

**Design and Validation of
Advanced Driver Assistance Systems**

Olaf Gietelink

Design and Validation of Advanced Driver Assistance Systems

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof.dr.ir. J.T. Fokkema,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op
maandag 19 november 2007 om 15:00 uur
door

Olaf Jeroen GIETELINK

werktuigkundig ingenieur
geboren te Amsterdam.

Dit proefschrift is goedgekeurd door de promotoren:

Prof.dr.ir. M. Verhaegen

Prof.dr.ir. B. De Schutter

Samenstelling promotiecommissie:

Rector Magnificus

Prof.dr.ir. M. Verhaegen

Prof.dr.ir. B. De Schutter

Prof. Dr. R. Tempo

Prof. Dr. H. Rohling

Prof.dr.ir. B. van Arem

Prof.dr.ir. J.A. Mulder

Ir. J. Ploeg

Prof.dr.ir. J. Hellendoorn

voorzitter

Technische Universiteit Delft, promotor

Technische Universiteit Delft, promotor

Politecnico di Torino

Technische Universität Hamburg-Harburg

Universiteit Twente

Technische Universiteit Delft

TNO Industrie en Techniek

Technische Universiteit Delft, reservelid

The work in this thesis was supported by the Netherlands Organisation for Applied Scientific Research TNO, the Netherlands Research School for Transport, Infrastructure and Logistics TRAIL, the TNO TRAIL Transport Research T3 program, the Transport Research Centre Delft of Delft University of Technology, the European 6th Framework Network of Excellence “HYbrid CONtrol: Taming Heterogeneity and Complexity of Networked Embedded Systems (HYCON)”, contract number FP6-IST-511368, and a Travel Grant from the Netherlands Organisation for Scientific Research NWO.



TRAIL Thesis Series T2007/11, The Netherlands TRAIL Research School

Published and distributed by: TRAIL Research School

P.O. Box 5017

2600 GA Delft

The Netherlands

T: + 31 (0) 15 27 86046

F: + 31 (0) 15 27 84333

E: info@rsTRAIL.nl

ISBN: 978-90-5584-091-5

Keywords: advanced driver assistance systems, hardware-in-the-loop simulation, controller design and validation, randomized algorithms

Copyright © 2007 by Olaf Gietelink.

Author E-mail: olaf.gietelink@gmail.com

All rights reserved. No part of the material protected by this copyright notice may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without written permission of the author.

Printed in The Netherlands by Gildeprint B.V.

Preface

As long I can remember, I have been fascinated by anything that drives, flies, or floats. This fascination and my desire to know ‘how things work’ encouraged me to take up a career in technology and study mechanical engineering. After finishing my studies, I thought had left university for good and joined the Netherlands Organisation for Applied Scientific Research TNO. During my first year at TNO, several projects on advanced driver assistance systems, as well as the plans for the construction of the VeHIL laboratory, encouraged me to investigate this research area in more depth. This resulted in the formulation of a Ph.D. project, and by 2003 I was back at university again for a few days a week. Some years later I can look back at a turbulent experience, which allowed me to work at the forefront of technology and to learn a great deal on many topics, not in the least place in personal areas.

Besides moments of euphoria, every Ph.D. project has its pitfalls and disappointments. Nevertheless, I could always count on my supervisors, who motivated me to explore new ideas in this exiting field of research. I am therefore especially grateful to Michel Verhaegen and Bart De Schutter at the Delft Center for Systems and Control (DCSC) of Delft University of Technology, and Jeroen Ploeg at TNO Science and Industry who have supervised the work in this thesis. I have always been impressed with the ingenuity of Bart’s comments on my manuscripts, and I could not think of a better and more committed supervisor. On the practical side of my research, Jeroen played an important role in managing the never-ending troubles of practical demonstrator vehicles and acting as a sounding board on the practical application of theoretical research. His perseverance in getting things done has contributed significantly to the results in this thesis. Bart and Jeroen, thank you so much for your help and support over the past years! I also would like to thank the other members of the Ph.D. examination committee for their time and effort in reviewing the manuscript.

Furthermore, I am grateful to Leo Kusters for his help in setting up the initial project proposal. It was his vision that led to the construction of the VeHIL laboratory, and he played an important role in obtaining the financial support from TNO and TRAIL Research School. I would also like to thank Ben Jansen, Marjolein Baart, and Cees Ruijgrok of TNO Traffic and Transport for their confidence in my proposal. Furthermore, the financial support of the European 6th Framework Network of Excellence ‘HYCON’ and the Transport Research Centre Delft of Delft University of Technology is gratefully acknowledged.

Many colleagues at the Integrated Safety department of TNO in Helmond have contributed to this thesis in one way or the other through their collaboration in research projects. I would especially like to thank Rob van de Pijpekamp, Bart Scheepers, Chris Huijboom, Edwin Stierman, Zoltán Papp, Martijn Koopman, and Pieter Schutyser for their support in several VeHIL projects. Furthermore, Joris Coolen, Fred Gordebeke, and Ton Ratten took care of the instrumentation, preparation, and maintenance of prototype vehicles that were

used as demonstrators in the case studies. Stefanie Buijssen, Floris Leneman, Dirk Verburg, Marcel Wantenaar, and Walter Renes, amongst others, were responsible for the continuous improvement of the simulation environment PreScan and developing some of the simulation models that were implemented in the simulation work of this thesis. Ineke Glaser of the TNO Information Services department has been very helpful with collecting literature.

