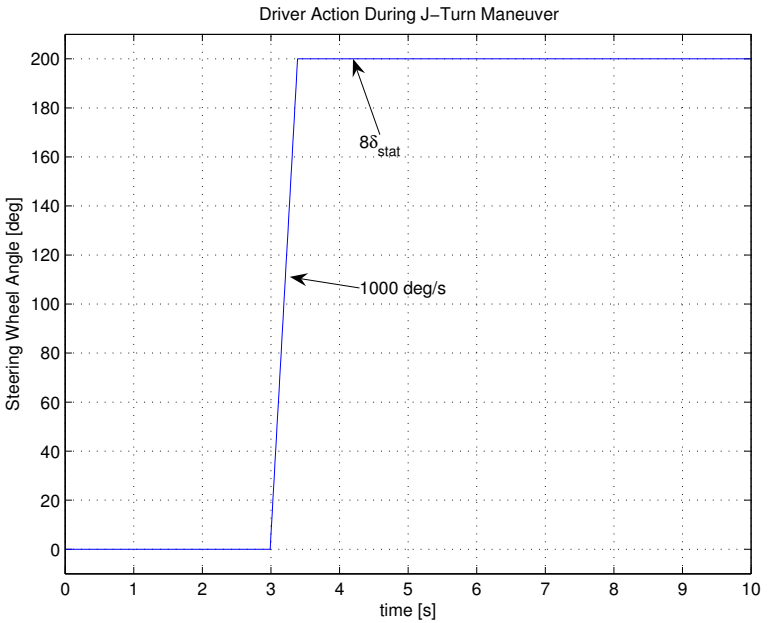

Figure 8.2 Driver input during the J-turn maneuver.

son and Reinsch, 1971], and the ports of these algorithms from ALGOL to C in [Bond, 2008].

The modified active set algorithm presented in Chapter 7 would be a more suitable choice for the implementation. The results presented here use the standard active set algorithm, since a C version of the modified algorithm was not written during the course of the project. The simulation results are of course unaffected by the choice of active set algorithm.

Fishhook Maneuver

The fishhook maneuver described in the previous section and illustrated in Figure 8.1 was the primary test maneuver used for evaluating and tuning the control strategies.

Controller Inactive Figure 8.3 shows the effect of the fishhook maneuver when the controller is inactive. Rollover occurs after approximately 4.5 seconds, just after the maximum value of the second steering action is attained. The severe instability of the roll dynamics can clearly be seen. The roll rate increases faster than linearly, which is consistent with the

analysis in Chapter 3 which predicted increasingly unstable roll dynamics as the roll angle increases.

Controller Active Figure 8.4 shows the states of the vehicle when the controller is active. The roll angle limit used was 0.1 radians. It can be seen that this limit is not exceeded during the maneuver.

The yaw rate tracking is reasonable, although the reference is not followed exactly. One of the aims of performing yaw rate control is the limitation of the sideslip angle β. A maximum value of β can be defined as [Kiencke and Nielsen, 2000]:

$$\beta_{max} = 10^\circ - 7^\circ \frac{(v_x^2 + v_y^2)}{(40)^2} \tag{8.1}$$

This maximum value of the sideslip value is illustrated in Figure 8.4. It can be seen that β remains within the allowed range throughout the maneuver.

Control Allocation Figures 8.5, 8.6 and 8.7 show the desired virtual controls F_{xT}, M_T and F_{yT}, as well as the generalized forces that were obtained using the resulting braking actions. The 'predicted' or expected generalized forces are also shown. These are calculated in the control allocation algorithm and are given by:

$$v_o = Bu_o$$

where v_o are the expected generalized forces, u_o are the actual controls obtained by the control allocator and B is the control effectiveness matrix. It is interesting to examine these signals to ascertain whether the model used in the control allocation algorithm is accurate.

It can be seen that both of the virtual control commands F_{xT} and M_T are met to a good degree of accuracy. In addition, the generalized forces calculated with the model used by the control allocation algorithm agree closely with the actual generalized forces that were obtained. This is particularly clear in the case of F_{yT}, illustrated in Figure 8.7. Although F_{yT} is not used as a virtual control signal, the force determined by the control allocation model is very close to the actual force generated. This may be seen as a validation of the approximations made about the friction characteristic in Chapter 6, namely, that the friction ellipse can be well approximated by piecewise linear functions during extreme maneuvering.

Constraints Figure 8.8 shows the forces acting on each of the tires. As expected, due to the load transfer the rear right tire loses contact with the

road for a time during the simulation. Since the control strategy is based on limitation of the roll angle and not on prevention of wheel lift-off, this is acceptable behaviour.

Figure 8.9 illustrates the desired and actual brake pressures during the simulation. The importance of rate constraints in the control allocation algorithm can be seen by comparing Figure 8.9 with Figure 8.10, in which the rate constraints have been deactivated. As can be seen in Figure 8.10, disregarding rate constraints can cause large discrepancies between the desired and actual brake pressures, which would result in degraded controller performance.

Switching Figure 8.11 shows the lateral acceleration during the simulation. Figure 8.12 shows both the lateral acceleration and the filtered version $\hat{a}_y$, given by (4.13), which is used for switching. It is clear that incorporating gradient information allows earlier switching, which gives improved performance.

Trajectory following The trajectories of the vehicle during the fishhook maneuver with and without control are shown in Figure 8.13. It can be seen that the vehicle follows the desired fishhook trajectory when the controller is active. A comparison of the entire trajectory between the controlled and uncontrolled vehicles is difficult since the uncontrolled vehicle rolls over during the maneuver. Under the assumption that the maneuver is performed for collision avoidance, it can be seen that the radii of curvature achieved by the controlled vehicle are comparable to or better than the uncontrolled vehicle.

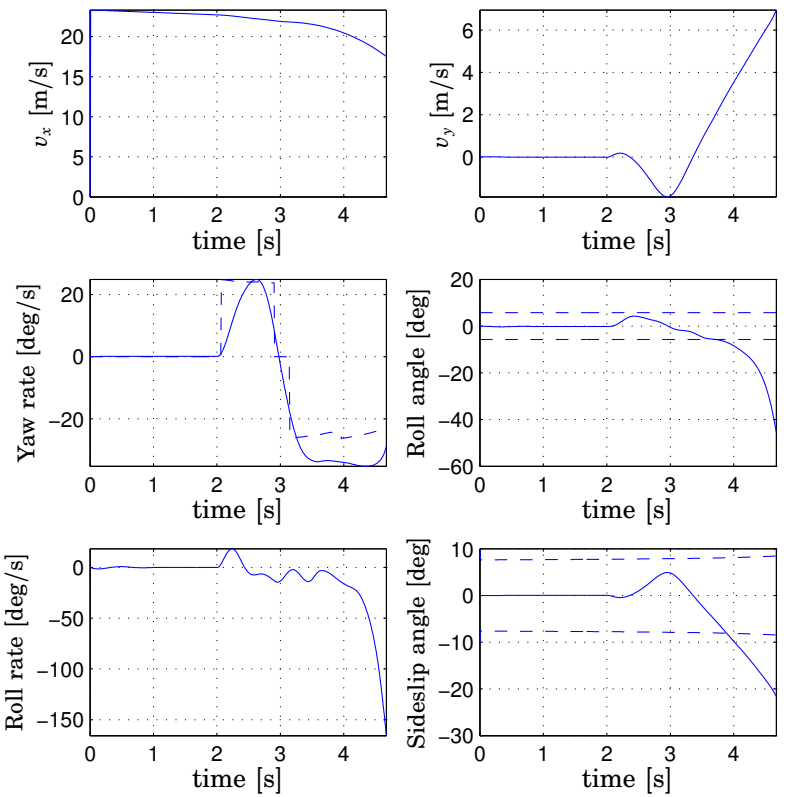

Figure 8.3 Vehicle states during the fishhook maneuver with the controller inactive. The severe instability of the roll dynamics is clearly visible. The rapid growth of both the roll angle and roll rate is evident. The simulation ceases to run after rollover occurs.

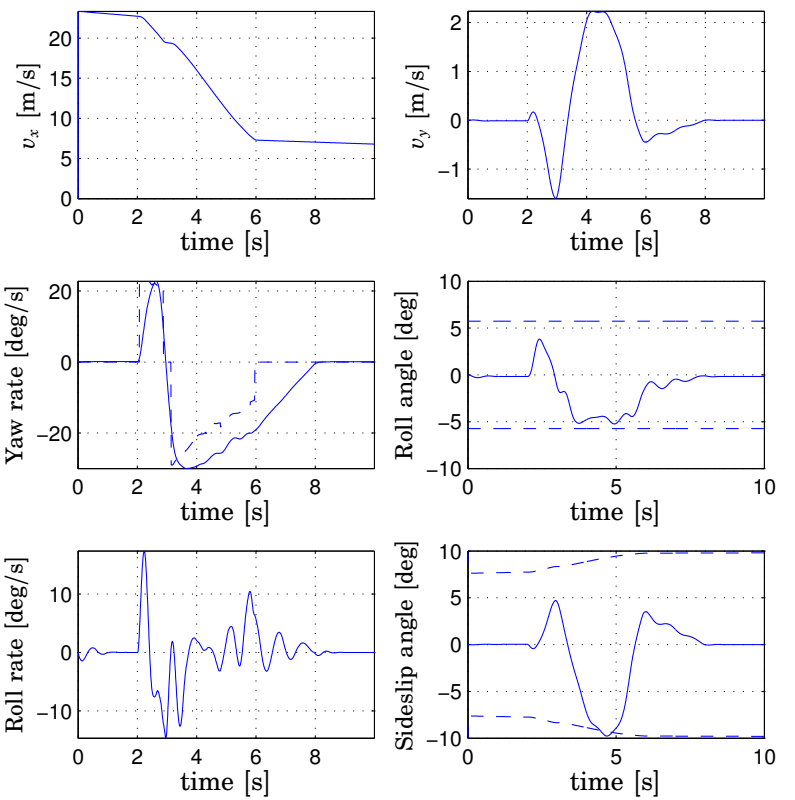

Figure 8.4 Vehicle states during the fishhook maneuver with the controller active. The dashed lines are the yaw rate reference, maximum allowed roll angle and maximum sideslip angle respectively. It is clear that the maximum allowed roll angle is never exceeded. In addition, yaw rate reference tracking is reasonable, and the sideslip angle limit in (8.1) is not exceeded.

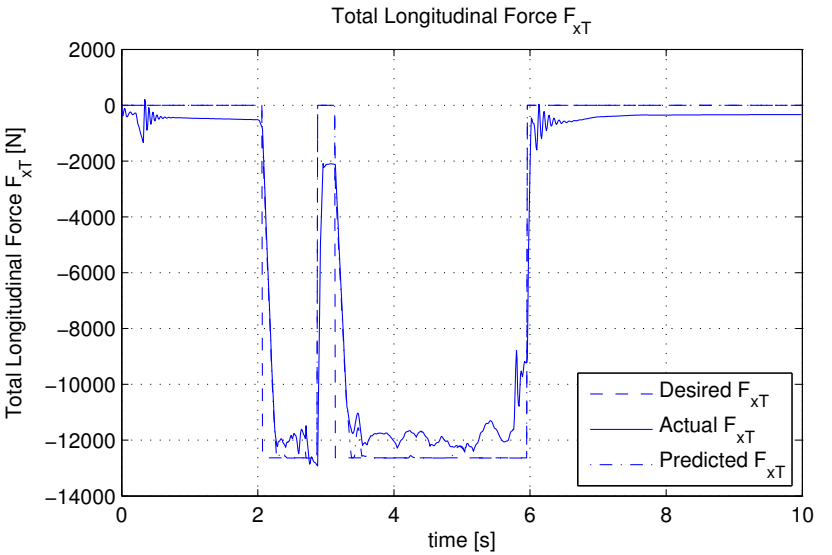

Figure 8.5 Total longitudinal force F_{xT} during the fishhook maneuver. The dashed line is the virtual control, the solid line is the obtained force. The dash-dotted line is the expected total force, calculated using the control allocation model.

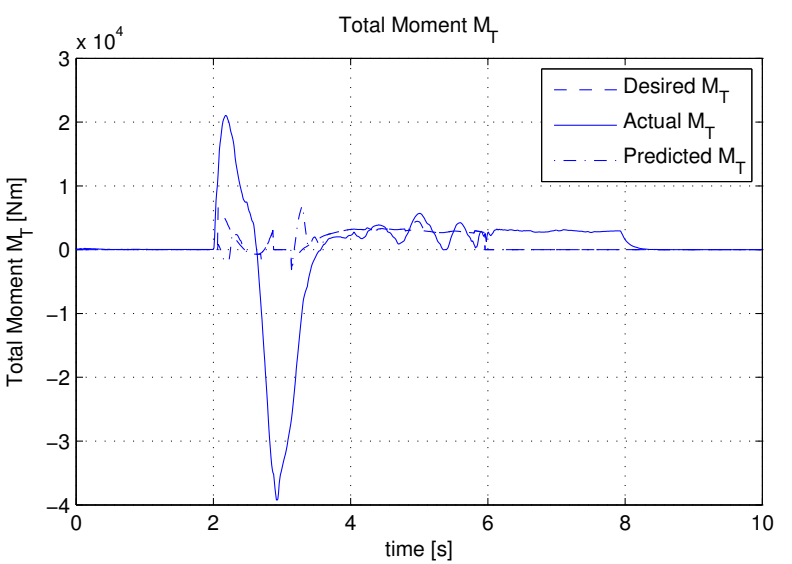

Figure 8.6 Total moment M_T during the fishhook maneuver. The spike at 3 seconds occurs when the controller is off.

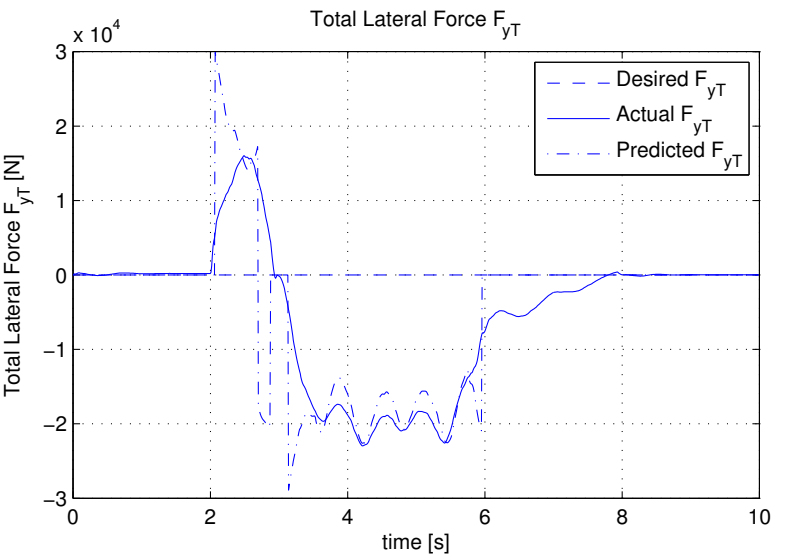

Figure 8.7 Total lateral force F_{yT} during the fishhook maneuver. The total lateral force is not used as a virtual control signal in this strategy. The virtual control command is set to zero, and the corresponding weight in the control allocator is chosen to be small. It is however interesting to observe that the predicted value of F_{yT} calculated by the control allocation algorithm (dotted line) corresponds well with the actual force (solid line). This confirms that the approximations of the friction characteristic, used in the design of the control allocator, are valid, at least during this maneuver.

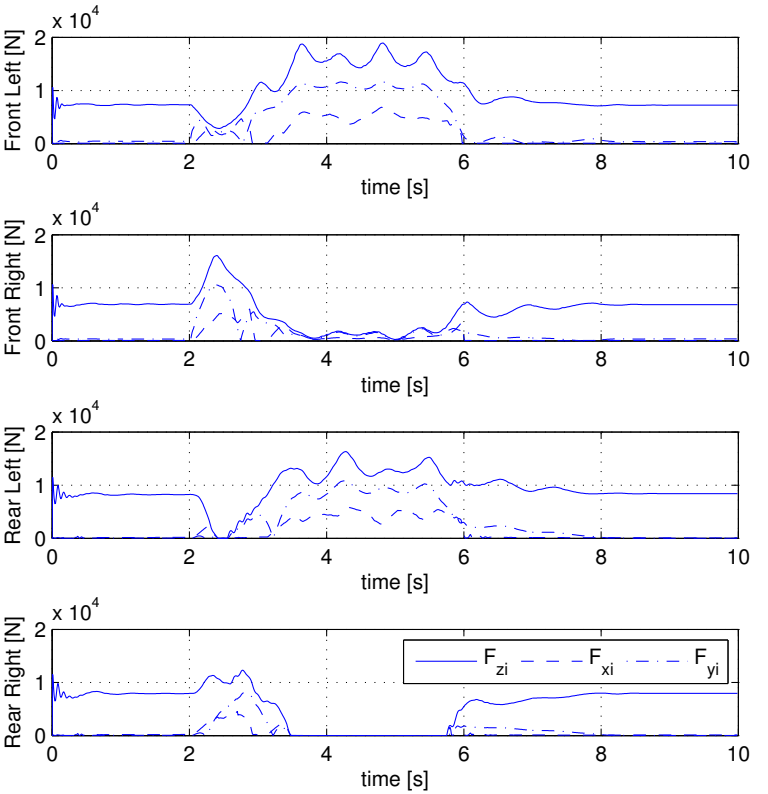

Figure 8.8 Tire forces during the fishhook maneuver with the controller active. The solid lines are the normal forces F_{zi} which, scaled with the friction coefficient μ, correspond to the maximum available force. The dashed lines are the longitudinal (braking) forces, and the dash-dotted lines are the lateral forces. Recall that the relations between these forces are quadratic, given by the friction ellipse.