The case studies in this thesis have been carried out in conjunction with other TNO projects, such as the AV3 project, in which Kamel Labibes, Paul Lemmen, Johan van den Heuvel, Wannes van der Mark, and Christiaan Lievers contributed to the results of Chapter 8. Similarly, the test results of the cooperative adaptive cruise control system, presented in Chapter 7, were carried out within the framework of the TNO SUMMITS program, for which Falke Hendriks and Friedes Lameris supported the VeHIL experiments and test drives. Together with Sven Jansen, Dehlia Willemsen, Mark Lammers, and Hans-Martin Duringhof, I also participated in the PReVENT subproject SASPENCE. I would like to thank all project partners and especially Andrea Saroldi from Centro Ricerche FIAT for their collaboration in this European project, which contributed to the results of Chapter 6.

I would also like to thank Minoo Shah of Delphi Corporation and co-workers for the pleasant collaboration in the VeHIL pre-crash testing project and writing a joint paper on the topic. The release for publication of the test results with a driver warning system by DAF is also acknowledged.

During the summer of 2004 I stayed at the California PATH program as a visiting scholar to collaborate on research on advanced driver assistance systems, fault management, and related research topics. I would like to thank Jim Misener for his help in arranging this visit, and Steve Shladover, Xiao-Yun Lu, Delphine Cody, and Swe Kuang for their collaboration during that period. The financial support through a Travel Grant from the Netherlands Organisation for Scientific Research NWO is greatly appreciated.

I am grateful to all the students whom I had the chance to supervise during their internship and or Masters thesis at TNO: Freek Liefhebber, Raimond Haan, Arnoud van den Dobbelen, Wei Huang, and Robin van Tongeren. Their work and practical assistance provided a substantial contribution to this thesis.

The fellow Ph.D. students and the secretaries at the DCSC department have always made the research work go easier. I could always turn to them for help on formalities, \LaTeX , MATLAB, English writing style, and other topics. I am especially grateful to Redouane Hallouzi for his collaboration on fault management research, which contributed to Chapter 3, writing a joint paper, supervising students, and carpooling to Delft for many years.

Furthermore, I am fortunate to be surrounded by so many good friends. Several of them, especially Wessel and my fellow Ph.D. students and paranimfs Matthijs and Rutger took the effort of reviewing the manuscript and providing out-of-the-box suggestions on my research. During these years we also kept ourselves mutually motivated on our research, but also made sure that we had the necessary distraction from it.

My parents Peter and Ylva have always been there for me and provided me with the skills to succeed in my academic achievements, which culminated in this thesis. Finally, there are not enough words to thank Hester for her endless love and support. During many evenings, weekends, and holidays over the past years she had to relinquish a lot of our time together. She always stood by me with happy and motivating words, and I thank her for that with all my heart. I look forward to our journey together in years to come.

Olaf Gietelink

Contents

Preface	v
1 Introduction	1
1.1 The cost of road traffic to society	1
1.2 Advance of intelligent transportation systems	3
1.3 Problem formulation	6
1.4 Objectives and scope of this thesis	8
1.5 Outline of this thesis	10
1.6 Contributions of this thesis	11
1.7 Publications by the author	12
2 Advanced driver assistance systems (ADASs)	13
2.1 Overview of the state of the art in ADASs	13
2.1.1 Forward collision warning systems	13
2.1.2 Adaptive cruise control systems	17
2.1.3 Pre-crash systems	18
2.2 Enabling technologies for ADASs and their challenges	19
2.2.1 Environment sensor technology	20
2.2.2 Signal processing and control engineering	21
2.2.3 Human-centered design	22
2.2.4 Drive-by-wire technology	23
2.2.5 Automotive communication systems	23
2.2.6 Summary of technological challenges	24
2.3 Requirements and evaluation criteria for ADASs	24
2.3.1 Abstraction of the ADAS control configuration	24
2.3.2 Stability requirements for the control system	25
2.3.3 Functional performance requirements	28
2.3.4 Driver interaction requirements	29
2.3.5 Dependability requirements	30
2.3.6 Compatibility with operating conditions	35
2.4 Scenario definition by microscopic traffic modeling	35
2.4.1 Single-lane traffic modeling	35
2.4.2 Subscenarios in single-lane traffic	36
2.4.3 Modeling and calibration of single-lane scenario parameters	38
2.4.4 Calibration of the Gipps driver model	41
2.5 Impact of disturbances and faults on dependability	43

2.5.1	Fault terminology	44
2.5.2	Classification of faults in ADASs	44
2.5.3	Modeling the form of a fault	45
2.6	Summary	46
3	Fault-tolerant state estimation	47
3.1	Introduction to fault management	47
3.1.1	Fault-tolerant system architecture	47
3.1.2	Robustness and redundancy for component fault tolerance	48
3.1.3	Fault tolerance for control systems by analytical redundancy	48
3.2	Model-based fault detection and isolation (FDI)	49
3.2.1	Parameter estimation methods	50
3.2.2	Dedicated state estimators	50
3.2.3	Fault-detection filters	52
3.2.4	Unknown input observers	52
3.2.5	Generalized observer scheme	53
3.2.6	Parity equations	53
3.2.7	Change detection methods	54
3.2.8	Fault diagnosis	55
3.2.9	Considerations for model-based FDI for ADASs	56
3.3	Demonstrator vehicles	57
3.3.1	Prototype instrumentation	57
3.3.2	Vehicle modeling	59
3.4	State estimation by extended Kalman filtering (EKF)	63
3.4.1	Model equations	63
3.4.2	Observability, implementation, and tuning of the EKF	64
3.4.3	Validation of the state estimation	66
3.5	Fault management of vehicle state sensor faults	68
3.5.1	Generalized observer scheme of EKFs	68
3.5.2	Sensor fault detection and isolation	69
3.5.3	Reconfiguration of state estimation	69
3.5.4	Validation results	69
3.6	Fault management for relative motion estimation	70
3.6.1	Definition of relative motion	70
3.6.2	Data association for sensor fusion	71
3.6.3	Sensor fusion by Kalman filtering	72
3.6.4	Validation results	73
3.7	Summary	74
4	Development of ADASs with vehicle hardware-in-the-loop simulations	75
4.1	Challenges in the ADAS development process	76
4.1.1	Requirements and specification phase	76
4.1.2	Verification and validation	77
4.2	State-of-the-art tool chain	78
4.2.1	Model-in-the-loop simulation	78
4.2.2	Software-in-the-loop simulation	79
4.2.3	Hardware-in-the-loop simulation	79

4.2.4	Full-scale test drives	81
4.3	Vehicle hardware-in-the-loop (VeHIL) simulations	81
4.3.1	Working principle of the VeHIL simulation	81
4.3.2	Substitution of a vehicle dynamics model by a vehicle under test	84
4.3.3	Substitution of a simulated target by a moving base	86
4.3.4	Fault injection for validation of fault management systems	87
4.3.5	Representativeness of VeHIL	88
4.3.6	Added value of VeHIL in the development process of ADASs	89
4.4	VeHIL test results for ADAS applications	90
4.4.1	Sensor calibration	91
4.4.2	Adaptive cruise control (ACC) system	91
4.4.3	Forward collision warning system	92
4.5	Summary	95
5	A methodological framework for probabilistic validation of ADASs	97
5.1	Objectives and methods for control system validation	97
5.1.1	A case study: The ACC control problem	98
5.1.2	Characterization of performance measures by cost functions	98
5.1.3	Verification of the system specifications	99
5.1.4	Evaluation of the cleared parameter set	100
5.1.5	Grid-based techniques to cover the parameter set	101
5.1.6	Evaluation of worst-case performance	102
5.1.7	The need for a new methodology	103
5.2	Randomized algorithms for control system validation	103
5.2.1	Motivation for a probabilistic approach	103
5.2.2	Problem definition for Monte Carlo sampling	104
5.2.3	Upper bounds on the sample complexity	106
5.2.4	Formulation of a randomized algorithm	108
5.2.5	Characteristic properties of randomized algorithms	111
5.3	Methods for reduction of the sample complexity	111
5.3.1	Reduction of the parameter set	112
5.3.2	Sequential estimation using the binomial bound	113
5.3.3	Sequential estimation for a multiplicative accuracy	115
5.3.4	Importance sampling	118
5.3.5	Sample complexity of importance sampling	121
5.3.6	Kernel density estimation of the importance sampling PDF	123
5.3.7	Random number generation	125
5.3.8	Summary of methods for sample complexity reduction	126
5.4	Adaptive importance sampling (AIS)	127
5.4.1	A new algorithm for probabilistic validation	127
5.4.2	Extension to the multi-dimensional problem	129
5.4.3	Convergence of the AIS algorithm	133
5.5	A methodology using PreScan, VeHIL, and test drives	134
5.5.1	Definition of validation objectives	135
5.5.2	Definition of the parameter set	136
5.5.3	System design	137
5.5.4	Generation of a simulation model of the system	138

5.5.5	System construction, integration, and verification	138
5.5.6	Sensitivity analysis of the performance measure	138
5.5.7	Preliminary validation with adaptive importance sampling	139
5.5.8	Improvement of the validation results with VeHIL tests	139
5.5.9	Evaluation of the system benefit with test drives	141
5.6	Summary	141
6	Case study: Validation of a driver information and warning system	143
6.1	A system for safe speed and safe distance	143
6.1.1	Functional requirements	143
6.1.2	Relevant scenarios for driver warning	144
6.1.3	System architecture and prototype description	144
6.1.4	Sensor fusion and scenario assessment	145
6.2	Definition of the validation objectives	146
6.3	Definition of the parameter set	148
6.4	Software-in-the-loop (SIL) simulation in PreScan	148
6.5	Preliminary validation with PreScan-SIL simulation	149
6.6	Functional validation with VeHIL tests	149
6.6.1	Experimental setup	151
6.6.2	Definition of an efficient test schedule	152
6.6.3	Experimental results	152
6.7	The role of test drives	157
6.8	Summary	157
7	Case study: Validation of a cooperative adaptive cruise control system	159
7.1	Introduction to longitudinal vehicle control	159
7.1.1	Gain scheduling for longitudinal vehicle control	160
7.1.2	Control during transitional maneuvers	164
7.1.3	String stability considerations for longitudinal control	165
7.2	Cooperative vehicle control	167
7.2.1	Added value of vehicle-to-vehicle communication	168
7.2.2	An algorithm for cooperative adaptive cruise control (CACC)	168
7.2.3	Hybrid automaton for CACC	169
7.3	Definition of the validation objectives	171
7.4	Definition of the parameter set	171
7.5	PreScan trend study	172
7.5.1	Modeling of cooperative vehicle control	172
7.5.2	Definition of a test schedule for sensitivity analysis	172
7.5.3	Simulation results	172
7.6	Sensitivity analysis with VeHIL experiments	174
7.7	Test drives with the CACC system	175
7.8	Summary	176
8	Case study: Validation of a pre-crash system (PCS)	179
8.1	Tools and methods in the design and validation of PCSs	180
8.1.1	Definition of the validation objectives	180
8.1.2	Definition of the parameter set	181
8.1.3	Pre-crash system design	181