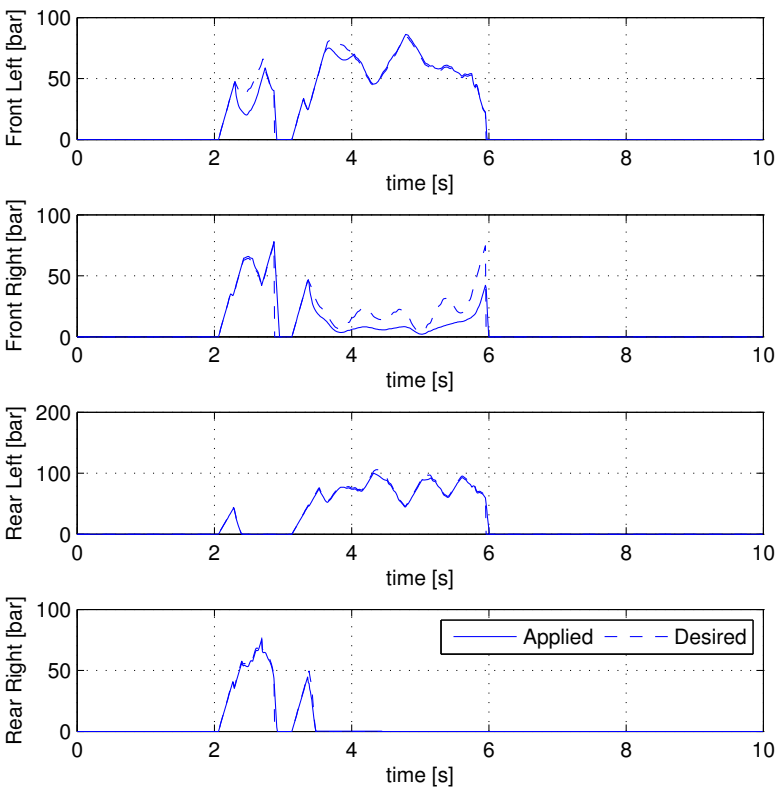

Figure 8.9 The brake pressures for each wheel during the fishhook maneuver. The dashed lines is the desired brake pressures, and the solid lines are the achieved brake pressures. The importance of rate constraints is clearly shown by comparing this figure with Figure 8.10, in which rate constraints are neglected. In this example, the constraints are assumed to be linear, with the constants known exactly. In reality, the constraints may be nonlinear, with unknown parameters.

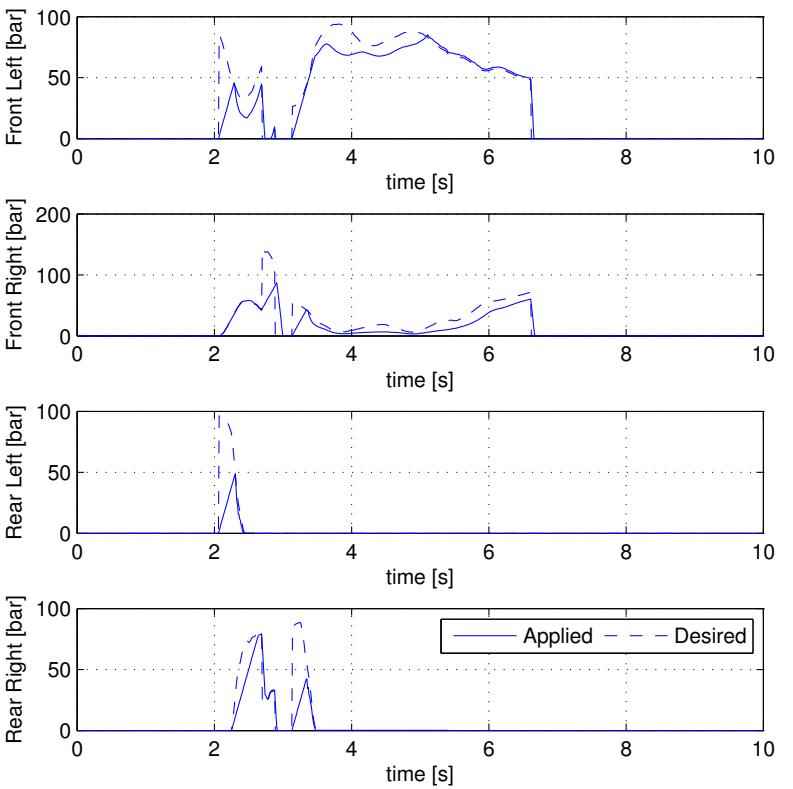

Figure 8.10 The brake pressures for each wheel during the fishhook maneuver, without rate constraints in the control allocator. The discrepancies between the commanded and achieved pressures are clearly seen. In particular, the desired brake pressures take large values initially, which are not possible to achieve. By including rate constraints, an optimum may be found which is achievable in reality.

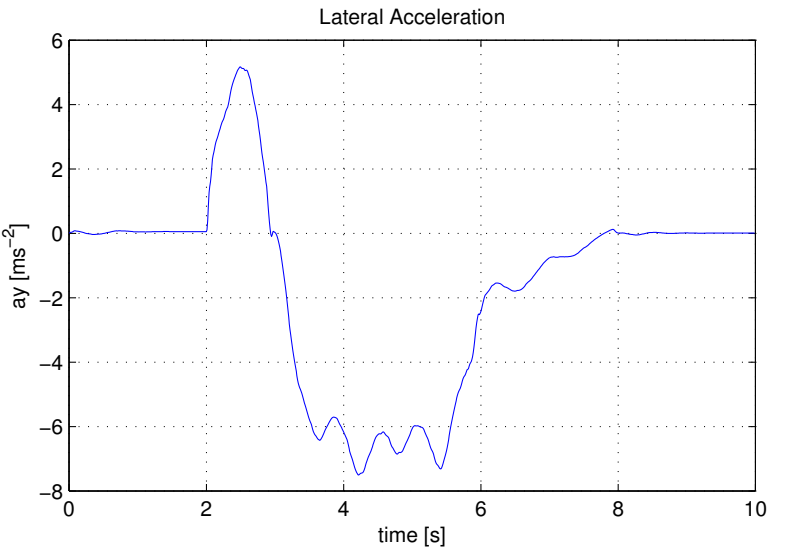

Figure 8.11 The lateral acceleration a_y.

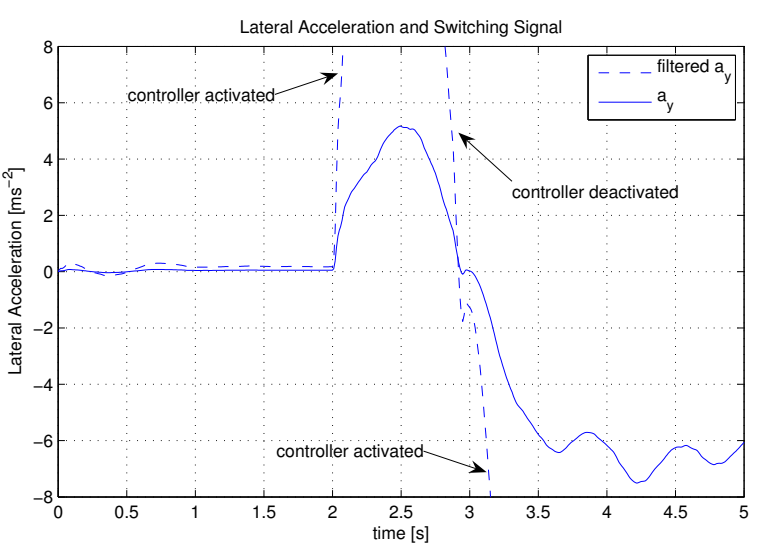

Figure 8.12 A close-up of the lateral acceleration a_y and switching signal $\hat{a}_y$.

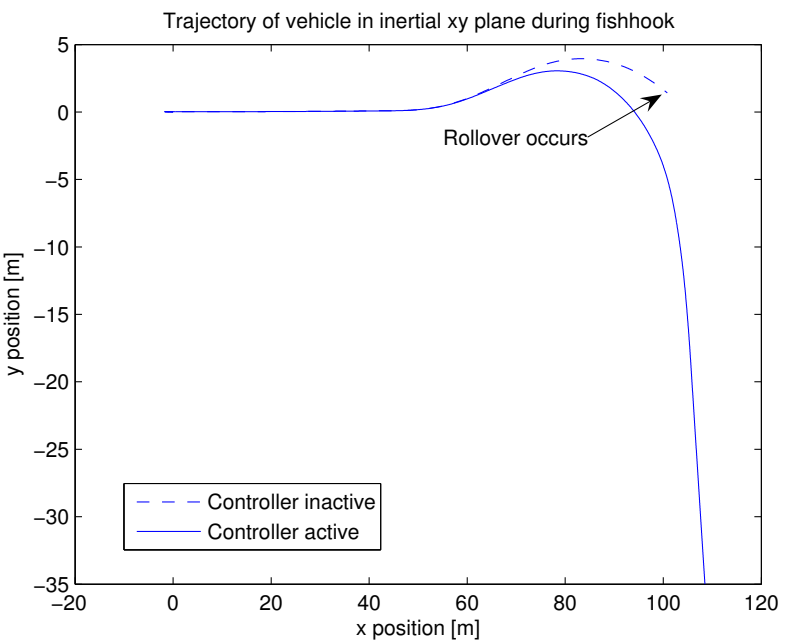

Figure 8.13 Vehicle trajectory in the inertial xy plane during the fishhook maneuver. The solid line corresponds to the trajectory of the vehicle when the controller is active, and the dashed line is the trajectory when the controller is inactive. It is interesting to note that the radius of curvature is acutally smaller in the controlled case. This is made possible in part by the reduction in speed caused by braking, and in part by the allocation of the braking forces to generate the desired moment.

J-Turn Maneuver

The control algorithm was tuned using the fishhook maneuver. To test the resulting algorithm, the J-turn maneuver was used. The vehicle states for the J-turn maneuver with the nominal load of 420kg with the controller inactive are shown in Figure 8.14. With the given loading conditions, rollover does not occur, but yaw instability occurs, resulting in a skid (as can be seen from the sideslip angle plot). The vehicle states for the same maneuver with the controller active are shown in Figure 8.15. Both the roll angle and the sideslip angle remain within the desired limits. The yaw rate does not track the reference exactly, but yaw stability is nevertheless preserved.

This result shows that the proposed VDC system is successful not only in preventing rollover, but also in stabilizing the yaw dynamics for maneuvers in which yaw instability arises without rollover.

Trajectory following Figure 8.16 shows the trajectory of the vehicle during the J-turn maneuver, with and without control. When the controller is active, the vehicle is able to perform a turn with a smaller radius of curvature than the uncontrolled vehicle, which skids. This is similar to the results found using the fishook maneuver, and again highlights the effects of performing control allocation, namely that yaw and roll stability may be preserved without compromising the vehicle's trajectory. It is easy to imagne that such behaviour could be critical in cases when extreme maneuvering is required for collision avoidance.

Robustness In order to test the robustness of the system to uncertainty in the loading conditions, simulations were made using a larger load of 860kg located at a height of 1.3m over the roll axis. The load parameters in the controller were left unchanged. The additional mass and greater height of the load increases the propensity for rollover. Figure 8.17 shows the states of the vehicle with the new loading conditions during a J-turn without control. The greater load causes a rollover to occur, whereas the same maneuver with a smaller load lead to a skid as in Figure 8.14. The effects of using the controller (without the correct load parameters) can be seen in Figure 8.18. The controller is capable of preventing rollover, and the yaw dynamics are stabilized. The roll angle is maintained within the desired limits, although the sideslip angle exceeds its limits for a time. This indicates that exact knowledge of the loading conditions is not required for the controller to be effective, at least in the context of this maneuver.

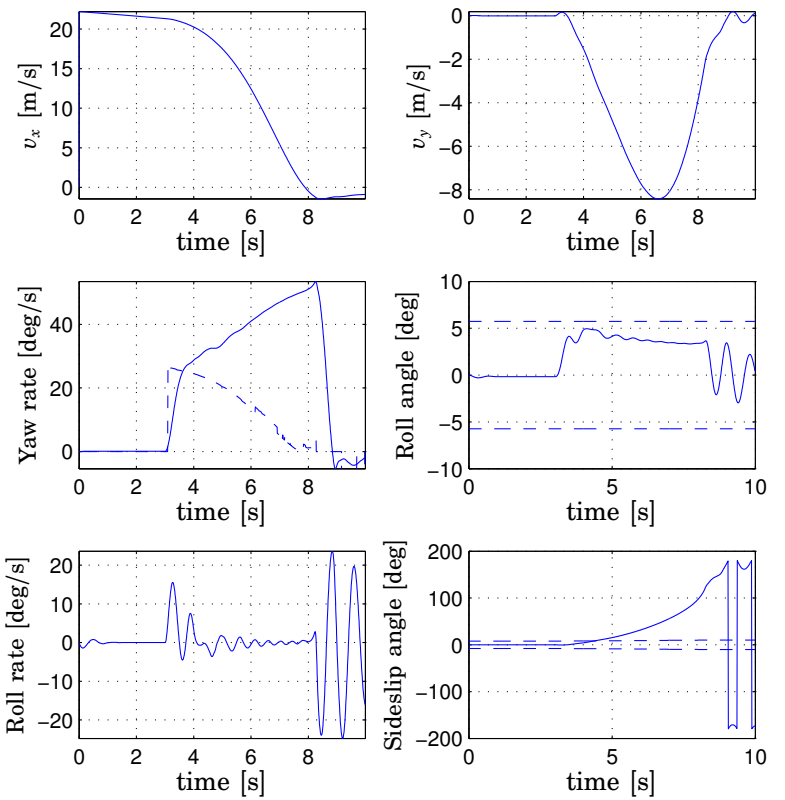

Figure 8.14 Vehicle states for the J-turn maneuver with the controller inactive. The instability of the yaw dynamics is clearly seen in the yaw rate and sideslip angle. Indeed, the vehicle skids and turns through over 180 degrees.

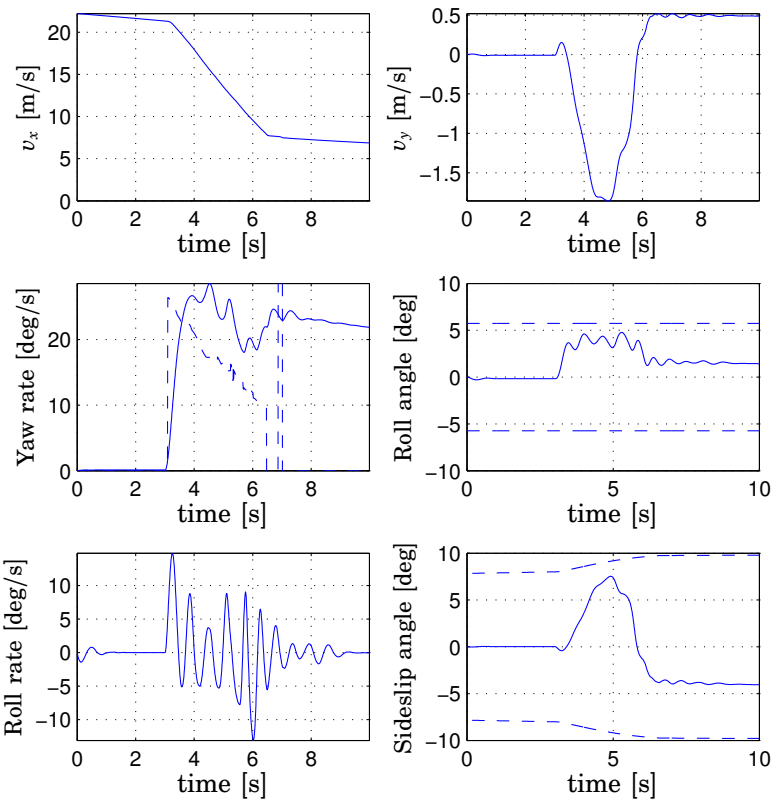

Figure 8.15 Vehicle states for the J-turn maneuver with the controller active. Both the roll angle and the sideslip angle remain within the prescribed limits. Yaw rate reference tracking is reasonable.

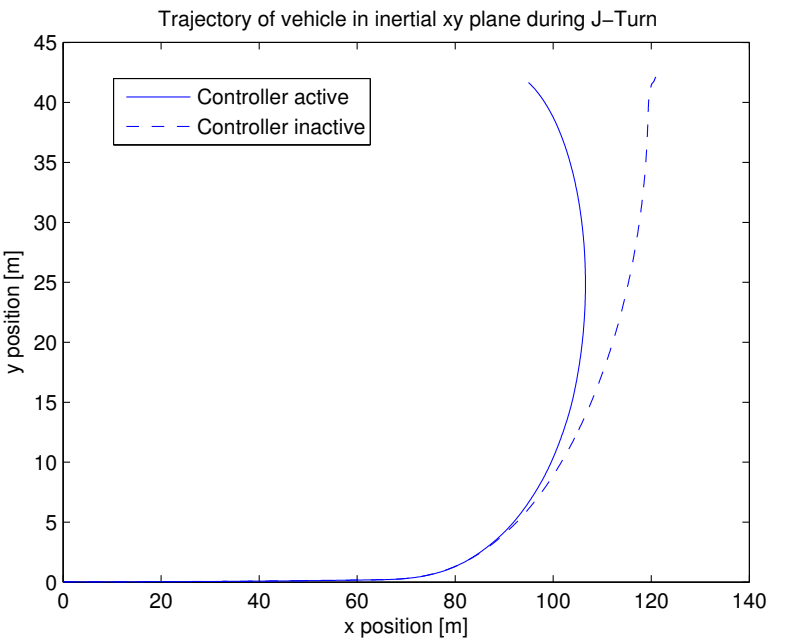

Figure 8.16 Vehicle trajectory in the inertial xy plane during the J-turn maneuver. The solid line corresponds to the trajectory of the vehicle when the controller is active. The controller succeeds in reducing the radius of curvature obtained by the maneuver, as well as stabilizing the yaw dynamics and sideslip angle. This could be extremely important, in the cases when the maneuver is performed in order to avoid an obstacle.

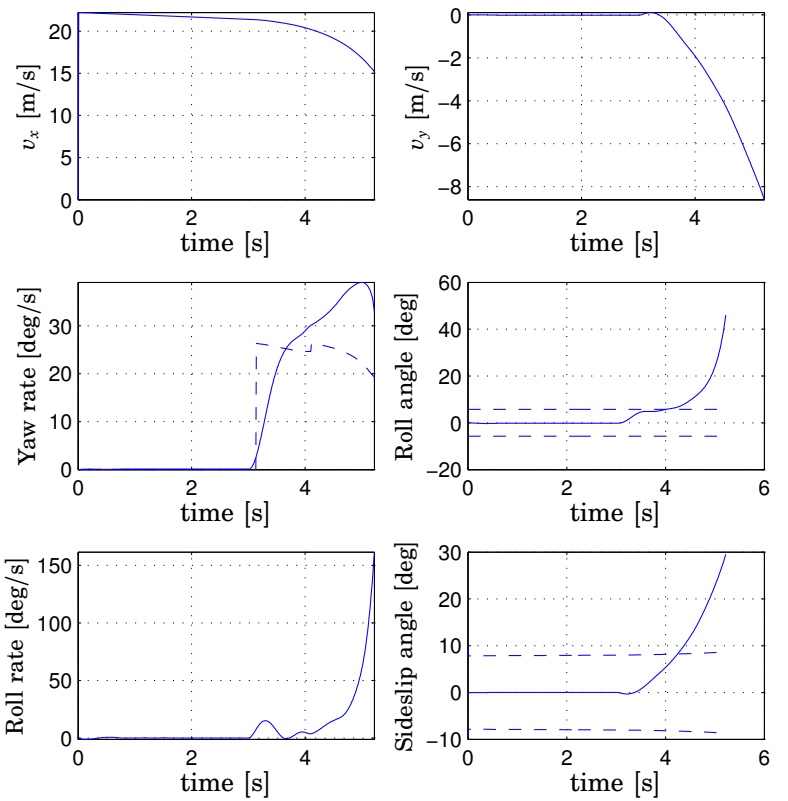

Figure 8.17 Vehicle states for the J-turn maneuver with additional load with the controller inactive. Rollover occurs in this case due to the presence of the additional load.