8.1.4	System modeling	181
8.1.5	System construction, integration, and verification	182
8.1.6	Challenges in system validation	182
8.2	VeHIL testing of PCSs	182
8.3	Accident study and parameter set definition	184
8.3.1	Categorization of relevant scenarios	184
8.3.2	Crash parameters	185
8.4	Specification and design of the PCS	188
8.5	Generation of a PreScan model of the PCS	189
8.6	Test schedule selection for pre-crash testing	191
8.7	Sensitivity analysis with VeHIL experiments	193
8.7.1	Head-on collision test	193
8.7.2	Rear-end collision test	196
8.7.3	Validation of the system dependability	196
8.8	Benefit assessment of pre-crash sensing	197
8.9	Summary	199
9	Conclusions and recommendations	201
9.1	Conclusions and contributions of this thesis	201
9.2	Critical discussion of the results	204
9.3	Recommendations for future research	206
	Bibliography	208
	Glossary	233
	Notation and symbols	233
	Coordinate systems and sign conventions	239
	List of abbreviations	241
	Summary	243
	Samenvatting (Summary in Dutch)	245
	About the author	247
	List of TRAIL Thesis Series publications	249

Chapter 1

Introduction

This thesis starts in Sections 1.1 and 1.2 with an overview of societal problems and technological solutions in the field of road traffic. The problem formulation is derived in Section 1.3. Section 1.4 defines the thesis objectives and Section 1.5 presents the outline of this thesis. Finally, Sections 1.6 and 1.7 summarize the contributions and publications that resulted from this thesis.

1.1 The cost of road traffic to society

Since its introduction over a century ago, the automobile has enabled individual mobility for an ever growing part of humankind. The passenger car provides multi-purpose flexible transportation, and plays an important economic, social, and cultural role in human society. Unfortunately, motorized traffic also has several adverse effects on society:

- *Accessibility*: With the dramatic increase of cars on the road since the 1950s, traffic congestion is an ever growing problem. Within the past 25 years, traffic congestion has increased by almost 1000 %, as illustrated in Figure 1.1. Not only are traffic jams a source of driver discomfort, they are also responsible for increased fuel consumption and lost productivity. A recent study by the European Commission [49] shows that these external costs of traffic congestion will increase to 1 % of the gross domestic product (GDP) of the European Union (EU) by 2010.
- *Sustainability*: According to the World Health Organization, road traffic is the main source of urban air pollution and accounts for more than one quarter of greenhouse gas emissions [42]. It therefore has a significant effect on the natural environment, not only on a local scale (smog), but also globally (climate change). In addition, air pollution has major implications for public health. The external costs in terms of air pollution, fuel consumption, climate change, noise pollution, and landscape effects are estimated at 6 % of the EU GDP.
- *Safety*: The human cost of road traffic is also increasing. Another study by the World Health Organization shows that worldwide, an estimated 1.2 million people are killed in road accidents every year and as many as 50 million are injured [187]. These figures are expected to increase by about 65 % over the next 20 years. Throughout the

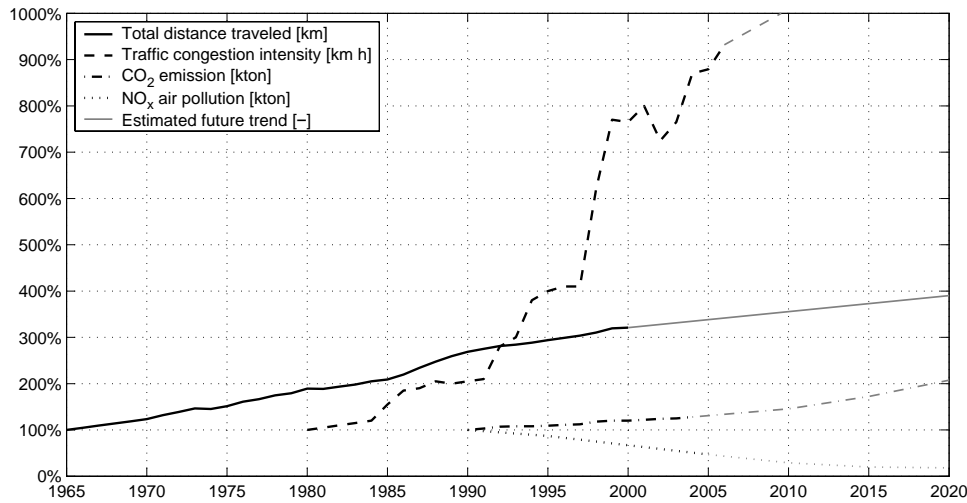


Figure 1.1: Indexed trends in mobility [108], traffic congestion [224], greenhouse gas emissions [155], and NO_x air pollution [49] for the EU. Future estimated trends are also shown. The trend for other hazardous exhaust emissions is similar to that for NO_x.

world, road safety is thus one of the largest public health issues, but has not yet been fully recognized as such. The external costs amount to 5 % of the GDP, and the global cost is over US\$ 500 billion, even without attaching a cost to psychological stress and suffering experienced by survivors and their families.

Accidents are an important source of traffic jams [224], and vice versa, traffic congestion increases the accident risk [203]. Congestion also aggravates air pollution [211], such that a joint solution to the above-mentioned problems is advisable. Figure 1.1 shows that these problems will continue to grow, unless there is new commitment to prevention. Government policies, *e.g.*, by the European Commission [49], are therefore directed at implementing stringent regulations and supporting research and development programs. The EU goals are to halve the number of accidents by 2010 [50], cut back hazardous emissions [53], reduce global warming related CO₂ emissions from passenger cars [54], and reduce congestion in order to retain economic competitiveness [50].