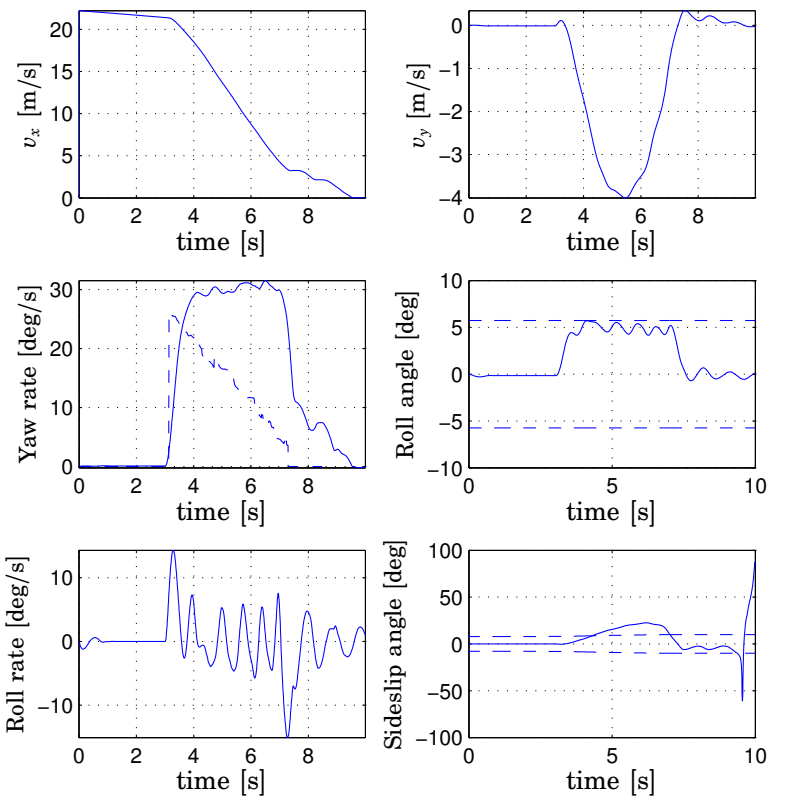

Figure 8.18 Vehicle states for the J-turn maneuver with additional load with the controller active. Rollover is prevented, and the roll angle remains within the prescribed limits. The sideslip angle is stabilized, although it does not remain within the given limits. The behaviour of the sideslip angle in the final second of the simulation is caused by the vehicle coming to rest, and thus may be ignored.

8.3 Real Time Performance

Although the simulations carried out in this chapter are very useful for examining the behaviour of the control system, they ignore real time issues, which are important in this application, particularly since the algorithms are rather computationally intensive.

To examine the real time performance of the algorithms, they were tested in open loop on a dSPACE Autobox, running at 1GHz. Example data from simulations was used. The average computation time with this experimental setup was found to be of the order of 0.2ms, considerably less than the sample time of 10ms. Of course, the hardware used is considerably more powerful than production hardware, but it is clear that the computational demand is not unreasonable.

8.4 Discussion

The simulation results presented in this chapter indicate that the proposed algorithms perform well in a number of situations. In this section the peculiarities, merits and drawbacks of the proposed algorithms are discussed.

Rollover Detection and Switching

The proposed PD-based switching method using the lateral acceleration measurement was found to work well. The early detection of an oncoming rollover event is vital to the performance of the overall control system. It was observed that even very small delays in the activation of the controller lead to degraded performance. In particular, the desired values of both the braking force F_{xT} and the total moment M_T can not be met simultaneously. The explanation for this can be found by considering Figure 4.9. As the roll angle, and thus the load transfer, increases, the forces that may be generated by the tires on the inside of the turn decrease. This limits the range of values that may be obtained simultaneously for F_{xT} and M_T. The control allocator attempts to minimize the allocation error for these virtual controls, according to their respective weights, but if the load transfer becomes too large it may be impossible to achieve both desired values at the same time.

It is possible that the PD-based switching strategy may encounter problems in a real implementation, due to noise on the lateral acceleration signal. If this were to be the case, it is possible that a more complex switching algorithm, taking into account other variables, may be required. Some possibilities for such an algorithm were discussed in Chapter 4.

Control Strategy

The strategy based on using a threshold value of the lateral acceleration for switching and a P controller for the total braking force, as described in (5.7), is presented in [Schofield *et al.*, 2006]. It was successful in preventing rollover, although oscillations were observed in the control signal. The modified strategy in (5.9) using a constant braking command coupled with the PD detection scheme exhibited better performance. It is conceivable that a more advanced control methodology could be applied to the problem in order to increase the performance. However, more complex strategies are likely to be more dependent on models, and therefore vehicle parameters. Since a reliable scheme for obtaining the necessary parameters has not yet been developed, the design of more complex strategies may be unnecessary.

The yaw rate controller is rather more complex than the roll controller. In simulations it has performed very consistently, the main problem being the coupling with the total longitudinal force F_{xT}. The desired value of F_{xT} is used for decoupling, which implies that the yaw controller may work poorly in cases when the allocation error is large, that is, when the desired virtual controls cannot be achieved.

An interesting observation is that in the simulated maneuvers, the yaw rate is virtually always larger than the reference. This may be due to modeling errors, although the model validation performed in Chapter 3 indicated that the two-track model captured the yaw dynamics of the simulation model rather well. A more likely explanation is that control allocation errors are to blame. It can be seen that the relative allocation errors for the yaw rate are much larger than for the longitudinal force, giving errors in the total moment and thus the yaw rate. Since the primary task of the controller is rollover prevention, the weightings on the longitudinal force in the control allocator are higher than those on the total moment.

Control Allocation

Tire Force Approximation The proposed control allocation strategy, based on the use of linear approximations of the tire force characteristics, worked remarkably well. The approximation, given by:

$$\nu F_y = (\sigma \mu F_z + F_x) \operatorname{sign} \delta$$

originates from the assumption that tire forces are permanently saturated when the controller is active. This implies that the maximum achievable lateral forces depend only on the friction coefficient and the normal force, and not on the slip angles. This is useful since the slip angles are not measurable. Although this constitutes a major assumption, it was found

that this assumption appeared to be valid. Figures 8.5, 8.6 and 8.7 show that the values of F_{xT}, F_{yT} and M_T that were obtained closely match the values predicted by the model used in the control allocator. In fact, the tuning parameters ν and σ were not required; the algorithm worked very well with both parameters set to one. This is of course related to the fact that the friction coefficient was assumed to be known.

In reality, the friction coefficient may not be known. This represents the primary uncertainty in the system. In Chapter 10, a method of friction estimation is presented, the results of which may be used by the rollover mitigation controller to improve performance.

Summary

In this chapter simulation results have been presented which confirm the operation of the proposed vehicle dynamics control algorithm. Standard test maneuvers relevant for rollover prevention have been used as test cases. Although simulation by itself cannot conclusively demonstrate the effectiveness of the algorithms, the advanced nature of the simulator used ensures a high level of realism, and negates many issues such as unmodeled dynamics. Nevertheless, it is important to be conscious of the fact that problems such as parameter uncertainties may present themselves in real-world experiments. In particular, knowledge of the friction coefficient is likely to be critical to the performance of the controller. In Chapter 10, a friction estimation method will be developed.

9

Sensor Fusion for Yaw Rate Measurements

9.1 Introduction

Accurate information is a prerequisite for optimal operation of both control and estimation algorithms. While cost constraints typically restrict the quantity of sensors installed in automotive vehicles, there is nevertheless a considerable amount of information available. Information about a given signal is generally not constrained to reside solely in the sensor signal associated with it, but rather may be present in a variety of sensor signals. This implies that there is scope for improving the quality of a given measured signal by using other measurements.

In practice, of course, measurements always contain errors. Measurement noise is always present, but signals may also be corrupted by offsets, drift and gain errors. Appropriate signal processing can be used to reduce the effects of these errors, but caution should be exercised to avoid excessive loss of data.

Knowledge of the yaw rate $\dot{\psi}$ is important for numerous VDC systems, including the rollover mitigation controller presented in this thesis, as well as the friction estimation approach described in the following chapter. Throughout the thesis it is assumed that the vehicle sensor package includes a gyroscopic yaw rate sensor. The sensor output is typically subject to offset and gain errors.

This chapter describes a sensor fusion approach to improving the measured yaw rate signal. A model-based approach is taken, using a bicycle model which takes the steering angle as input and varies with longitudinal velocity. Wheel angular velocities, available from ABS sensors, are used as additional information. An offset error in the yaw rate measure-

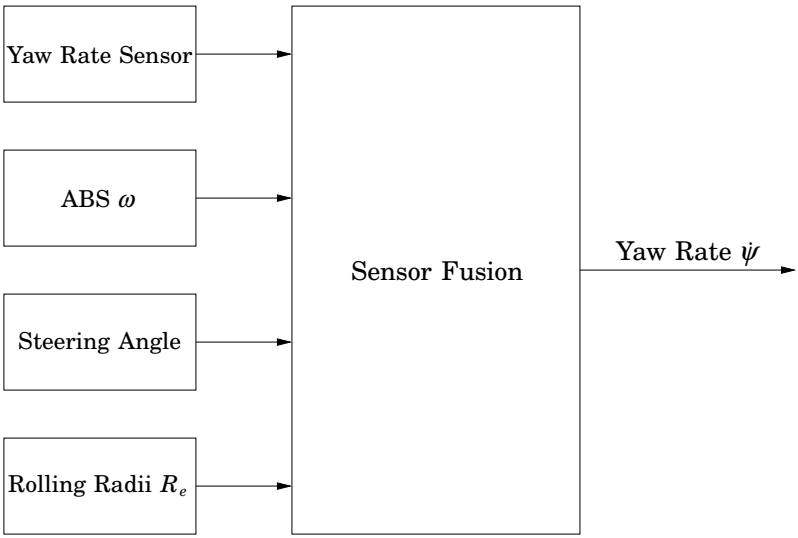

Figure 9.1 Sensor fusion diagram for yaw rate

ment is modelled. An offset error in the steering angle measurement is also included. Yaw rate information is also contained within the lateral acceleration measurement, but this is affected by road bank angle. For this reason the lateral acceleration measurement is not included in the sensor fusion model. The lateral acceleration could be included, if appropriate bank angle correction was available. A similar sensor fusion approach is taken in [Gustafsson *et al.*, 2001], although the bicycle model and corresponding steering information is not used. Instead, the yaw dynamics are assumed to be driven by a random disturbance. The combination of wheel angular velocity measurements with acceleration sensor signals may also be used in vehicle velocity observers. This approach is taken in [Imsland *et al.*, 2005; Imsland *et al.*, 2006], where nonlinear observers are designed.

Figure 9.1 illustrates the signals involved in the approach described in this chapter. The sensor fusion algorithm uses the yaw rate gyro sensor signal $\dot{\psi}_s$, the rear wheel angular velocities ω^{rl} and ω^{rr} from the ABS sensors, the steering angle δ and the estimated effective rolling radii R_e for the rear wheels. The estimation of the rolling radii will be described in more detail in the following chapter.

Problem Formulation

The yaw rate measurement signal provided by the gyro sensor is typically

subject to an offset:

$$\dot{\psi}_s = \dot{\psi} + \Delta^o_{yaw}$$

where $\dot{\psi}_s$ is the gyro sensor signal, $\dot{\psi}$ the true yaw rate and Δ^o the (not necessarily constant) offset. Similarly, the steering angle sensor signal may also be subject to an offset:

$$\delta_s = \delta + \Delta^o_{steer}$$

For convenience, δ refers to the steering angle of the front wheels, rather than the steering column. Scale factor errors may also be present, but are not considered further here. By using information such as the steering angle and the angular velocities of the wheels, it is intended to improve the quality of the yaw rate sensor signal. This is to be accomplished using a model-based approach, in which sensor offsets are included in the model.

9.2 Model

To relate the steering angle to the yaw rate, a single-track (bicycle) model can be used. The bicycle model was introduced in Chapter 3. For a constant longitudinal velocity v_x, the bicycle model is described by a linear system:

$$\begin{pmatrix} \ddot{\psi} \\ \dot{v}_y \end{pmatrix} = \underbrace{\begin{pmatrix} \dfrac{-a^2 C_F - b^2 C_R}{I_{zz} v_x} & \dfrac{b C_R - a C_F}{I_{zz} v_x} \\ \dfrac{b C_{R_\alpha} - a C_{F_\alpha}}{m v_x} - v_x & -\dfrac{C_F + C_R}{m v_x} \end{pmatrix}}_{A} \begin{pmatrix} \dot{\psi} \\ v_y \end{pmatrix} + \underbrace{\begin{pmatrix} \dfrac{a C_F}{I_{zz}} \\ \dfrac{C_F}{m} \end{pmatrix}}_{B} \delta \quad (9.1)$$

In general, the A matrix depends nonlinearly on the longitudinal velocity v_x. The bicycle model can then be described by a linear parameter-varying (LPV) system:

$$\dot{x} = A(v_x)x + B\delta$$

where $x = \begin{pmatrix} \dot{\psi} \\ v_y \end{pmatrix}$ In order to incorporate the sensor offsets into the model, the state vector may be extended:

$$x_e = \begin{pmatrix} \dot{\psi} \\ v_y \\ \Delta^o_{yaw} \\ \Delta^o_{steer} \end{pmatrix}$$

Assuming that the offsets have no dynamics, the following system is obtained:

$$\dot{x}_e = \begin{pmatrix} \frac{-a^2C_F - b^2C_R}{I_{zz}v_x} & \frac{bC_R - aC_F}{I_{zz}v_x} & 0 & -\frac{aC_F}{I_{zz}} \\ \frac{bC_{R_\alpha} - aC_{F_\alpha}}{mv_x} - v_x & -\frac{C_F + C_R}{mv_x} & 0 & -\frac{C_F}{m} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} x_e + \begin{pmatrix} \frac{aC_F}{I_{zz}} \\ \frac{C_F}{m} \\ 0 \\ 0 \end{pmatrix} \delta_s \tag{9.2}$$

Measurements

In addition to the yaw rate sensor signal, yaw rate information is contained within the wheel angular velocities. Considering the relative motion of the rear wheels upon cornering, and assuming no sideslip occurs, the following relation holds:

$$l\dot{\psi} = R_e^{rr}\omega^{rr} - R_e^{rl}\omega^{rl} \tag{9.3}$$

where R_e is the effective rolling radius of the respective wheels, and l is the rear track width. Although this information may contain scale factor errors, due to uncertainties in R_e, it is not expected to contain offset errors. The following observations can therefore be defined:

$$\begin{aligned} y_1 &= \dot{\psi}_s = \dot{\psi} + \Delta_{yaw}^o + n_1 \\ y_2 &= \dot{\psi} = \frac{1}{L}(R_e^{rl}\omega^{rl} - R_e^{rr}\omega^{rr}) + n_2 \end{aligned}$$

The observations are related to the state according to:

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} x_e + \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} \tag{9.4}$$

where n_1 and n_2 are assumed to be Gaussian noise. In practice, the variance of n_2 (the noise on the wheel angular velocity signal) is much greater that that of n_1 (the yaw rate sensor signal).

9.3 Estimation Methods for Sensor Fusion

The Kalman filter [Åström and Wittenmark, 1990] is a standard tool for sensor fusion. It is also possible to reformulate the problem to allow a

recursive least squares (RLS) solution [Åström and Wittenmark, 1995], but in this investigation, attention will be restricted to the Kalman filter and its derivatives.

The Kalman filter, more correctly referred to as the Stratonovich-Kalman-Bucy filter [Shakhtarin, 2006], is a recursive estimator for linear systems which minimizes the prediction error variance. For nonlinear systems, the Extended Kalman Filter (EKF) can be used, which provides a first-order approximation of the nonlinearities. However, the EKF can be computationally demanding, and may not always perform well, depending on the type of nonlinearities present.

Unscented Kalman Filter

The Unscented Kalman Filter (UKF) was proposed relatively recently in [Julier and Uhlmann, 1997] as a more efficient extension of the Kalman filter than the EKF. The underlying idea is that a Gaussian distribution is more easily approximated than an arbitrary nonlinear function.

Since the system model used for sensor fusion is nonlinear, the UKF is a natural choice of method. The operation of the UKF will be briefly outlined here, following the presentation in [Wan and Van Der Merwe, 2000].

The UKF treats general nonlinear systems of the form:

$$\begin{aligned} x_{k+1} &= f(x_k, v_k) \\ y_k &= h(x_k, n_k) \end{aligned} \tag{9.5}$$

where v_k and n_k are zero mean process and measurement noises respectively. The UKF is based on the *unscented transformation*, which is a means of propagating the distribution of a random variable undergoing a nonlinear transformation. This involves the sampling of the distribution at a specific set of points, known as *sigma points*, propagating these through the nonlinearity and reconstructing the transformed random variable.

For a random variable x with dimension L having a mean $\overline{x}$ and covariance P_x, the sigma vectors χ_i and their corresponding weights W_i are:

$$\begin{aligned} \chi_0 &= \overline{x} \\ \chi_i &= \overline{x} + \left(\sqrt{(L+\lambda)P_x}\right)_i, \quad i = 1, \ldots, L \\ \chi_i &= \overline{x} - \left(\sqrt{(L+\lambda)P_x}\right)_{i-L}, \quad i. = L+1, \ldots, 2L \\ W_0^m &= \lambda/(L+\lambda) \\ W_0^c &= \lambda/(L+\lambda) + (1-\alpha^2+\beta) \\ W_i^m &= W_i^c = \frac{1}{2(L+\lambda)} \end{aligned} \tag{9.6}$$

where $\lambda = \alpha^2(L+\kappa) - L$ is a scaling parameter, α is a parameter determining the spread of sigma points, κ is a secondary scaling parameter, β is a parameter used for incorporation of information about the distribution, and W_i^m and W_i^c are the weights on the mean and covariance respectively.