Examples of policy measures for fighting congestion are traffic management systems and road tolling. Emission standards (*e.g.*, the European Euro I–IV standards) define limits on exhaust emissions, and have been quite successful in reducing hazardous emissions, as shown in Figure 1.1. However, CO₂ emissions are likely to increase by 2 % per year due to the large increase in road traffic.

Although the total number of vehicle accidents is rising, Figure 1.2 shows that the relative level of vehicle safety, expressed per distance traveled, has improved. This increase in passenger safety is primarily due to advances in *passive safety*, such as increased crash-worthiness and the introduction of seat belts and airbags. Unfortunately, many possibilities for improvement in passive safety have now been exhausted, and the safety potential seems to have flattened out. Advanced technology is expected to play an important role in further alleviating the societal problems of road traffic.

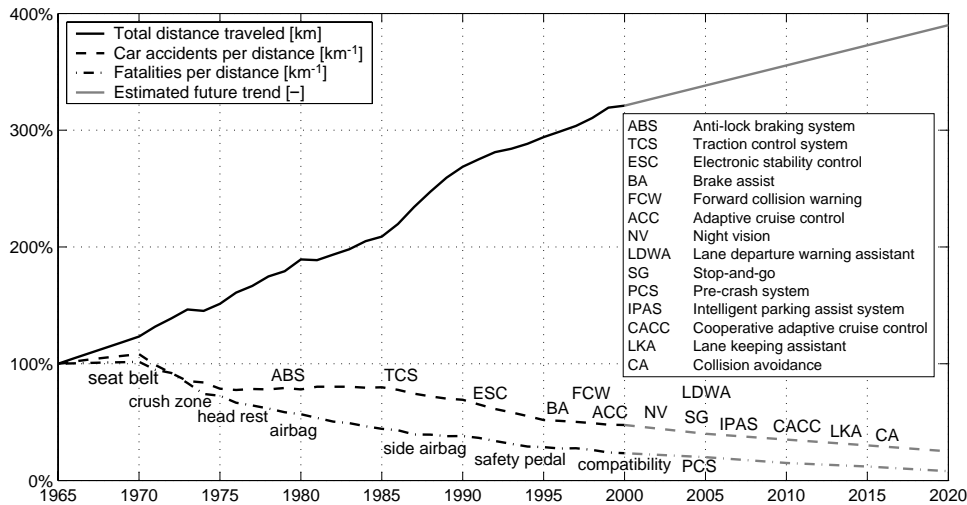


Figure 1.2: Total number of road accidents and fatalities per total distance traveled, indexed on 1965 data for the EU [108]. In addition, the graph shows when passive safety systems (which reduce fatalities in case of an accident) and active safety systems (which assist in avoiding an accident) have first been introduced (or are expected to be introduced) to the market, as well as the expected safety potential of these systems [89, 118, 274].

1.2 Advance of intelligent transportation systems

Due to advances in micro-electronics, there is an increasing trend in the automotive industry to integrate sensors, actuators, microcomputers, and information processing for the engine, drivetrain, suspension, and brake systems. The first steps in this direction were digitally controlled combustion engines and cruise control systems in the 1970s. This was paralleled by the development of electronic braking and advanced vehicle handling functions in order to increase vehicle safety by means of *active safety*, as shown in the historical overview by Isermann *et al.* [115]. These *driver assistance systems*, such as anti-lock braking system (ABS), traction control system, brake assist, and electronic stability control (ESC), offer possibilities for improving traffic safety by assisting the driver in his driving task. Especially the mass introduction of ABS [26] and ESC [151] have contributed to the steady decrease in the relative number of road fatalities in the 1990s, as shown in Figure 1.2.

In addition, several active front steering systems and even four-wheel steering are being introduced to improve lateral vehicle stability. Furthermore, active roll stabilization and (semi-)active suspension systems have been implemented in vehicles to resolve the conflict between driver comfort and vehicle handling [111]. To exploit the full potential of controlling longitudinal, lateral, and vertical vehicle dynamics, there is a clear need towards the use of drive-by-wire technology for vehicle actuators, such as electronic throttle control, electromechanical braking, electronic transmission control, and steer-by-wire [115].

For further improvement in accessibility, sustainability, and safety, there is also a huge potential within the field of *intelligent transportation systems* (ITS). ITS is a collective term that covers technology supported services for participants in the traffic system, such as pedestrians, drivers, fleet managers, road authorities, and public transport operators. ITS

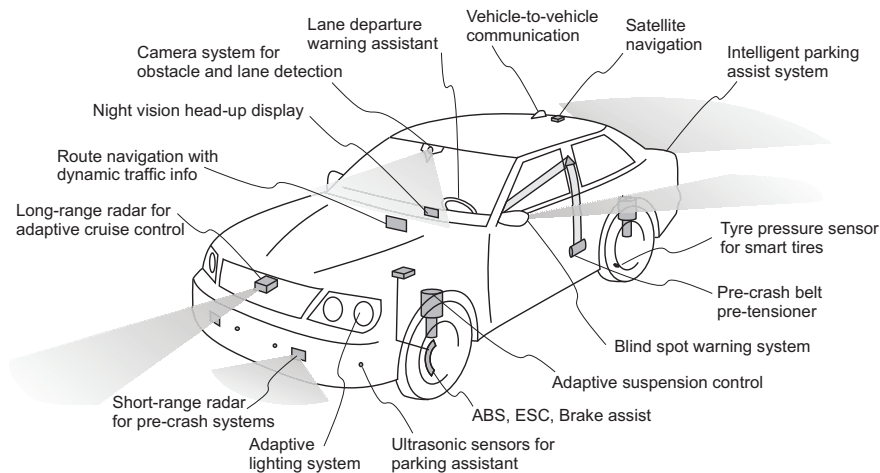


Figure 1.3: Illustration of various ADASs and sensor systems.

incorporates a wide range of commercial (*e.g.*, logistics systems, electronic toll collection) and public (*e.g.*, traffic control, public transport systems) services using information and communication technology. It also covers *advanced driver assistance systems* (ADASs), which have the potential to significantly reduce the number and severity of road accidents. An ADAS is defined as a vehicle control system that uses environment sensors to improve driving comfort and/or traffic safety by assisting the driver in recognizing and reacting to potentially dangerous traffic situations.