For the state estimation problem, the random variable is the state of the system, as well as the noise vectors:

$$x_k^a = \begin{pmatrix} x_k \\ v_k \\ n_k \end{pmatrix}$$

The algorithm is initialized with:

$$\begin{aligned}
\hat{x}_0 &= E\{x_0\} \\
P_0 &= E\{(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T\} \\
\hat{x}_0^a &= E\{x^a\} = \begin{pmatrix} \hat{x}_0 \\ 0 \\ 0 \end{pmatrix} \\
P_0^a &= E\{(x_0^a - \hat{x}_0^a)(x_0^a - \hat{x}_0^a)^T\} \\
&= \begin{pmatrix} P_0 & 0 & 0 \\ 0 & P_v & 0 \\ 0 & 0 & P_n \end{pmatrix}
\end{aligned} \tag{9.7}$$

At each sample, the following calculations are performed. The sigma points are calculated:

$$\chi_{k-1}^a = (\,\hat{x}_0^a \quad \hat{x}_0^a \pm \sqrt{(L+\lambda)P_{k-1}^a}\,) \tag{9.8}$$

The sigma points are then propagated through the dynamics:

$$\begin{aligned}
\chi_{k|k-1}^x &= f(\chi_{k-1}^x, \chi_{k-1}^v) \\
\hat{x}_k^- &= \sum_{i=0}^{2L} W_i^m \chi_{i,k|k-1}^x \\
P_k^- &= \sum_{i=0}^{2L} W_i^c (\chi_{i,k|k-1}^x - \hat{x}_k^-)(\chi_{i,k|k-1}^x - \hat{x}_k^-)^T \\
\mathcal{Y}_{k|k-1} &= h(\chi_{i,k|k-1}^x, \chi_{i,k-1}^n) \\
\hat{y}_k^- &= \sum_{i=0}^{2L} W_i^m \mathcal{Y}_{i,k|k-1}
\end{aligned} \tag{9.9}$$

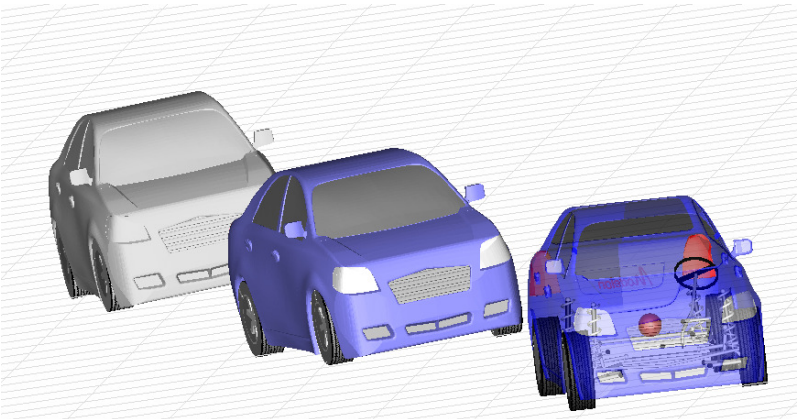

Figure 9.2 Simulation of a Volvo S40 using VehicleDynamics Library [Modelon, 2008]. Simulation results using known sensor offsets were used to validate the operation of the sensor fusion algorithm before testing on measured data, in which the sensor offsets are unknown.

The update is then performed as:

$$
\begin{aligned}
P_{\hat{y}_k\hat{y}_k} &= \sum_{i=0}^{2L} W_i^c (\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-)(\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-)^T \\
P_{\hat{x}_k\hat{y}_k} &= \sum_{i=0}^{2L} W_i^c (\chi_{i,k|k-1} - \hat{x}_k^-)(\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-)^T \\
\mathcal{K} &= P_{\hat{x}_k\hat{y}_k} P_{\hat{y}_k\hat{y}_k}^{-1} \\
\hat{x}_k &= \hat{x}_k^- + \mathcal{K}(\hat{y}_k - \hat{y}_k^-) \\
P_k &= P_k^- - \mathcal{K} P_{\hat{y}_k\hat{y}_k} \mathcal{K}^T
\end{aligned}
\tag{9.10}
$$

Note that an external input u is not included in the above description. This may be added to the time update equations (9.9) without difficulty.

A UKF based on the model (9.2) and equations (9.7) to (9.10) was implemented using the ReBEL estimation toolbox for Matlab [ReBEL, 2008].

Validation Using Simulation Data Testing the resulting observer directly using real-world measurement data is difficult due to the fact that the actual offsets, in addition to the true yaw rate, are generally not known. In order to validate the operation of the observer, data from vehicle simulation software was used. The VehicleDynamics Library, developed by Modelon AB [Modelon, 2008] for use with the simulation tool Dymola,

contains high-fidelity vehicle models implemented in the object-oriented, equation-based modeling language Modelica [Fritzson, 2003]. A model of a Volvo S40 was used for the validation, illustrated in Figure 9.2. Known offsets were added to the steering angle and yaw rate measurements. Exact rolling radius and velocity values were used, which in combination with the lack of measurement noise allows the validity of the sensor fusion model to be easily evaluated, in a manner not possible with measured data from an actual vehicle.

Figures 9.3 and 9.4 show the results of using a UKF to estimate the states of the sensor fusion model. The simulated driving situation is free rolling from an initial velocity of 25ms^{-1}, with a small constant steering angle. Offsets of 0.01 were added to the yaw rate and steering angle measurements.

The yaw rate estimate, shown in Figure 9.3, is clearly better than the measured signal, corrupted by the offset. Figure 9.4 shows the estimates of the offsets. The yaw rate offset estimate is very close to the true value, while the steering offset estimate has a small bias. This is most likely due to the fact that the steering angle is propagated through the bicycle model before it appears in the output. Parameter errors in the bicycle model will therefore affect this estimate.

Testing With Vehicle Data The results from the simulation data tests would appear to confirm the operation of the sensor fusion algorithm. In light of this, the algorithm was then tested using real vehicle data. The data used was from a Volvo S40, collected in Arjeplog, Sweden during the Road Friction Estimation (RFE) project [Vägverket, 2008b]. More details about this project and data collection will be forthcoming in the following chapter.

Figures 9.5 and 9.6 show the results of using the UKF with the sensor fusion model. Since the correct yaw rate is not known, it is difficult to quantify the degree of improvement. Figure 9.6 indicates the presence of a constant steering angle offset. A yaw rate sensor offset also appears to be present, although the estimate does not converge to a constant value. This may be due to the higher level of yaw excitation towards the end of the measurement sequence. The sensor fusion model is primarily valid for small steering angles, and the relation between wheel angular velocities and yaw rate assumes zero wheel slip (both lateral and longitudinal). The variation of the yaw rate estimate may therefore be due to outside of the region of model validity. It would be relatively straightforward to include some form of conditional updating, where estimates are only updated under certain excitation criteria, in the sensor fusion algorithm. Such a criterion is already included in the algorithm to switch off updating for very low velocities (to prevent numerical problems in the model).

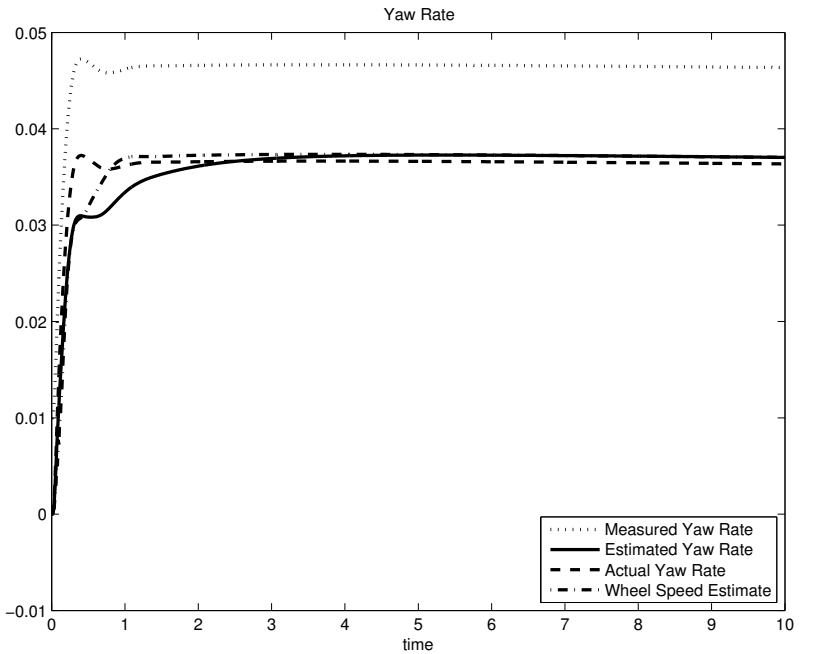

Figure 9.3 Yaw rate estimate using simulation data from VehicleDynamics Library, with known offsets. The estimate is obtained using a UKF.

Another possible explanation for variation in the yaw rate offset estimate is that it is common for suppliers to implement offset estimation algorithms within the sensors. This is typically done using rule-based methods, such as examining the sensor output when the vehicle is at rest. Depending on the algorithm used, it is possible that the sensor occasionally updates its internal offset estimate (which may or may not be accurate) leading to a variation in the observed offset in the sensor output.

9.4 Linear Parameter Varying Observer

The Unscented Kalman Filter used above has a higher degree of computational complexity than a standard Kalman filter. As mentioned previously, the sensor fusion model is linear for fixed longitudinal velocity. However, the use of a linear Kalman filter would sacrifice accuracy in situations where the velocity deviates significantly from the nominal velocity used

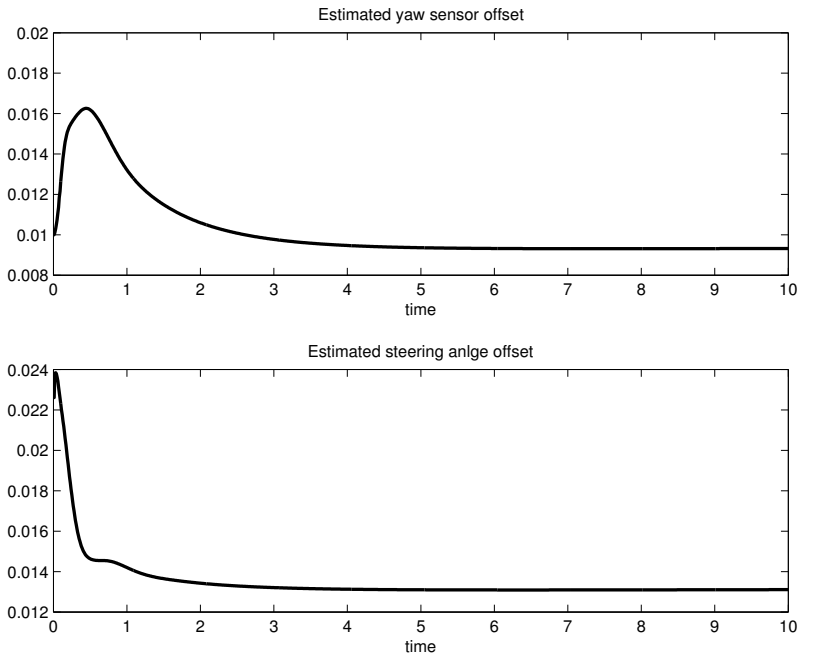

Figure 9.4 Yaw rate sensor offset and steering angle offset estimates using simulation data from VehicleDynamics Library. The offsets used were 0.01 for both the yaw rate and the steering angle. The estimates were obtained using a UKF.

for design. On the other hand, the UKF may be unnecessarily complex for this application.

It was decided to test the use of a fixed Kalman gain, calculated for some fixed nominal velocity v_x^{nom}, in combination with the LPV sensor fusion model. This may be referred to as a linear parameter varying observer. The term 'observer' is more appropriate than 'Kalman filter' here since the estimates are not optimal for $v_x \neq v_x^{nom}$. The resulting observer is described by the following equation:

$$\dot{x}_e = A_e(v_x)x_e + B_e\delta_s + K^{nom}(y - Cx_e) \tag{9.11}$$

where $A_e(v_x)$ and B_e are the system matrices of the sensor fusion model (9.2), and K^{nom} is the fixed Kalman gain corresponding to the nominal velocity v_x^{nom}.

The stability of the LPV observer (9.11) cannot be guaranteed since it is a time varying system. An argument can be made that since v_x is slowly time varying, and the poles of $A_e(v_x) - K^{nom}C$ all lie in the left

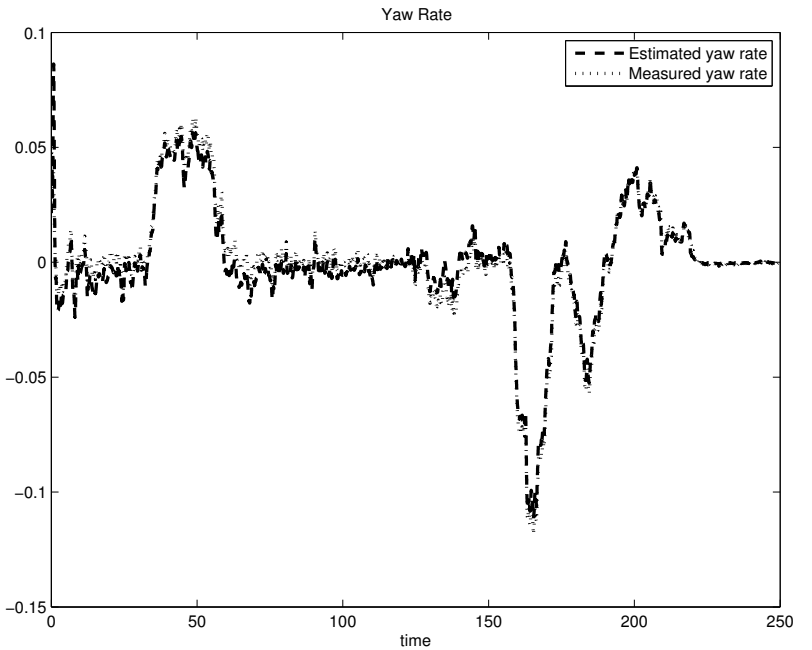

Figure 9.5 Yaw rate estimate with UKF using measurement data from a Volvo S40

half plane for relevant values of v_x, the system should be stable. Another alternative, if stability guarantees are required, is the approach taken in [Darengosse *et al.*, 2000], where a quadratically-stable LPV flux observer for an induction motor is designed.

Figures 9.7 and 9.8 show the results of using the LPV observer with the data used above from the Volvo S40. The performance is comparable with the UKF, although the behaviour of the two algorithms differs slightly. In particular, the estimated yaw rate sensor offsets are rather different. On the other hand, the steering angle offset estimates agree quite well.

9.5 Conclusions

In this chapter a model-based sensor fusion method is presented for improving the yaw rate measurement. By using a bicycle model, steering angle information can be incorporated with rear wheel angular velocity measurements to provide additional information about the yaw rate. In particular, sensor offsets in both the yaw rate signal and steering angle

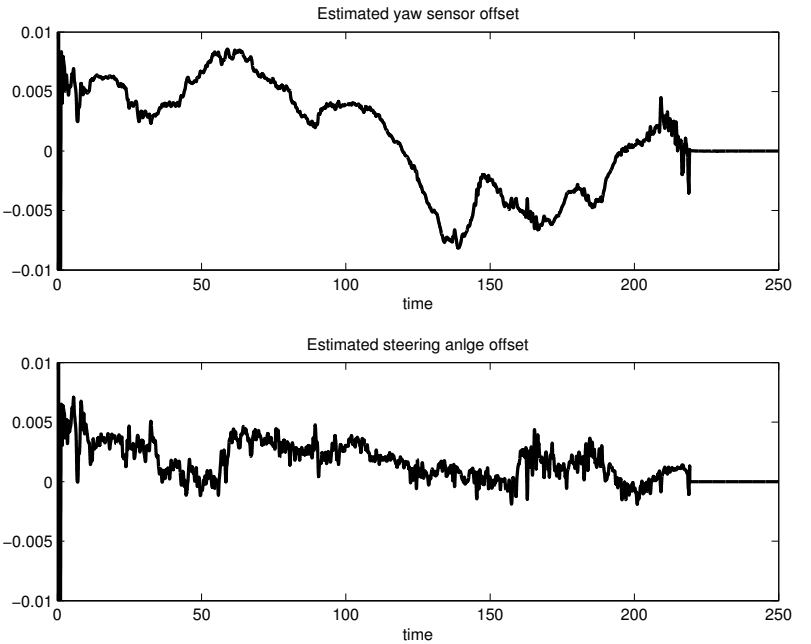

Figure 9.6 Yaw rate sensor offset and steering angle offset estimates with UKF using measurement data from a Volvo S40

signal can be taken into account.

An Unscented Kalman Filter was used to test the concept. Despite being considerably less computationally demanding than an Extended Kalman Filter, the UKF is nevertheless a complex algorithm. An observer based on a Kalman gain obtained for a nominal longitudinal velocity and using the LPV dynamics of the bicycle model was implemented and tested successfully.

An important point to note is that there is an assumption that the effective rolling radii of the undriven wheels are known. More specifically, the relative sizes are of primary importance, since the relation in (9.3) will be incorrect if the relative sizes of the rolling radii are not correct. The absolute values are of less importance, since these determine the value of the estimated yaw rate while cornering, a situation where other modeling errors also occur.

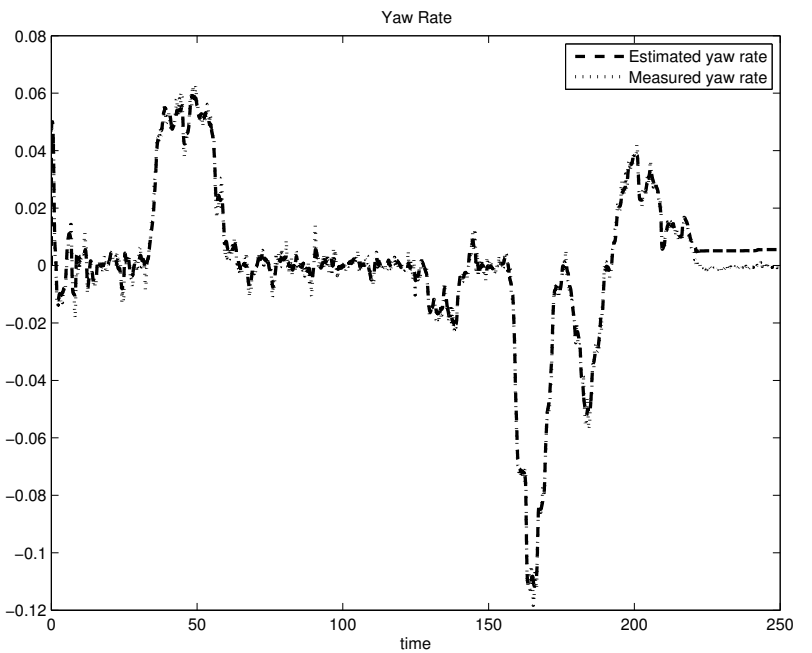

Figure 9.7 Yaw rate estimate with the LPV observer using measurement data from a Volvo S40

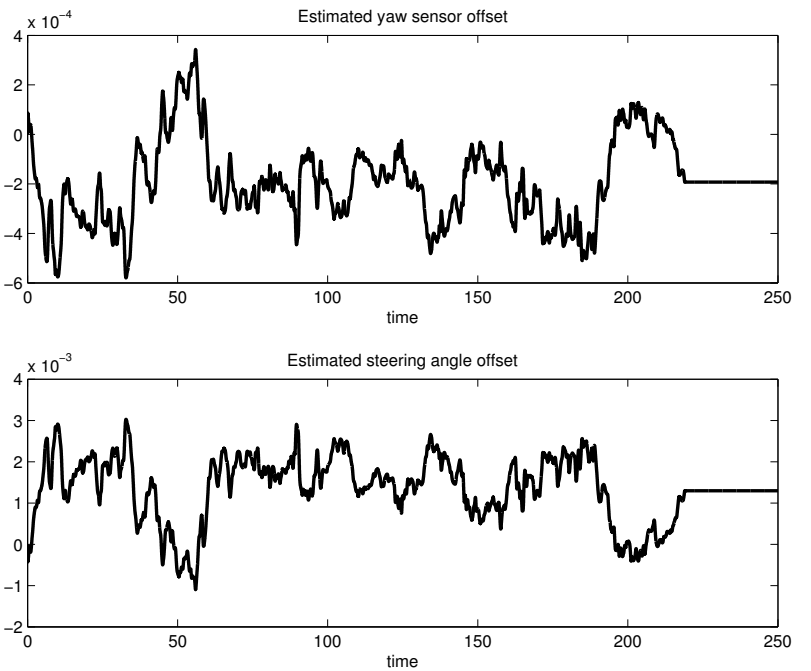

Figure 9.8 Yaw rate sensor offset and steering angle offset estimates with the LPV observer using measurement data from a Volvo S40

10

Model-Based Friction Estimation

The forces generated at the point of contact between the tire and the road are highly dependent on the properties of both the road surface and the tire. The key property for a given combination of road surface and tire is the coefficient of friction μ between the two. Although the interaction between tire and road is complex, the maximum available force is determined by the friction coefficient μ and the normal force on the wheel.

Vehicle dynamics control systems, such as the rollover mitigation controller in this thesis, are often concerned with the maximum force available. As such, the coefficient of friction may be regarded as the most important tire-road interaction parameter in such systems. The ability to reliably estimate μ is therefore important for maximizing the performance of vehicle control systems. In addition, it may be useful to present the driver with friction information. Numerous VDC systems are intended to correct mistakes made by the driver. Some of these mistakes may be avoided by providing the driver with appropriate information about the current road conditions.

Determination of μ is straightforward in cases where tire forces are saturated, such as under hard braking. The difficulty lies in obtaining a friction estimate under more normal driving circumstances, in which the tire slip is smaller. In these cases a model-based approach can be advantageous. By fitting data to a model of the tire characteristic, the model parameters, including μ, may be estimated. This approach may allow the estimation of friction without requiring tire force saturation.

Critical to the success of a model-based approach is the choice of model structure. As observed in Chapter 2 a large number of tire models exists. Since parameter estimation is to be performed, it is desirable to choose a model with a small number of parameters. The brush model is well suited to these requirements, containing only stiffness and friction parameters.

A key reason why the brush model is useful for friction estimation at lower slip is its description of the tire mechanics in the contact patch, which provides a relation between the force generated by the tire and the stiffness and friction parameters. Although the influence of the friction on the generated force is small for very low slips, the physical relationship described by the brush model allows identification of the friction from moderate slip values.

Previous Work

The importance of friction estimation is reflected by the considerable amount of work that has been done in the area. Industrial results are however seldom published. There is a great deal of variation in the approaches taken for friction estimation. A number of methods are based on the use of specific sensors, often with the aim of identifying the type of road surface rather than providing a numerical friction value. The classification of the road surface into discrete classes, such as asphalt, snow, ice or gravel, is often used as an alternative to computing a numerical estimate. The advantages are that such algorithms are typically simpler, and that information from different types of signals may be easily integrated. For example, the variance of the wheel speed measurements may be used as an indication of the coarseness of the road surface. A very low variance could indicate an icy road, while a very high variance could indicate a gravel road. Such an approach, using only existing sensors, is taken in [Gustafsson, 1997].

A disadvantage of classification methods is that there is likely to be considerable variation in friction values within each of the given surface classes. Simple classifications such as 'snow' and 'ice' disguise the fact that there are a myriad of possible conditions of such surfaces. The friction properties of these different conditions may be presumed to vary widely. For this reason, approaches based on assigning a nominal friction value based on a surface classification must be viewed with caution.

A commonly used approach is to estimate the tire stiffness, based on estimates of force and slip. The stiffness estimate is then used to indicate a corresponding friction level, possibly via some form of road surface classification. Such an approach is taken in [Lee *et al.*, 2004] and [Gustafsson, 1997]. However, while there is clearly correlation between the tire stiffness and friction levels, such approaches should be viewed as providing only an indication of the friction.

A number of sensor arrangements have been suggested for friction estimation. These include microphones which record the sound made by the tires, strain sensors within the tire to determine the deflection, and optical sensors used to determine the road surface type. While a number of approaches are promising, the introduction of a new sensor into a pro-

duction vehicle is always difficult due to cost reasons, in particular if its range of possible applications is small.

While many slip-based estimation methods are intended for use in the presence of longitudinal slip, methods based on lateral excitation also exist. In [Pasterkamp and Pacejka, 1997], a neural network is used to determine friction by considering the relation between lateral force and self-aligning torque, based on the brush model.

Applications Utilizing Friction Estimates

There are many potential uses of the friction estimate. As stated previously, virtually all VDC applications would benefit from knowledge of the friction. Friction information could be used for deciding which active safety function to activate. For example, a sudden large steering input in a low friction situation is likely to lead to a skid, whereas the same steering input in a high friction scenario may lead to a rollover (depending on the vehicle type).

Knowledge of the shape of the friction characteristic, possibly obtained through estimation of both friction and tire stiffness using the brush model, provides information on the slip values for which maximum friction is obtained. This information could be used to generate slip references for slip controllers, for example in ABS. Tire stiffness estimates can also be used in vehicle models, such as the bicycle model, which are often used for reference generation and estimation.

Friction estimates may also be sent from the vehicle to some form of central database. In this way individual vehicles act as sensors, providing information about friction at specific points in the road network. Such systems are investigated within the *Slippery Road Information Systems* (SRIS) project [Vägverket, 2008c].

10.1 Friction Estimation Using The Brush Model

From the discussions above it is clear that some method of estimating both the tire stiffness and friction, using existing sensor information and with a minimum of assumptions, would be desirable. A means of achieving this is provided by the brush model, presented in Chapter 2, which represents a physically derived relation between tire slip and force, depending on the stiffness and friction parameters. This method of friction estimation has been studied in [Svendenius, 2007a].

The principle of the method is to identify the friction and stiffness parameters of the static force-slip relationship described by the brush model. Clearly, there will be some requirement on the excitation provided by the

force and slip data in order to compute accurate estimates of the parameters. Other practical considerations also exist. One such consideration is that a vehicle reference velocity is required in order to accurately compute the tire slip. In [Svendenius, 2007a] this reference is provided by the rotational velocities of the undriven wheels. Obviously, this method cannot be used during braking, or for vehicles with drive on all axles, since slip occurs on all wheels in these cases. For this reason, friction estimation using this approach has thus far only been applied to the cases of acceleration or driving at constant velocity. This has been successfully tested on passenger cars in [Svendenius, 2007a].

Brush Model Validation

In order to use the brush model as a basis for friction estimation, it is desirable to validate the model. Extensive validation of the brush model, based on measurement data collected at Colmis proving ground in Arjeplog, Sweden [Colmis, 2008] and Volvo's test track at Hällared, Sweden, using VTI's purpose-built test vehicle BV12 (Figure 10.1), is presented in [Svendenius, 2007b] and [Svendenius, 2007c]. The brush model was found to agree well with the experimental data.

Model Structure

The brush model, derived in Chapter 2, is given by the following relation, in the case of pure longitudinal slip:

$$F_x = -C_x\sigma_x + \frac{C_x^2\sigma_x|\sigma_x|}{3\mu F_z} - \frac{C_x^3\sigma_x^3}{27\mu^2F_z^2} \tag{10.1}$$

The longitudinal force may be normalized with the normal force, giving $f_x = F_x/F_z$. The normalized brush model is then:

$$f_x = -C_{0x}\sigma_x + \frac{C_{0x}^2\sigma_x|\sigma_x|}{3\mu} - \frac{C_{0x}^3\sigma_x^3}{27\mu^2} \tag{10.2}$$

where C_{0x} is the normalized stiffness.

This model is valid up to the cutoff slip σ_x^0, after which the force is assumed to saturate. This is represented by:

$$f_x = \begin{cases} -C_{0x}\sigma_x + \dfrac{C_{0x}^2\sigma_x|\sigma_x|}{3\mu} - \dfrac{C_{0x}^3\sigma_x^3}{27\mu^2} & \text{if} \quad |\sigma_x| < \sigma_x^0 \\ -\mu \operatorname{sign}\sigma_x & \text{otherwise} \end{cases} \tag{10.3}$$

The estimation problem is to obtain the normalized stiffness C_{0x} and friction μ given measurements of the slip σ_x and normalized force f_x.

Figure 10.1 VTI's BV12 measurement vehicle at Colmis proving ground, Arjeplog, Sweden. This vehicle has an additional wheel for which the normal force can be set, and sensors to determine the resulting tire forces, allowing the friction between the tire and the road surface to be measured. Such measurements were used to validate the brush model.

Determination of Force and Slip

In order to estimate the friction parameters, information about the tire force and slip is required. Neither quantity is directly measurable, but must be computed from other information.

Driving Force The total driving force acting on the vehicle may be determined from the estimated crankshaft torque. Losses in the transmission, as well as torque required for the angular acceleration of the wheels, may be accounted for in a model-based manner.

Specifically, the normalized force f_x at the driven wheels is the quantity of interest. This is obtained as:

$$f_x = \frac{F_x^d}{F_z^d}$$

where F_x^d is the total driving force acting at the driven wheels and F_z^d is the corresponding normal force. The normal force varies during load transfer. This is accounted for using a simple model.

Slip The wheel slip is the difference between the circumferential velocity of the wheel and the velocity of the wheel hub relative to the ground.

The circumferential velocity may be obtained from the rotational velocity and the effective rolling radius of the wheel. The relative velocity between the wheel hub and the ground is more difficult to obtain. When travelling in a straight line, this corresponds to the longitudinal velocity of the vehicle. When cornering, corrections must be made to account for the rotational motion of the vehicle.

As discussed in Chapter 2, slip is a normalized quantity. The physical slip σ_x will be used in this chapter:

$$\sigma_x = \frac{v_x - \omega R_e}{\omega R_e}$$

The longitudinal velocity of the vehicle may be calculated in situations where some or all of the wheels are rolling freely. For two wheel drive vehicles, the undriven wheels (typically the rear wheels for front wheel driven passenger cars) may be used to calculate the longitudinal velocity during acceleration and normal driving. During braking, and for vehicles with all wheel drive systems, such an approach cannot be used since slip occurs on all wheels.

Accurate determination of the wheel slip is important for achieving accurate estimation results. The primary sources for error in the slip calculation are the longitudinal velocity and the rolling radii of the wheels. The measured angular velocities of the wheels are typically noisy, but appropriate filtering may be used to improve the signals, as discussed in detail in [Svendenius, 2007a]. An error in the effective rolling radius leads to an offset in the resulting slip-force characteristic (the characteristic will not pass through the origin). To counter this, the rolling radii are updated online based on the estimated load transfer and nominal tire compression parameters. It is also conceivable to estimate the error in the effective rolling radius in addition to the friction an stiffness parameters from the force and slip data, but this would complicate the estimation algorithm.

Additional effects have been found to introduce errors in the slip calculation. These include flexibility in the suspension during acceleration and tire relaxation length, which may be compensated for. The compensation may be performed using nominal parameters, or parameters obtained from experimental data.

Effective Rolling Radius Correction

Information on vehicle velocity is obtained via the angular velocity of the wheels. This requires that the effective rolling radii R_e of the wheels are known. If the velocity estimate is to be accurate, which is a necessity for accurate slip computation and therefore accurate friction estimation, the effective rolling radii must also be known accurately.

In practice, a reasonably accurate value of R_e is usually available, but this is valid only for static conditions. In dynamic conditions, load transfer causes R_e to vary. This variation must be accounted for.

The effective rolling radius for each wheel may be modeled as the nominal radius corrupted by two factor errors, the load-based variation ΔR and an additional error δR:

$$R_e = R_e^{nom}(1 + \Delta R + \delta R)$$

The load-based variation ΔR may be modeled as being linearly dependent on the load change:

$$\Delta R = \frac{k_l(F_{z,nom} - F_z)}{R_e^{nom}}$$

The additional error term δR encompasses other error sources, such as inaccuracy in the nominal rolling radius and errors in the normal force estimator. Since the undriven wheels are used for velocity reference calculation, and slip is normalized, it is the relative error between the rolling radii of the driven and undriven wheels that is of primary importance. It is possible to use an observer to update δR to reduce this error. When driving in a straight line, the products of the rolling radius and angular velocity of each wheel should be equal. In this situation an observer of the following form may be used:

$$\delta R^i(k+1) = \delta R^i(k) + K(R_e^{fl}(k)w^{fl}(k) - R_e^i(k)w^i(k))$$

where K is the observer gain, and the index $i \in [rl, rr]$. The front left wheel is used for comparison here, but an average based on both front wheels could also be used.

Individual Wheel Reference Speed Generation

In order to calculate the longitudinal slip for each wheel, the velocity of the wheel hub in the x direction must be known. If the vehicle velocity v_x at the centre of gravity of the vehicle is known, then the hub velocities may be calculated using the yaw rate and the vehicle geometry in the following way:

$$\begin{aligned} v^{fl} &= (v_x - l\dot{\psi})\cos\delta + (v_y + a\dot{\psi})\sin\delta \\ v^{fr} &= (v_x + l\dot{\psi})\cos\delta + (v_y + a\dot{\psi})\sin\delta \\ v^{rl} &= v_x - l\dot{\psi} \\ v^{rr} &= v_x + l\dot{\psi} \end{aligned} \tag{10.4}$$

where l is half the track width and a is the distance from the front axle to the centre of gravity. Assuming the yaw rate is available, for example using the algorithm described in Chapter 9, and that the geometric parameters are known, the key information is the vehicle velocity v_x.

Data Filtering Using Storage Bins

Although the estimation problem at hand, that of estimating constant parameters of a static relation, appears relatively simple, there are practical considerations which complicate the problem. Issues of excitation relating to recursive estimation of dynamical system parameters are well known. However, similar problems arise in this application. In the case of least squares estimation, identifiability is related to the invertability of the covariance matrix, which is in turn related to the information content of the measured signals in some sense. In the simplest case, this implies that data must be spread out sufficiently to allow the identification of a linear relationship. In the case of the nonlinear tire characteristic, data must be sufficiently spread out in the force-slip plane to allow correct identification of both parameters. Even if a block-based parameter identification approach is used, such as standard least squares, there may be problems if a large number of data points are concentrated in one area. This is a relatively common scenario, as a constant acceleration on a given road surface would only yield a set of force-slip points concentrated at one point, making identification of the parameters difficult.

To avoid these issues, the force-slip axes may be divided into regions, which may be denoted as storage bins. The division of the axes into such bins can be arbitrary. In each bin, a local average of force and slip data is generated. A nonlinear weighting function is used to reduce the effect of outliers, by requiring a certain number of measurements within the bin before computing an average. The weighting function also includes a saturation, used to prevent the problems described above related to data being restricted to a specific region during certain driving sequences.

The effect of using this approach is to gather information about the friction characteristic over time during a given driving sequence for which the estimation algorithm is active. As previously stated a key assumption is that the friction parameters are either constant during this period, or undergo an abrupt change (such as a change in surface type). The latter situation is to be dealt with separately, but will nevertheless use the information stored in the storage bins.

The storage bin approach is practically motivated, and lacks thorough statistical analysis. A discussion of the statistical proprieties is given in [Svendenius, 2007a], but it should be recalled that the quantities of data involved are generally small and that asymptotic properties are of minimal practical value in this application.

Parameter Optimization

When a suitably large number of data bins have been populated to provide information on the friction characteristic, parameter estimation can be performed. Since the brush model is nonlinear in the parameters, linear estimation techniques such as least squares cannot be applied directly. However, least squares may be used with a simplified model structure. This approach may be useful in cases where the range of slip values is small. When more information is available, nonlinear estimation using Gauss-Newton optimization may be performed.

Least Squares Estimation The brush model structure in (10.3) contains three terms, and is nonlinear in the parameters μ and C_{0x}. For small slip values, however, the third term may be neglected, and an appropriate choice of parameters allows the formulation of a least squares problem:

$$y(k) = \phi^T(k)\theta(k) + e(k) \tag{10.5}$$

where the observed output $y(k)$ is the normalized longitudinal force f_x, and $e(k)$ is Gaussian noise. The regressor vector is:

$$\phi(k) = \begin{pmatrix} -x(k) \\ x(k)|x(k)| \end{pmatrix} \tag{10.6}$$

where $x(k)$ is the slip value at time k. The parameter vector is:

$$\theta(k) = \begin{pmatrix} C_{0x} \\ \theta_2 \end{pmatrix} \tag{10.7}$$

The least squares estimate found by minimizing the cost function:

$$V(x, y, \theta) = \sum_{k=1}^{N} \frac{1}{2} w(k)(y(k) - \phi^T(k)\theta(k))^2$$

where $w(k)$ is a weight for each sample. The solution is given by:

$$\hat{\theta} = (\Phi^T W \Phi)^{-1} \Phi^T W \mathcal{Y}$$

where $\Phi = (\,\phi(1) \quad \dots \quad \phi(N-1)\,)^T, \mathcal{Y} = (\,y(2) \quad \dots \quad y(N)\,)$ and $W = Iw$.

With the regression model defined by (10.5), (10.6) and (10.7), an estimate of C_{ox} is obtained directly and an estimate of μ may be obtained as $\hat{\mu} = \hat{C}_{0x}^2/(3\hat{\theta}_2)$. As mentioned above, this method is only reliable when the third term of the brush model is very small, that is, in cases of low slip. At higher slips nonlinear estimation must be used in conjunction with the full brush model.

Gauss-Newton Optimization The problem of finding the least squares parameter estimates using the full brush model is a nonlinear optimization problem. Such problems can in general not be solved directly but must by solved using iteration.