Research into these intelligent vehicle systems has been initiated by car manufacturers, research organizations, and government-industry partnerships in the 1980s, most notably in Europe within the PROMETHEUS project [263] and subsequent European Framework Programmes; in the USA within the California PATH (Partners for Advanced Transit and Highway) program [214] and the Intelligent Vehicle Initiative [94]; and in Japan within the Advanced Safety Vehicle program [164]. As a result of these research programs, most car manufacturers currently have a wide range of intelligent vehicle systems on the market that intend to increase driving comfort and traffic safety. Figure 1.3 provides an illustration of some of the systems that are currently available to customers or that are under development. As reviewed in more detail by Shladover [216], Tsugawa [244], and Bishop [18], these systems can be categorized according to the hierarchical levels of the driving task:

- *Driver information systems* aim to support the driver on the *strategic level* of the driving task. Although the driver remains in full control, his situation awareness can be increased by additional information [249]. Examples are advanced route navigation, traffic sign recognition, night vision, and adaptive light control.
- *Driver warning systems* support the driver on the *maneuvering level* of the driving task and actively warn the driver of a potential danger. The driver can then take appropriate actions in order to mitigate or completely avoid the dangerous event. Examples are parking assistant, lane departure warning assistant, blind spot warning, intersection collision warning, driver drowsiness warning, and forward collision warning systems [176].

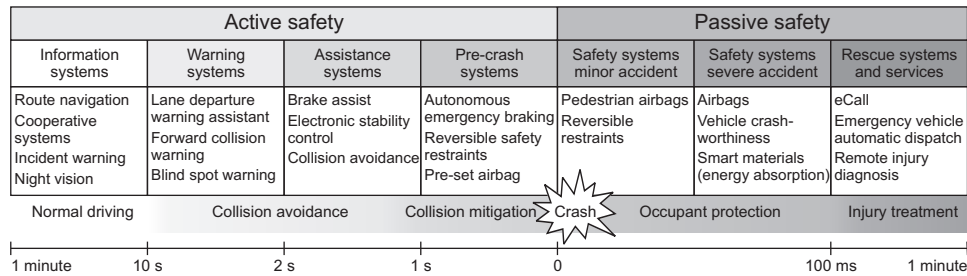


Figure 1.4: Integrated safety encompasses both active and passive safety systems that cooperate in the time interval directly before, during, and directly after an incident.

- *Intervening systems* provide active support to the driver on the *control* level of the driving task. These systems are able to take over longitudinal or lateral control of the vehicle and perform parts of the driving task automatically, such as lane keeping [116], intelligent speed adaptation [30], and adaptive cruise control (ACC) [265].
- *Integrated passive and active safety systems* include all systems (including some of the above) that work towards vehicle safety in a cooperative manner [62], as illustrated in Figure 1.4. During normal driving the driver is informed about the traffic environment and any potential danger. If there is no driver reaction to this information, a warning may be issued to indicate the need for corrective measures. As the time to collision decreases to several seconds, systems may actively assist in avoiding the collision. In addition to conventional passive safety systems that are activated *during* the crash, the crash severity can be mitigated by deploying safety measures *before* the actual collision occurs. For example, autonomous emergency braking during the last second before the collision can reduce the impact speed. Other examples of these *pre-crash systems* are seat belt pre-tensioners (at 500 ms) [240] and optimization of the airbag deployment (at 10 ms) [166]. After an incident, emergency response services may also be alerted through an automatic emergency call system (eCall) [256].
- *Fully automated systems* are the next step beyond driver assistance, and operate without a human driver in the control loop. Automated highway systems use fully automated passenger cars that drive autonomously on dedicated lanes, and have been a topic of intense research, most notably by the California PATH program [215]. Automated highway systems are expected to significantly benefit traffic safety and throughput [212], but are currently not considered for short-term introduction [219].

As human error is a contributing factor in more than 90 % of all accidents [243], it is likely that ADASs have great potential for reducing the number of traffic accidents. According to several surveys (e.g., [89, 118, 274] and the references therein), ADASs may even prevent up to 40 % of traffic accidents, depending on the type of ADAS, the accident scenario, and the traffic environment. Apart from an increase in traffic safety and driving comfort, it is expected that ADASs will also reduce traffic congestion [22], exhaust gas emissions [109], and CO₂ emissions [213]. On the other hand, a few studies have shown that some ADASs only have a limited, or even negative, effect on traffic safety or traffic flow [247, 279]. It seems that the penetration level, as well as the implementation policy, are determining factors for the impact of ADASs on road safety and traffic efficiency [47].

1.3 Problem formulation

Despite potential benefits for accessibility, sustainability, and safety, the current market penetration of ADASs is limited [93]. Main challenges in this respect are liability exposure [260], regulatory issues [246], the cost-benefit ratio [52], and customer acceptance [16]. This relates to several problems faced by manufacturers in the development of ADASs.