Newton's method for solving such problems uses the Hessian to determine the step length of the parameter update:

$$\theta(k+1) = \theta(k) - H^{-1}\nabla V \tag{10.8}$$

where H is the Hessian of the cost function V. Since computation of the Hessian is time-consuming, the Gauss-Newton method approximates it as $H \approx 2J^T WJ$, where J is the Jacobian of the error vector e. The parameter update law is then given by:

$$\theta(k+1) = \theta(k) - (J^T WJ)^{-1} J^T We \tag{10.9}$$

The Jacobian is given by:

$$J = \begin{pmatrix} \dfrac{\partial e_1}{\partial C_{0x}} & \cdots & \dfrac{\partial e_N}{\partial C_{0x}} \\ \dfrac{\partial e_1}{\partial \mu} & \cdots & \dfrac{\partial e_N}{\partial \mu} \end{pmatrix}^T$$

The partial derivatives have the form:

$$\frac{\partial e}{\partial C_{0x}} = \begin{cases} -\sigma_x + \dfrac{2}{3}\dfrac{C_{0x}\sigma_x|\sigma_x|}{\mu} - \dfrac{1}{9}\dfrac{C_{0x}^2\sigma_x^3}{\mu^2} & \text{if } |\sigma_x| < \sigma_x^0 \\ 0 & \text{otherwise} \end{cases}$$

$$\frac{\partial e}{\partial \mu} = \begin{cases} -\dfrac{1}{3}\dfrac{C_{0x}^2\sigma_x|\sigma_x|}{\mu^2} + \dfrac{2}{27}\dfrac{C_{0x}^3\sigma_x^3}{\mu^3} & \text{if } |\sigma_x| < \sigma_x^0 \\ -\operatorname{sign}\sigma_x & \text{otherwise} \end{cases}$$

where $\sigma_x^0 = 3\mu/C_{0x}$ is the cutoff slip.

The Gauss-Newton method is clearly more computationally complex than linear least squares. In addition, the initial estimates must be sufficiently close to the optimum values in order to obtain convergence. However, since the parameters are physical quantities with relatively small ranges of reasonable values, good initial estimates are available. In order to deal with the computational complexity, only one iteration of the Gauss-Newton method is performed at each sample.

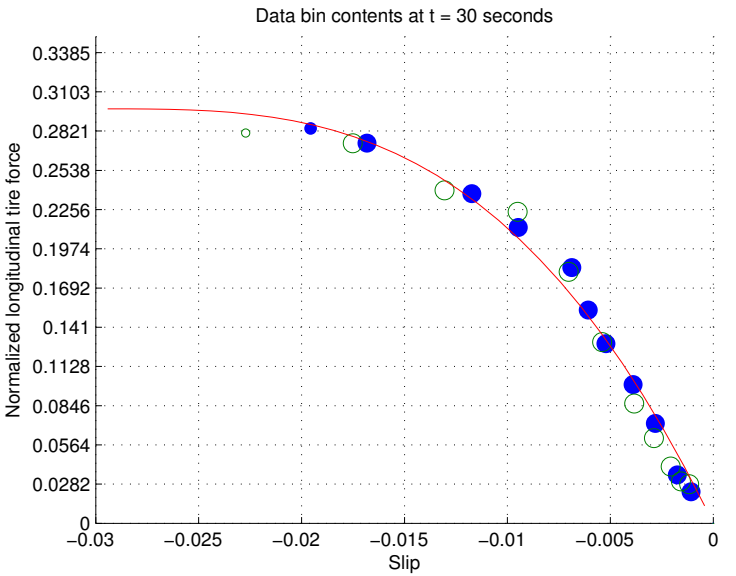

Figure 10.2 Storage bin contents for friction estimation on snow. The filled circles represent the force bin data points, and the hollow circles represent the slip bin data points. The solid line represents the brush model, using the latest parameter estimates. The grid divisions on the y axis correspond to the force bin intervals, which are uniformly spaced. The slip bin intervals are not uniformly spaced, and are not shown in the figure.

Estimation Results

Friction estimation using the above technique has been tested on extensive measurement data, collected primarily at Colmis proving ground in winter conditions, using a Volvo S40. An example of the results is shown in Figure 10.2, which illustrates the storage bin contents and resulting estimates after a gentle acceleration on snow. The solid line, representing the brush model corresponding to the estimates obtained after 30 seconds of driving, can be seen to fit the data points very well.

From the figure, it can be seen that several data points are close to maximum utilization (maximum force). Clearly, if sufficient data corresponding to maximum utilization is available, the friction level may be easily obtained. The advantage of the brush model is that the friction level may be reasonably accurately determined without the need for maximum utilization of the friction. Based on the available data, roughly 60-70% utilization is sufficient to provide reliable friction estimates.

10.2 Change Detection

The friction estimation method described above is intended for estimating constant parameters. The need to retain measurements throughout the force-slip plane, which motivated the use of the storage bin approach, stands in conflict with the need to 'forget' old measurement data in the event of changing parameters.

Nevertheless, the assumption of constant parameters is justified by the fact that the estimation algorithm is typically only active during certain sequences, when sufficient excitation is present. Gradual parameter changes under the course of such sequences may safely be neglected. However, it is possible that more drastic changes, such as a change in road surface type, may occur while the estimator is active. In such events it is desirable to identify that a change has occurred, and to obtain a new set of estimates as rapidly as possible.

Recursive estimation approaches based on exponential forgetting may be tuned to respond rapidly to abrupt changes in the parameters, but this is achieved by discarding all but the most recent data. As previously stated, such an approach is incompatible with the need to retain data throughout the force-slip plane, which is a requirement for accurate estimation of the parameters.

To illustrate the problems that arise with the storage bin approach in the event of sudden parameter changes, it is useful to investigate some specific cases. A change in road surface from snow to asphalt will be considered first. The data was gathered using a Volvo S40 at Colmis proving ground. Figure 10.3 shows the storage bin contents and resultant friction estimate after 36 seconds of driving on snow, performing a slow acceleration. The filled circles represent the force bin data points, and the hollow circles represent the slip bin points. The sizes of the circles are proportional to the number of data points in the corresponding bin. The solid line represents the brush model characteristic, using the current friction and stiffness estimates. It can be seen that the brush model fits the data very well. At 36 seconds, the road surface changes to asphalt. Without a means of performing change detection, the estimation algorithm continues to collect data points and generate estimates. Figure 10.4 shows the storage bin contents after 45 seconds of driving, 9 seconds after the change of road surface. The increase in tire stiffness and friction level can clearly be seen from the data collected after the surface change, but old data, collected while driving on snow, remains in the storage bins. The parameter estimates, which are now determined using data from both before and after the surface change, are therefore very poor. In fact, the use of saturation in the storage bins will prevent the estimates from converging to the correct values even if the vehicle continues to drive on the new

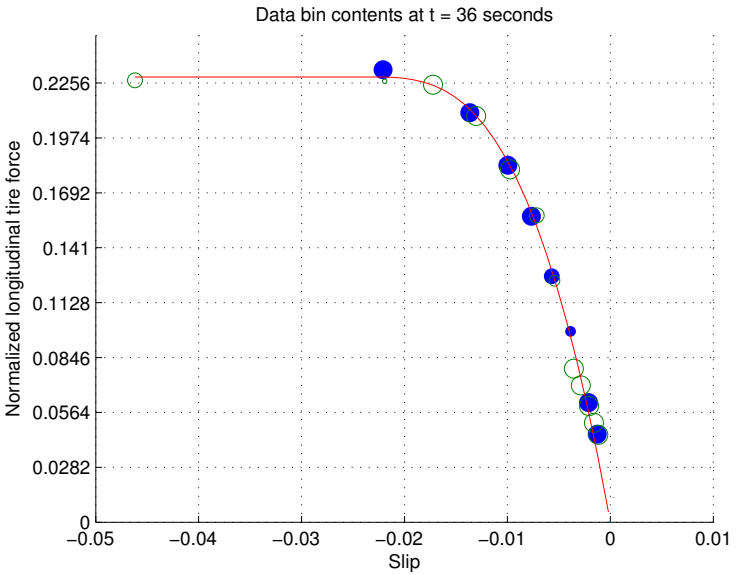

Figure 10.3 Storage bin contents after driving on snow. The filled and hollow circles are the force and slip bin data points respectively.

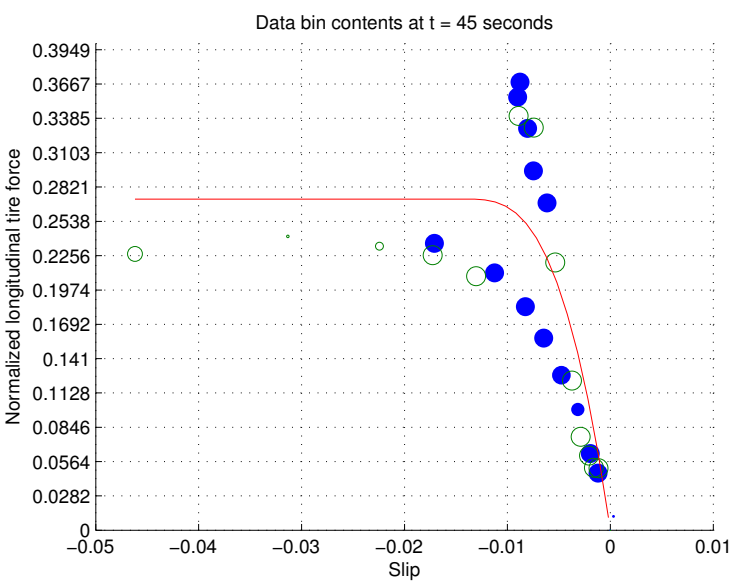

Figure 10.4 Storage bin contents after the change from snow to asphalt.

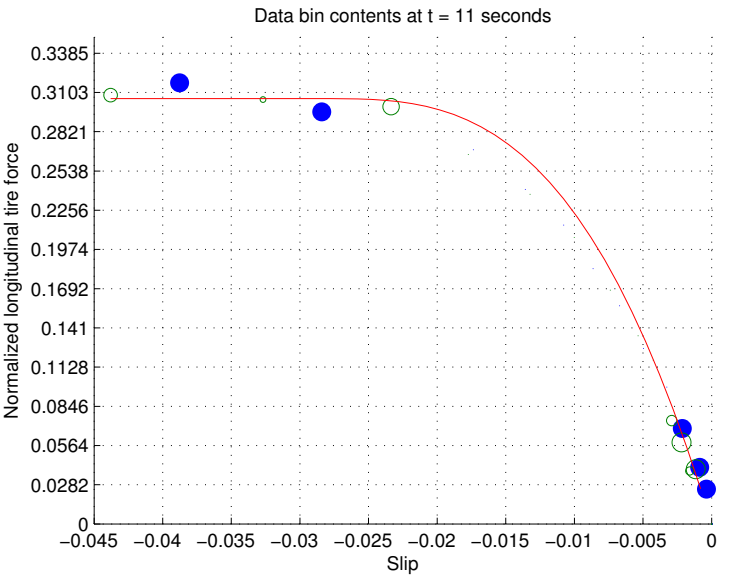

Figure 10.5 Storage bin contents after driving on snow.

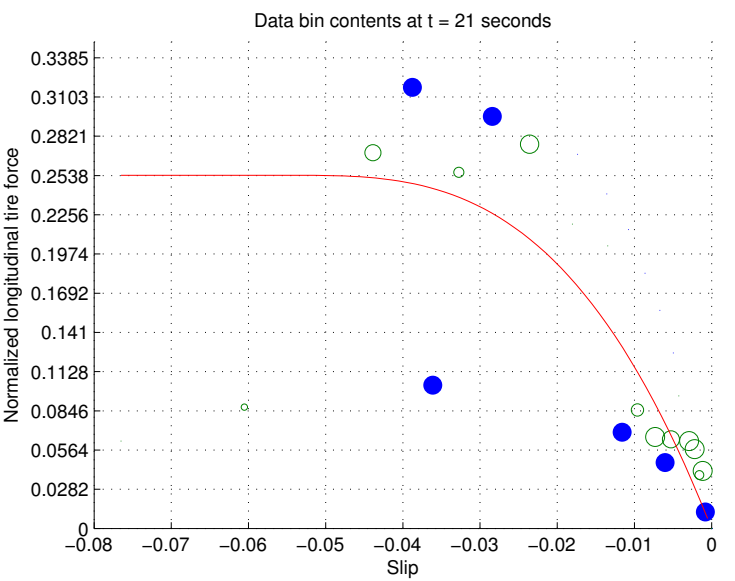

Figure 10.6 Storage bin contents after change from snow to ice.

surface for a long time. This illustrates the need for a change detection algorithm in conjunction with the storage bin method.

The above example illustrated the effects of a surface change from a low friction surface (snow) to a higher friction surface (asphalt). Clearly, any change detection algorithm must also be able to detect a change to a lower friction surface. An example of this is a change from snow to ice. The following example illustrates a surface change from snow to ice, using data collected with a Volvo S40 as in the above example, performing a slow acceleration. Figure 10.5 shows the storage bin contents just before the surface change, after driving on snow. Although there are only a small number of data points, reasonable estimates appear to be obtained.

After the change from snow to ice, data points continue to be collected, resulting in poor estimates as in the previous example. The storage bin contents after 21 seconds are shown in Figure 10.6. Again, the resulting friction characteristic is determined by data from both surface types, and will not converge to the new surface type due to the saturation in the storage bin method.

Change Detection Algorithm

The examples above illustrate the two cases which may occur during a surface change, namely from low friction to high friction (in a relative sense), and the reverse. An intuitive method for designing a change detection algorithm is to examine the behaviour of the storage bin contents during these changes. A heuristic method is likely to be both simple and effective, since it is relatively straightforward to determine what is happening by examining plots such as Figures 10.4 and 10.6.

From Figure 10.4, it is clear to see that a change from a low friction surface to a higher friction surface is likely to be accompanied by an increase in tire stiffness. This in turn is likely to result in data points having a higher normalized force than the current friction estimate, occurring at slips lower than the current cutoff slip (the point at which the tire characteristic saturates). Such behaviour may be used as a simple method to detect a change from low friction to high friction.

A subtle point here is that information from the force bins (F-bins) should be used. At a change to a surface with higher friction, previously empty force bins corresponding to levels higher than the friction level of the old surface are rapidly populated with data points. As has been noted however, these points typically occur for lower slip values, and therefore the slip bins in which they reside are likely to contain data from the previous surface. This in turn implies that there will be a delay in the adjustment of the values in the slip bins due to the filtering.

In Figure 10.6, illustrating the storage bin contents after a change from snow to ice, it is also clear to see the change that has taken place.

However, the situation is slightly more complex than the the case of a change from low to high friction. A change in stiffness is noticeable, but it is less pronounced than in the change from snow to asphalt. The most obvious sign of a change is lower normalized force for larger slips.

As was the case with detection of changes to higher friction, selecting the correct storage bin to use is important. In the event of a change to a low friction surface, it may be expected that higher slips than previously encountered will occur. This will yield new data points in previously empty slip bins which have force values lower than the current friction estimate. This is visible in Figure 10.6, with a slip bin data point at $\sigma_x = -0.06$.

Data from the force bins may also be used, but it is likely that the force bins in question will already be populated with information from the old surface, meaning that the effects of the surface change may be difficult to see. In Figures 10.5 and 10.6, it can be seen that the force bin comprising the interval $0.0846 \leq f_x < 0.1128$ contained no data before the surface change, and therefore clearly shows the effects of the surface change.

A heuristic change detection algorithm may now be constructed based on the observations above. For detection of changes to higher friction, it was found that simply counting the number of occurrences of force bin data values above the current friction level and below the cutoff slip was sufficient to detect changes. For changes to lower friction levels however, it was found necessary to include the residuals in order to perform successful switching. Both slip bin and force bin data was used. The resulting algorithm is summarized in pseudo code in Algorithm 10.1.

Results

Algorithm 10.1 was implemented in C and tested on the two examples of road surface changes described above, as well as numerous other data sets. The change detection flag set by the algorithm was used to reset the contents of the storage bins, effectively erasing all past data. Figure 10.7 shows the effect of using the change detection algorithm on the data set involving the change from snow to asphalt. Comparing this with Figure 10.4, it is clear that the information from before the surface change has been removed. This allows new estimates to be obtained using only data collected from the new surface. Although there are only a limited number of new data points, the new estimates are clearly better than the results in Figure 10.4.

In the case of the change from snow to ice, the results of using the change detection algorithm are shown in Figure 10.8. Again, there are only a small number of data points, but the characteristic is discernible and estimates can be obtained. The resulting friction estimate is roughly 0.1, which is reasonable for ice.

Tuning the change detection algorithm involves a tradeoff between

Algorithm 10.1: Change Detection Algorithm Pseudo Code

```
At each sample;
for each force value in F-bin do
    if force > current μ then
        if slip < cutoff then
            increment high counter
end
for each force value in S-bin do
    if force < current μ then
        if slip > cutoff then
            compute residual ;
            add residual to low counter 1
end
for each force value in F-bin do
    if force < brush model f_x then
        compute residual ;
        add residual to low counter 2
end
if any counter exceeds threshold then
    Change detect = true
```

rapid detection of changes and false activations. It was found to be relatively easy to perform this tuning, based on the data sets available. Figures 10.9 and 10.10 illustrate the response of the change detection algorithms and resulting parameter estimates for the surface changes described in this section. In the first case, the change from snow to asphalt occurred at 36 seconds, and in the second case the change from snow to ice occurred at 12 seconds.

In Figures 10.9, the time required for detection of the change of road surface can be seen to be less than two seconds. After the change is detected, the estimates converge to their new values in two to three seconds. This gives a total of roughly five seconds from the surface change to convergence of the estimates to the new parameters. Similarly, in Figure 10.10, approximately two seconds are required to detect the surface change, and a further two seconds are required for parameter convergence.

Clearly, the speed of both the change detection and subsequent parameter convergence are highly dependent on the level of excitation. Slow accelerations are performed in both of the examples, which provides reasonable excitation.

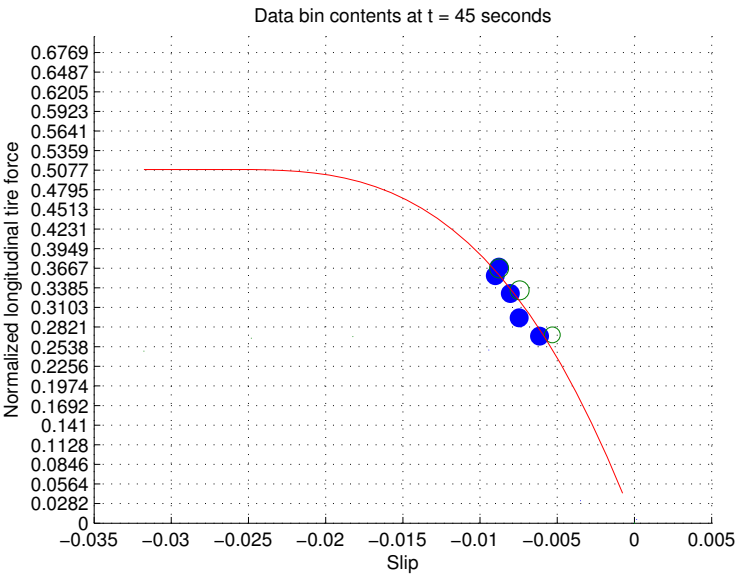

Figure 10.7 Bin contents after change from snow to asphalt, with change detection.

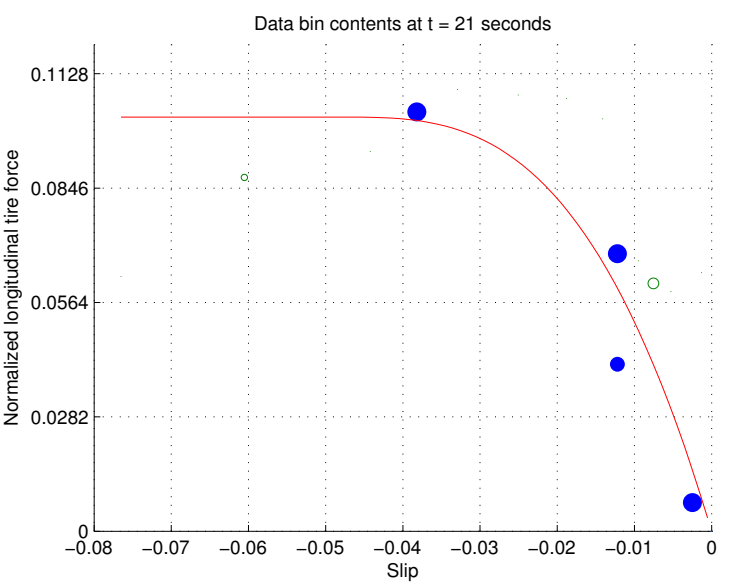

Figure 10.8 Bin contents after change from snow to ice, with change detection.

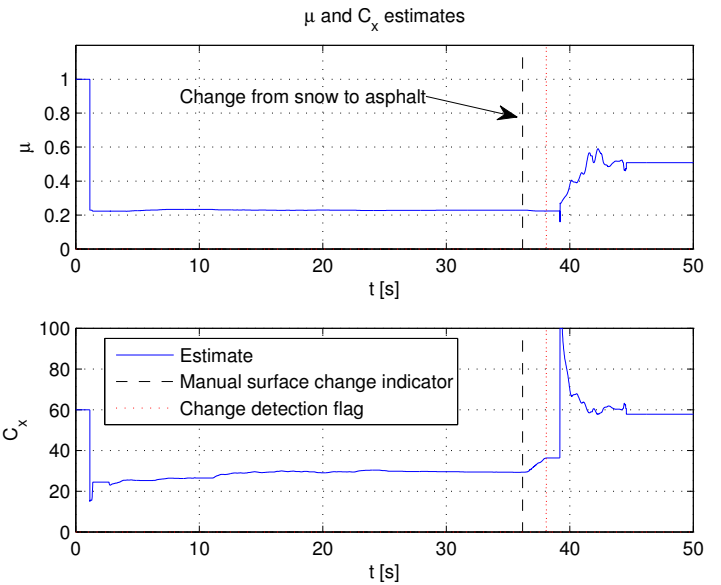

Figure 10.9 Parameter estimates during change from snow to asphalt.

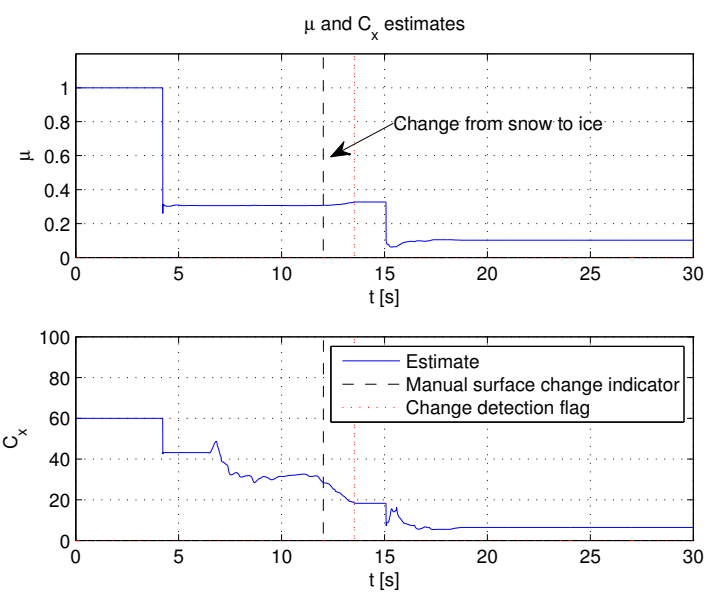

Figure 10.10 Parameter estimates during change from snow to ice.

Figure 10.11 Volvo FH12 test vehicle at Colmis proving ground, Arjeplog, Sweden, February 2008

10.3 Friction Estimation for Heavy Vehicles

The methods described in this chapter have been derived for front wheel drive passenger cars. They are however easily transferable to other vehicle types. To test the estimation algorithm on large trucks, measurement data was collected in February 2008 in Arjeplog, Sweden, using a Volvo FH12, shown in Figures 10.11 and 10.12.

Conversion of the existing estimation algorithm for use with the FH12 was straightforward, the main difference being that the vehicle uses rear wheel drive, meaning the front wheels must be used for the velocity reference. The truck was used without a trailer, but a cargo rack carrying an additional load was installed, bringing the total weight of the vehicle to 12 tonnes.

Figure 10.13 shows the storage bin contents after driving on a snow-covered lake, performing a slow acceleration. The estimation results indicate a friction level of roughly 0.43. Figure 10.14 illustrates the convergence of the parameters in this case.

Figure 10.15 shows the storage bin contents after driving on rough ice, again performing a slow acceleration. In this case the friction level is found to be roughly 0.36. Figure 10.16 shows the convergence of the parameters for this case.

These examples highlight the advantages of using numerical friction

Figure 10.12 The FH12 test vehicle performing a maneuver on a frozen lake at Colmis proving ground. Note that the surface consists of a thin layer of snow on top of uneven ice. The wide variety of possible driving surfaces highlights the problems of classification-based estimation techniques.

estimation, as opposed to surface classification. Road surfaces, particularly in winter conditions, form a spectrum of surface types, rather than discrete classes. In addition, the friction depends not only on the road surface but also on the tire. Comparing the results obtained with the passenger vehicle to those obtained with the truck, it appears that higher friction levels occur in the case of the truck, for similar surface types. This result may be expected, since the truck has a double rear axle and therefore has more driven wheels than the passenger car.

10.4 Conclusions

In this chapter, a model-based approach to friction estimation has been developed. Using the brush model, a relatively simple parameter estimation problem can be posed, allowing estimation of the friction and tire stiffness parameters. A change detection algorithm is proposed, which allows rapid detection of abrupt changes in surface condition. In addition, the estimation scheme has been modified for use with heavy vehicles, and tested using measurement data from a Volvo FH12.

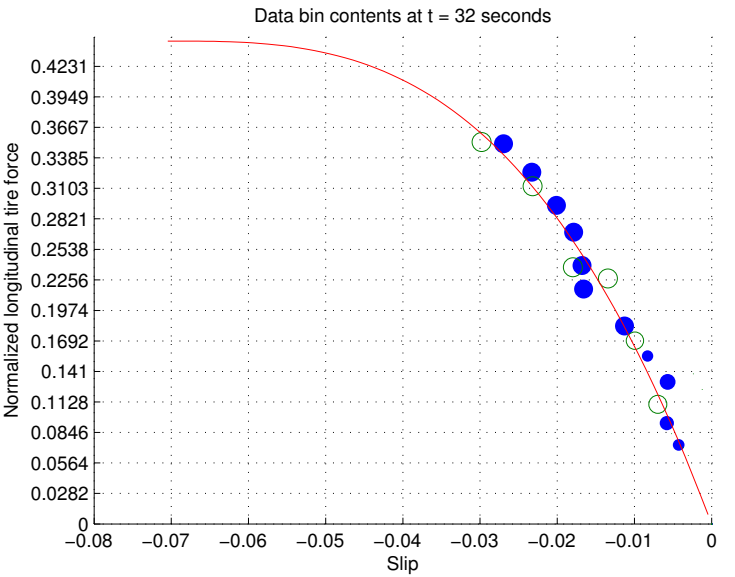

Figure 10.13 Storage bin contents for estimation on snow.

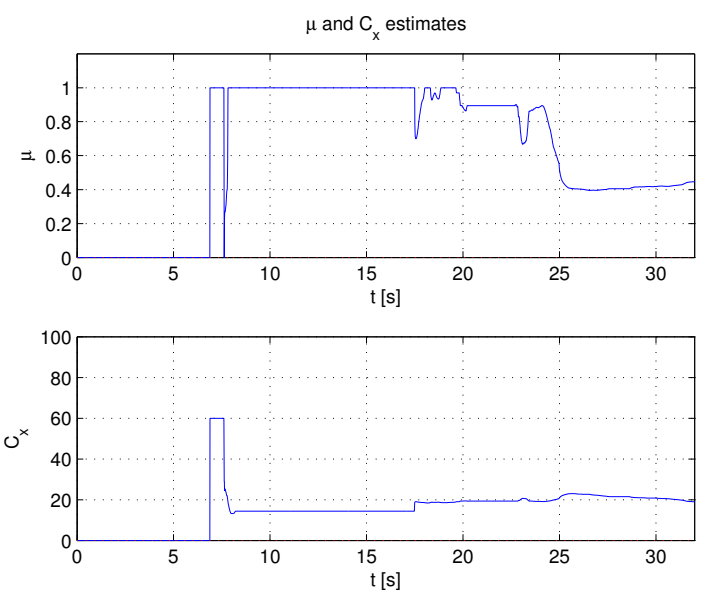

Figure 10.14 Parameter estimates for estimation on snow.

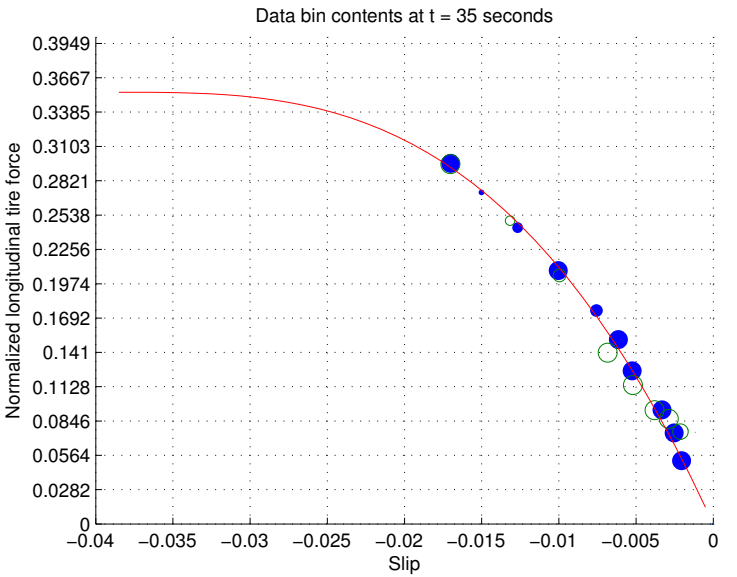

Figure 10.15 Storage bin contents for estimation on ice.

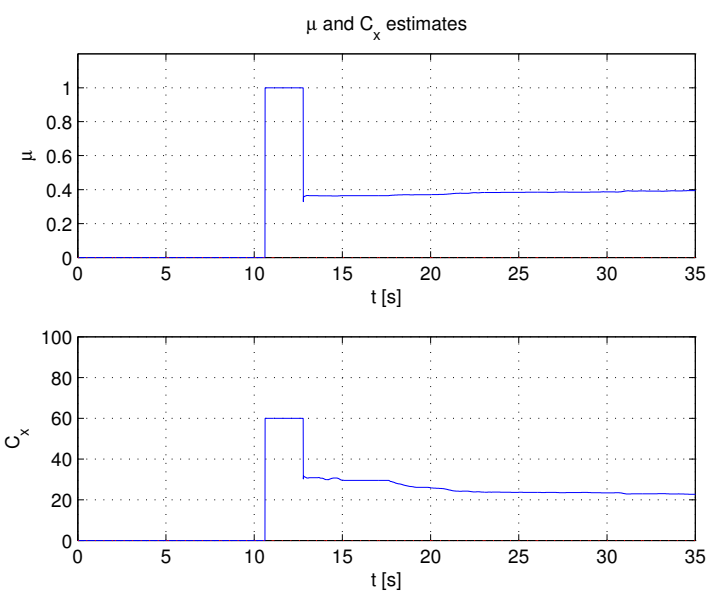

Figure 10.16 Parameter estimates for estimation on ice.

11

Conclusions

11.1 Summary of Contributions

Rollover Mitigation

In this thesis a control strategy for a vehicle dynamics control system has been presented. The primary aim of the system is the prevention of vehicle rollover, but the system also has similar functionality to today's Electronic Stability Program (ESP) systems in terms of yaw control. This follows the current trend in automotive control design towards the integration of control systems.

The control strategy is based on the use of control allocation, where the control design is performed with respect to generalized forces or virtual controls, which are then mapped to actuator commands. This mapping is nontrivial, since a number of constraints must be taken into account. The control allocation strategy uses convex optimization to obtain an optimal mapping from virtual controls to actuator commands, guaranteeing that the constraints are not violated. The use of convex optimization in this context requires careful formulation of the problem in order to obtain an algorithm suitable for real-time implementation.

The convex optimization problem arising from the control allocation algorithm is the most important factor determining the computational complexity of the controller. In order to minimize computation time, numerical methods for solving this type of convex optimization problem have been studied. It is known that active set methods can be used to solve such problems efficiently. In this thesis a modification to the classical primal active set algorithm is proposed. The modification is inspired by numerical methods for large-scale problems, which concentrate on identifying the optimal active set rapidly. The modified active set algorithm allows

the addition of multiple constraints to the working set, and uses a new method for updating the iterates. This gives the modified algorithm several attractive properties for real-time use, including reduced reliance on hotstarting for improved performance.

The modified active set algorithm is also well suited to aerospace control allocation problems, which are of similar structure to the vehicle dynamics application in this thesis, and require similar real-time performance.

Friction Estimation

Knowledge of the coefficient of friction is critical for the performance of vehicle dynamics control systems, including the rollover mitigation controller presented in this thesis. In Chapter 10 a model-based approach to friction estimation is presented, extending results described in [Svendenius, 2007a]. The physically-derived yet simple brush model is used. This model is well suited friction estimation because it contains only two parameters, namely the tire stiffness and friction coefficient. This allows a relatively simple estimation problem to be posed.

To avoid problems related to excitation, a data filtering method based on storage bins is employed. This method is well suited to the problem at hand, which involves the identification of a static relationship. However, the storage bin method suffers from problems during abrupt changes in road surface type. To solve these problems, a change detection method has been developed which is based on the information contained in the storage bins. The method has been tested on a number of data sets of maneuvers in winter driving conditions, and found to work reliably. In addition, the friction estimation methods have been extended for use with heavy vehicles, and tested using data collected from a large truck in winter conditions.

11.2 Conclusions and Future Work

Rollover Mitigation

The strategy has been shown to work well in simulations with a highly realistic vehicle model. The algorithm was observed to be capable of robustly preventing rollover, and relatively little tuning was required to obtain good performance.

Control Allocation The linear tire force approximation, which was important for the formulation of a suitable quadratic programming problem

in the control allocation strategy, was shown to be valid. The resulting QP problem can be solved fast enough to allow real-time implementation, and the control allocator was shown to be capable of accurately reproducing the desired generalized forces.

The strategy has a number of advantages over existing methods, which typically use rule-based algorithms to determine how braking force should be distributed. All of the wheels are used as actuators, rather than only the front wheels. The tuning parameters are few in number and are intuitive to understand. It was observed that relatively little tuning of the control allocator was required in order to achieve good performance.

The current algorithm requires knowledge of a number of important parameters, such as the friction coefficient and loading conditions. In addition, the algorithm, like virtually all VDC functions, is reliant on a number of additional functions, including observers and sensor signal processors. Knowledge of these parameters, and the operation of the supporting functions cannot always be guaranteed. To counteract such disturbances, it would be desirable to 'close the loop' on the control allocator, that is, to introduce an inner feedback loop capable of adjusting parameters in the control allocator. This would require observers for the virtual control signals. Feedback strategies for control allocation are discussed in [Tjønnås and Johansen, 2007b; Tjønnås and Johansen, 2008]

Robustness As previously stated, the strategy has been observed to be robust in the simulations, in the sense that rollover is prevented for a wide range of parameter values and operating conditions. However, it is conceivable that more could be done in the control design to guarantee robustness. The use of nonlinear damping to counteract the effects of uncertainty in the yaw rate control design is an example of this.

Vehicle trials Although the algorithm performs well in simulations, experiments are required in order to confirm the operation of the algorithm in a real vehicle. It was not possible to carry out full vehicle trials during the course project. It would be particularly interesting to carry out tests using both the rollover controller and the friction estimator simultaneously, since knowledge of the friction is perhaps the most important source of uncertainty in the controller.

Friction Estimation

The model-based approach used in the friction estimation was found to work well in a large number of situations. The method is easily transferable to different vehicles without the need for extensive recalibration. No additional sensors are required.

A key issue is the range of scenarios for which the estimator works correctly. Currently, the estimator is only activated during situations involving driving in a straight line. Work has been done to extend the region of applicability to involve gentle cornering, but as yet this has not been fully resolved. An important factor in these situations is the slip, which must be known accurately.

More complex extensions are the cases of braking, and driving in vehicles with all wheel drive. In these situations there are no free-rolling wheels to provide velocity references, and determining the slip is extremely difficult. Some work has been done using GPS information to provide a velocity reference. The work shows promise but satisfactory results have not yet been obtained. The use of a velocity reference independent of the wheels requires that the rolling radii be known very accurately in order to provide accurate slip values. When free-rolling wheels are used, the relative difference between rolling and driven wheel radii is of more importance.

12

Bibliography

Andreasson, J. (2006): *On Generic Road Vehicle Motion Modelling and Control*. PhD thesis, KTH, Aeronautics and Vehicle Engineering, Teknikringen 8, 10044 Stockholm.

Åström, K. J. and B. Wittenmark (1990): *Computer Controlled Systems—Theory and Design*. Prentice-Hall, Englewood Cliffs, New Jersey.

Åström, K. J. and B. Wittenmark (1995): *Adaptive Control*. Addison-Wesley, Reading, Massachusetts.