Problem 1: Lack of quantitative requirements and evaluation criteria

Field-operational tests have shown that customers will only accept a system that intervenes in their driving task if automatic control functions (*e.g.*, automatic braking) offer superior performance and blend in naturally with their own driving behavior [58]. Moreover, drivers will expect an ADAS to meet stringent requirements regarding *dependability*, which is defined as the trustworthiness of a *safety-critical* computer system to deliver the desired performance in spite of operating conditions, disturbances, and failure modes [141]. The demand for dependability naturally increases with increasing automation of the vehicle's driving task, since the driver must be able to fully depend on the ADAS. Failure of an automatic safety system simply cannot be tolerated, *e.g.*, automatic deployment of a pre-crash belt pre-tensioner should be executed if, and only if, a crash is imminent and unavoidable. However, it is often difficult to define quantitative requirements for performance and dependability, since there are currently no specific standards for ADASs available. An ADAS is therefore often designed 'as well as possible', and a qualitative assessment is carried out to judge whether the system is 'good' enough to be introduced on the market.

Problem 2: Increasing complexity of the system and its environment

An associated problem is that dependability is difficult to guarantee with the increasing complexity of automatic vehicle control systems. Today, software and electronics account for up to 40% of the total development costs of a passenger car [195]. This figure is expected to rise even further, as the increasing trend towards automatic safety systems implies a growing number of sensors, actuators, and control systems implemented in embedded systems. In addition, the integration of several ADASs and the interaction with other vehicle control systems creates ever more *complex* systems. A system is said to be complex when the level of interactions reaches the point where they cannot be thoroughly planned, understood, anticipated, and guarded against [147].

The interaction with a human driver introduces even more complexity, since, unlike a skilled pilot in an aircraft, a driver has not been trained to use an ADAS and can behave unpredictably to warnings or automatic intervention. Drivers will also differ considerably in reaction time and skill in operating the system. This puts high demands on the control system and how it communicates with the driver in an unobtrusive way. Furthermore, the interaction with other road users in the traffic environment adds yet another level of complexity to the design of these systems. All these interactions may introduce unforeseen situations, which further complicates the design of the ADAS.

Problem 3: Challenges in robust and fault-tolerant control

The complex system design must be able to handle all kinds of disturbances, environmental conditions, and other operating conditions. The controller must therefore be extremely

robust against these perturbations. Furthermore, faults may occur in sensors, actuators, communication systems, and computer hardware and software, which may degrade the system dependability. One aspect that contributes to dependability is *fault-tolerant* behavior, *i.e.*, to maintain operational behavior in spite of faults [115]. Advanced fault-tolerant control techniques for sensor and actuator systems have already been applied in the aerospace industry for years [182], but are relatively new to the field of automotive mechatronic systems [115]. Fault-tolerant control in the unstructured environment of an automobile, subjected to a complex set of disturbances and failure modes, is therefore a challenging issue.

Problem 4: Difficulties in the validation of complex systems

In order to show that an ADAS exhibits adequate performance with high dependability, *validation* of a safety-critical control system is meant to determine whether disturbances and faults are handled correctly. Therefore, the ADAS must be tested for the wide variety of complex traffic situations that the system should recognize and manage. ADASs are therefore usually tested by test drives on a test track, but it is impossible to cover all operating scenarios by exhaustive testing. It is especially difficult to accurately reproduce the conditions and failure modes under which the control system operates. Due to disturbances, test drive results may also be unreliable and difficult to analyze. Extensive safety precautions must be taken to ensure the safety of test drivers and prototypes during safety-critical scenarios. Consequently, manufacturers are facing longer development times, whereas they have an increasing desire for a shorter time-to-market of their products. Obviously, the costs for the validation process increase. It is estimated that testing and evaluation of an automotive control system may take up to 50 % of the total development costs [100].

Problem 5: Lack of reliable simulation tools and methods

To facilitate the design and validation, simulation tools are increasingly employed in the automotive industry. Simulations are relatively cheap and are ideal for analyzing and understanding the physical relations between the vehicle's components and the performance of the control system. However, the correctness of the simulation results is strongly dependent on the simulation model, the simulation parameters, and the interpretation by the simulation engineer. Simulation of an ADAS control system, integrated with environment sensors and actuators, is especially difficult because of the complexity that is involved in modeling sensors, actuators, vehicle dynamics, and the traffic environment. This underlines the need to take model uncertainty into account during the design and simulation process of the control system. On the other hand, during evaluation of the product, test personnel often do not have detailed modeling insight in the system. It is therefore desired to have the capability to assess the performance and dependability of black box systems, but still be able to derive potential critical scenarios from the design phase. Vice versa, it is advisable to have an easy feedback of validation results to the control design phase. This interaction between design and validation has motivated a trend in the automotive industry towards the use of *model-based design*, where the simulated control system is used throughout the development cycle, and directly generates the real-time code [245]. A prerequisite for model-based design is the availability of accurate simulation models and the capability to validate these models. However, with regard to ADAS development, these tools and methods are currently not adequate. Not only the design, but especially the validation of ADASs, thus requires a growing effort in the development process of these systems.

1.4 Objectives and scope of this thesis

In view of the above problem formulation, the main objective of this thesis can be defined:

To develop an efficient model-based methodological framework and associated tools for the design and validation of advanced driver assistance systems, such that the performance and dependability of these systems can be guaranteed.

The problem formulation and thesis objective can be further decomposed into four research objectives that are covered within the scope of this thesis:

1. Define quantitative requirements for performance and dependability.
2. Improve the performance and dependability of ADASs.
3. Develop tools and methods for model-based design and validation.
4. Make the validation process more efficient.