Bengtsson, J. (2001): "Adaptive cruise control and driver modeling." Licentiate Thesis ISRN LUTFD2/TFRT--3227--SE. Department of Automatic Control, Lund Institute of Technology, Sweden.

Bengtsson, J. (2004): *Closed-Loop Control of HCCI Engine Dynamics*. PhD thesis ISRN LUTFD2/TFRT--1070--SE, Department of Automatic Control, Lund Institute of Technology, Lund University, Sweden.

Björck, Å. (1996): *Numerical methods for least squares problems*. SIAM.

Bond, C. (2008): "Linear algebra ports from Algol to C." Internet. http://www.crbond.com/linear.htm.

Bordignon, K. A. (1996): *Constrained control allocation for systems with redundant control effectors*. PhD thesis, Virginia Polytechnic Institute and State University.

Boyd, S. and L. Vandenberghe (2004): *Convex Optimization*. Cambridge University Press, New York, NY, USA.

Carlson, C. R. and J. Gerdes (2003a): "Nonlinear estimation of longitudinal tire slip under several driving conditions." *American Control Conference, 2003. Proceedings of the 2003*, **6**, pp. 4975–4980 vol.6.

Carlson, C. R. and J. C. Gerdes (2003b): "Optimal rollover prevention with steer by wire and differential braking." In *Proceedings of the ASME International Mechanical Engineering Congress and Exposition*. Washington, D.C. USA.

CEmACS (2007): "Complex embedded automotive control systems." Internet. http://www.hamilton.ie/cemacs/.

Chen, B. and H. Peng (2001): "Differential-braking-based rollover prevention for sport utility vehicles with human-in-the-loop evaluations." *Vehicle System Dynamics*, **36:4–5**, pp. 359–389.

Chen, B. C. and H. Peng (1999): "A real-time rollover threat index for sports utility vehicles." In *Proceedings of the American Control Conference*.

Chen, B. C. and H. Peng (2000): "Rollover prevention for sports utility vehicles with human-in-the-loop evaluations." In *5th Int'l Symposium on Advanced Vehicle Control*.

Colmis (2008): "Colmis proving grounds." Internet. http://www.colmis.se.

Dahlberg, E. (2001): *Commercial Vehicle Stability - Focusing on Rollover*. PhD thesis, Royal Institute of Technology, Stockholm, Sweden.

Darengosse, C., P. Chevrel, and R. Le Doeuff (2000): "A linear parameter-varying flux observer: design and experimentation." *Power Electronics Specialists Conference, 2000. PESC 00. 2000 IEEE 31st Annual*, **3**, pp. 1112–1117 vol.3.

Davidson, J. B., F. J. Lallman, and W. T. Bundick (2001): "Real-time adaptive control allocation applied to a high performance aircraft." In *Proceedings of the 5th SIAM Conference on Control & Its Applications*.

Forkenbrock, G. J., W. Garrot, M. Heitz, and B. C. O'Harra (2002): "A comprehensive experimental examination of test maneuvers that may induce on-road, untripped, light vehicle rollover - phase iv of nhtsa's light vehicle rollover research program." Technical Report. National Highway Traffic Safety Administration.

Fredriksson, J., J. Andreasson, and L. Laine (14-17 Dec. 2004): "Wheel force distribution for improved handling in a hybrid electric vehicle using nonlinear control." *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, **4**, pp. 4081–4086 Vol.4.

Fritzson, P. (2003): *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*. Wiley-IEEE Computer Society Pr.

Gäfvert, M. (2003): *Topics in Modeling, Control, and Implementation in Automotive Systems*. PhD thesis ISRN LUTFD2/TFRT--1066--SE, Department of Automatic Control, Lund Institute of Technology, Sweden.

Gäfvert, M. and J. Svendenius (2003): "Construction of novel semi-empirical tire models for combined braking and cornering." Technical Report ISRN LUTFD2/TFRT--7606--SE. Department of Automatic Control, Lund Institute of Technology, Sweden.

Gäfvert, M. and J. Svendenius (2005): "A novel semi-empirical tire model for combined slips." *Vehicle System Dynamics*, **5:43**.

Gäfvert, M., J. Svendenius, and J. Andreasson (2006): "Implementation and application of a semi-empirical tire-model in multi-body simulation of vehicle handling." In *Proceedings of the 8th International Symposium on Advanced Vehicle Control*. Taipei, Taiwan.

Gaspar, P., I. Szaszi, and J. Bokor (2006): "Rollover stability control in steer-by-wire vehicles based on an LPV method." *International Journal of Heavy Vehicle Systems*, **13:1–2**, pp. 125–143.

Gillespie, T. N. (1992): *Fundamentals of Vehicle Dynamics*. Society of Automotive Engineers, Warrendale, PA.

Gustafsson, F. (15-18 Sep 1996): "Estimation and change detection of tire-road friction using the wheel slip." *Computer-Aided Control System Design, 1996., Proceedings of the 1996 IEEE International Symposium on*, pp. 99–104.

Gustafsson, F. (1997): "Slip-based tire-road friction estimation." *Automatica*, **33:6**, pp. 1087–1099.

Gustafsson, F., S. Ahlqvist, U. Forssell, and N. Persson (2001): "Sensor fusion for accurate computation of yaw rate and absolute velocity."

Hager, W. W., C. Shih, and E. O. Lundin (1996): "Active set strategies and the lp dual active set algorithm." Department of Mathematics, University of Florida, Gainesville, FL.

Härkegård, O. (2002): "Efficient active set algorithms for solving constrained least squares problems in aircraft control allocation." In *Proceedings of the 41st IEEE Conference on Decision and Control*. Las Vegas, Nevada USA.

Härkegård, O. (2003): *Backstepping and Control Allocation with Applications to Flight Control*. PhD thesis Linköping Studies in Science and Technology. Thesis No 820, Department of Electrical Engineering, Linköping University, SE–581 83 Linköping, Sweden.

Härkegård, O. (2004): "Quadratic programming control allocation toolbox." Internet. http://www.mathworks.com/matlabcentral/.

Imsland, L., T. A. Johansen, T. I. Fossen, J. C. Kalkkuhl, and A. Suissa (2005): "Vehicle velocity estimation using modular nonlinear observers." In *IEEE Conf. Decision and Control, Sevilla, Spain*.

Imsland, L., T. A. Johansen, T. I. Fossen, J. C. Kalkkuhl, and A. Suissa (2006): "Nonlinear observer for vehicle velocity estimation." In *SAE World Congress*. Detroit, USA.

Jansson, J. (2005): *Collision Avoidance Theory with Application to Automotive Collision Mitigation*. PhD thesis Linköping Studies in Science and Technology. Dissertations. No. 950, Linköping University, Linköping University, 581 83 Linköping, Sweden.

Johansen, T., J. Kalkkuhl, J. Ludemann, and I. Petersen (2001): "Hybrid control strategies in abs." *American Control Conference, 2001. Proceedings of the 2001*, **2**, pp. 1704–1705 vol.2.

Johansen, T., I. Petersen, J. Kalkkuhl, and J. Ludemann (Nov. 2003): "Gain-scheduled wheel slip control in automotive brake systems." *Control Systems Technology, IEEE Transactions on*, **11:6**, pp. 799–811.

Johansson, B. and M. Gäfvert (2004): "Untripped SUV rollover detection and prevention." In *43rd IEEE Conference on Decision and Control*.

Johansson, R. and A. Rantzer, Eds. (2003): *Nonlinear and Hybrid Systems in Automotive Control*. Springer-Verlag, London. ISBN 1-85233-652-8.

Julier, S. J. and J. K. Uhlmann (1997): "A new extension of the kalman filter to nonlinear systems." In *Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, Simulation and Controls*.

Khalil, H. (2002): *Nonlinear Systems*, third edition. Prentice Hall.

Kiencke, U. and L. Nielsen (2000): *Automotive Control Systems: For Engine, Driveline and Vehicle*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Kristić, M., I. Kanellakoopulos, and P. Kokotović (1995): *Nonlinear and Adaptive Control Design*. John Wiley & Sons.

Lee, C., K. Hedrick, and K. Yi (2004): "Real-time slip-based estimation of maximum tire-road friction coefficient." *Mechatronics, IEEE/ASME Transactions on*, **9:2**, pp. 454–458.

Liu, Y. and J. Sun (21-23 Jun 1995): "Target slip tracking using gain-scheduling for antilock braking systems." *American Control Conference, 1995. Proceedings of the*, **2**, pp. 1178–1182 vol.2.

Mitschke, M. (1995): *Dynamik der Kraftfahrzeuge*, third edition. Springer.

Modelon (2008): "Vehicle dynamics library." Internet. http://www.modelon.se.

Moré, J. J. and G. Toraldo (1989): "Algorithms for bound constrained quadratic programming problems." *Numeriche Mathematik*, **55**, pp. 377–400.

Odenthal, D., T. Bunte, and J. Ackermann (1999): "Nonlinear steering and braking control for vehicle rollover avoidance."

Pacejka, H. B. (2002): *Tyre and Vehice Dynamics*. Butterworth Heinemann.

Pasterkamp, W. R. and H. B. Pacejka (1997): "The tyre as a sensor to estimate friction." *Vehicle System Dynamics*, **27:5**, pp. 409 – 422.

Plumlee, J. H., D. M. Bevley, and A. S. Hodel (2004): "Control of a ground vehicle using quadratic programming based control allocation techniques." In *Proceedings of the American Control Conference*. Boston, Massachusetts, USA.

Poonamallee, V., S. Yurkovich, A. Serrani, and D. Doman (2004): "A nonlinear programming approach for control allocation." *American Control Conference, 2004. Proceedings of the 2004*, **2**, pp. 1689–1694 vol.2.

ReBEL (2008): "Rebel : Recursive bayesian estimation library and toolkit for matlab." Internet. http://choosh.csee.ogi.edu/rebel/.

Schofield, B. (2006): "Vehicle dynamics control for rollover prevention." Licentiate Thesis ISRN LUTFD2/TFRT--3241--SE. Department of Automatic Control, Lund University, Sweden.

Schofield, B. (2008): "On active set algorithms for solving bound-constrained least squares control allocation problems." In *Proceedings of the 2008 American Control Conference*. Seattle, USA.

Schofield, B. and T. Hägglund (2008): "Optimal control allocation in vehicle dynamics control for rollover mitigation." In *Proceedings of the 2008 American Control Conference*. Seattle, USA.

Schofield, B., T. Hägglund, and A. Rantzer (2006): "Vehicle dynamics control and controller allocation for rollover prevention." In *Proceedings of the IEEE International Conference on Control Applications*. Munich, Germany.

Shakhtarin, B. I. (2006): "Stratonovich nonlinear optimal and quasi-optimal filters." *Journal of Communications Technology and Electronics*, **51:11**, pp. 1248–1260.

Slotine, J. E. and W. Li (1991): *Applied Nonlinear Control*. Prentice Hall.

Solyom, S. (2004): *Control of Systems with Limited Capacity*. PhD thesis ISRN LUTFD2/TFRT--1069--SE, Department of Automatic Control, Lund Institute of Technology, Sweden.

Spiegel, M. R. (1967): *Theory and Problems of Theoretical Mechanics*. Schaum Publishing.

Svendenius, J. (2003): "Tire models for use in braking applications." Licentiate Thesis ISRN LUTFD2/TFRT--3232--SE. Department of Automatic Control, Lund University, Sweden.

Svendenius, J. (2007a): *Tire modeling and Friction Estimation*. PhD thesis ISRN LUTFD2/TFRT--1077--SE, Department of Automatic Control, Lund University, Sweden.

Svendenius, J. (2007b): "Validation of the brush model towards VTI-measurement data recorded at Hällered 2005." Technical Report ISRN LUTFD2/TFRT--7616--SE. Department of Automatic Control, Lund University, Sweden.

Svendenius, J. (2007c): "Validation of the brush model towards VTI-measurement data recorded in Arjeplog 2006." Technical Report ISRN LUTFD2/TFRT--7617--SE. Department of Automatic Control, Lund University, Sweden.

Svendenius, J. and M. Gäfvert (2004a): "A brush-model based semi-empirical tire-model for combined slips." In *SAE World Congress*.

Svendenius, J. and M. Gäfvert (2004b): "A semi-empirical tire-model for transient combined-slip forces." In *AVEC '04*.

Svendenius, J. and M. Gäfvert (2005): "A semi-empirical tire model for combined slips including the effects of cambering." In Lugner and Plöchl, Eds., *Tyre Models For Vehicle Dynamics Analysis*, vol. 43. Taylor & Francis Group, Glasgow.

Svendenius, J. and M. Gäfvert (2006): "A semi-empirical dynamic tire model for combined-slip forces." *Vehicle System Dynamics*, **44:2**, pp. 189–208. Special Issue:AVEC '04: 7th International Symposium on Advanced VEhicle Control 23-27 August 2004 HAN University, Arnhem, The Netherlands.

Svendenius, J., M. Gäfvert, F. Bruzelius, and J. Hultén (2007): "Experimental validation of the brush tire model." In *September 2007 Meeting of the Tire Society*.

Svendenius, J. and B. Wittenmark (2003): "Brush tire model with increased flexibility." In *European Control Conference*.

Tjønnås, J. and T. A. Johansen (2005): "Optimizing nonlinear adaptive control allocation." In *Proceedings of the 16th IFAC World Congress*. Prague, Czech Republic.

Tjønnås, J. and T. A. Johansen (2007a): "On optimizing nonlinear adaptive control allocation with actuator dynamics." In *7th IFAC Symposium on Nonlinear Control Systems*. Pretoria, South Africa.

Tjønnås, J. and T. A. Johansen (2007b): "Optimizing adaptive control allocation with actuator dynamics." In *CDC*. New Orleans, USA.

Tjønnås, J. and T. A. Johansen (2008): "Adaptive control allocation." *Automatica*.

Tøndel, P. and T. A. Johansen (2005): "Control allocation for yaw stabilization in automotive vehicles using multiparametric nonlinear programming." In *American Control Conference*.

Vägverket (2008a): "Intelligent vehicle safety systems." Internet. http://www.ivss.se/.

Vägverket (2008b): "Road friction estimation - rfe." Internet. http://www.ivss.se/templates/ProjectPage.aspx?id=177.

Vägverket (2008c): "Slippery road information system." Internet. http://www.ivss.se/templates/ProjectPage.aspx?id=206.

Wan, E. and R. Van Der Merwe (2000): "The unscented kalman filter for nonlinear estimation." *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, pp. 153–158.

Wilkinson, J. H. and C. Reinsch (1971): *Linear Algebra*. Springer-Verlag.

Wise, K. A., J. S. Brinker, A. J. Calise, D. F. Enns, M. R. Elgersma, and P. Voulgaris (1999): "Direct adaptive reconfigurable flight control for a tailless advanced fighter aircraft." *International Journal of Robust and Nonlinear Control*, **9**, pp. 999–1012.

Wong, J. Y. (1993): *Theory of Ground Vehicles*, second edition. John Wiley and Sons.

Yi, J., L. Alvarez, R. Horowitz, and C. de Wit (2000): "Adaptive emergency braking control using a dynamic tire/road friction model." *Proceedings of the 39th IEEE Conference on Decision and Control*, **1**, pp. 456–461.

A

Nomenclature

The symbols used in this thesis are summarized in the table below. As far as possible, unique symbols have been used for each quantity, however, a small number of symbols are re-used in different contexts. Equation numbers of relevant definitions are provided.

Symbol	Description
Vehicle variables	
x	Longitudinal position in frame
y	Lateral position in frame
z	Vertical position in frame
v_x	Instantaneous velocity in x direction
v_y	Instantaneous velocity in y direction
$\dot{\psi}$	Yaw rate
ϕ	Roll angle
$\dot{\phi}$	Roll rate
θ	Pitch angle
β	Sideslip angle
Vehicle parameters	
m	Vehicle mass
m_e	Vehicle mass (empty)
m_b	Load mass
h	Height of CG above roll axis
h_e	CG height (empty)
h_b	Load height

I_{xx}	Moment of inertia about x–axis
I_{yy}	Moment of inertia about y–axis
I_{zz}	Moment of inertia about z–axis
I_{xz}	Product of inertia for x and z axes
a	Distance from front axle to CG position
b	Distance from rear axle to CG position
a_b	Load distance from front axle
l	Half track width
C_ϕ	Total roll stiffness
K_ϕ	Total roll damping
θ_r	Angle between roll axis and x–axis

Tire variables

v_{sx}	Longitudinal tire slip velocity
v_{sy}	Lateral tire slip velocity
v_c	Circumferential velocity
ω	Wheel rotational velocity
ω_0	Wheel rotational velocity (free rolling)
γ	Camber angle
F_x	Longitudinal tire force
F_y	Lateral tire force
F_z	Vertical tire force
F_{ax}	Longitudinal adhesive force in brush model
F_{ay}	Lateral adhesive force in brush model
M_z	Self-aligning torque
δ_x	Longitudinal deflection of tire
δ_y	Lateral deflection of tire
δ_{ys}	Lateral tire deformation due to slip
δ_{xb}	Longitudinal deflection of bristle
δ_{yb}	Lateral deflection of bristles
ψ	Normalized slip
α	Slip angle (2.4)
λ	Longitudinal slip (2.2)
σ_x	Longitudinal physical slip (2.3)

σ_y	Lateral physical slip
σ_x^0	Longitudinal limit slip
σ_y^0	Lateral limit slip

Tire parameters

R_0	Unloaded radius of tire
R_e	Effective rolling radius
R_e^{nom}	Nominal effective rolling radius
C_x	Longitudinal (braking) stiffness
C_y	Lateral (cornering) stiffness
C_λ	Longitudinal stiffness with respect to λ
C_α	Lateral stiffness with respect to α
C_m	Aligning stiffness
C_F	Front lateral stiffness (bicycle model)
C_R	Rear lateral stiffness (bicycle model)
c_x	Incremental longitudinal bristle stiffness
c_y	Incremental lateral bristle stiffness
k_l	Rolling radius load dependence
a	Size of contact patch
μ	Friction coefficient
μ_{sx}	Longitudinal static friction in brush model
μ_{sy}	Lateral static friction in brush model
μ_d	Dynamic (kinetic) friction in brush model

Controller parameters

K_r	Yaw rate controller gain
a_x^d	Desired longitudinal deceleration
$\hat{a}_x^{\text{on}}$	Threshold for controller switching
$\hat{a}_x^{\text{off}}$	Threshold for controller switching
σ	Tuning parameter in (6.13)
ν	Tuning parameter in (6.13)
W_v	Weighting matrix in (6.24)
W_u	Weighting matrix in (6.24)
γ	Parameter in (6.24)

Friction Estimation

θ	Parameters for estimation
ϕ	Regressor vector
Φ	Matrix of regression vectors
$\mathcal{Y}$	Vector of observations
V	Cost function
w	Weight in cost function
J	Jacobian matrix
H	Hessian matrix