Objective 1: Define quantitative requirements for performance and dependability

The ADAS performance is directly related to the impact that the system has on the societal issues mentioned earlier. This thesis specifically focusses on performance measures related to traffic safety, driver comfort, and traffic flow. An important starting point is the definition of quantitative performance measures and corresponding evaluation criteria for different ADAS types. In addition, quantitative measures for dependability should be defined, in terms of reliability, safety, and fault-tolerance. The operating conditions, for which an ADAS should meet these requirements, must be investigated. Especially microscopic traffic modeling with respect to inter-vehicle behavior is of interest, since this constitutes the environment, in which the ADAS operates. In addition, disturbances and failure modes that perturb the vehicle and its components must be identified.

Objective 2: Improve the performance and dependability of ADASs

Under the influence of these perturbations, the controller must exhibit adequate and dependable performance. Fault tolerance is an important system attribute for increasing that dependability. Especially in case of sensor faults, the operation of an ADAS is safety-critical, due to the dependence on reliable information on the traffic environment. Methods for fault detection and subsequent fault-tolerant control must therefore be investigated and extended for application in ADASs.

Objective 3: Develop tools and methods for model-based design and validation

Obviously, it must be validated whether the performance and dependability of ADASs meet their requirements. The Netherlands Organisation for Applied Scientific Research (TNO) [175] has recognized the importance of this validation issue. TNO has developed specific tools, such as the modeling environment PreScan that enables reliable simulation of sensors, vehicle dynamics, and traffic scenarios [250]. In addition, TNO has built a laboratory for efficient and accurate testing of ADASs using vehicle hardware-in-the-loop (VeHIL) simulations [74]. In the VeHIL laboratory, an ADAS-equipped vehicle can be tested in a

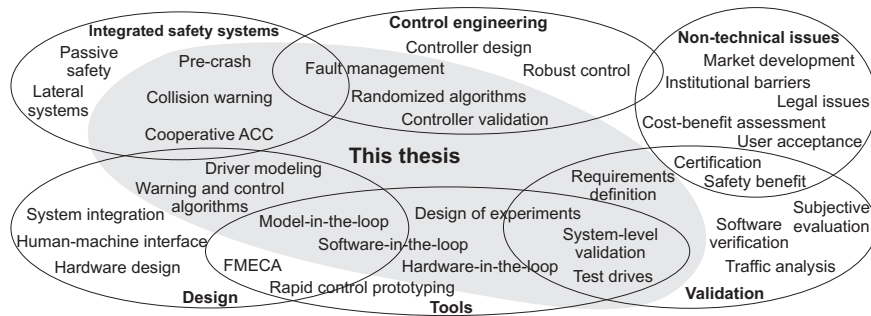


Figure 1.5: Scope of this thesis within the relevant research areas.

near-realistic traffic environment. PreScan and VeHIL have the potential to significantly accelerate the development process of ADASs. An important objective is therefore to integrate the use of PreScan and VeHIL with existing test tools, such as test drives, and complement the current automotive development process.

The representativeness of these simulation tools, compared to a real traffic environment, must be investigated in terms of traffic behavior, sensor signals, and vehicle dynamics. Any model uncertainty should be incorporated in the test program and associated with the reliability of the test results. This illustrates the need for accurate simulation models and the capability for validation of these models. Another objective is to investigate the relation between PreScan simulations and VeHIL experiments for the purpose of this model validation.

Objective 4: Make the validation process more efficient

In order to make the validation process more *efficient*, the use of the validation tools must be optimized to reduce the number of tests and simulation time, and consequently reduce development time and costs. More specifically, it is of interest to know how many PreScan simulations and VeHIL tests are required to guarantee the performance and dependability of an ADAS, and which scenarios should be tested. The main objective is therefore to derive a suitable test program that sufficiently (and efficiently) covers a representative subset of the entire *parameter set* (the combined set of operating conditions and failure modes), given a specific ADAS and specific requirements. Important issues in increasing this efficiency are efficient sampling techniques, obtaining a representative parameter subset, and obtaining some kind of guarantee on the outcome, even without exact knowledge on the control structure. Alternatively, with insight in the control structure available, the validation results should be used for improving the control system design.

Scope of this thesis

As the field of research of intelligent vehicle systems is very broad, it makes sense to limit the scope of these objectives, as illustrated in Figure 1.5. First of all, the main focus of the methodological framework is on the *validation* methods within the development cycle. The actual *design* of ADASs is limited to the design of a fault-tolerant control system for cooperative ACC. Furthermore, the ADASs are limited to longitudinal control and warning systems, which are validated on system level. Lower-level verification and higher-level

certification procedures are only briefly discussed. Related non-technical issues, such as user acceptance and legal issues are not covered in this thesis, nor are the subjective elements of driver behavior. However, we do invest in the development of a driver model for microscopic traffic simulation, which is relevant for ADAS validation. For the simulation strategy, use is made of randomized algorithms and design of experiments theory.

1.5 Outline of this thesis

The research objectives presented in the previous section will be investigated using the structure that is illustrated in Figure 1.6.

Chapter 2 starts with an overview of state-of-the-art ADASs, as well as a functional decomposition into enabling technologies. A number of challenges that currently exist within the ADAS development process are identified. These are further explored by defining requirements and evaluation criteria for performance and dependability. The operating conditions for ADASs are also investigated by defining a parameter set of generic traffic scenarios using a microscopic traffic modeling approach. In addition, disturbances and failure modes that may perturb the controller are identified.

Chapter 3 presents methods to increase the dependability of control systems by fault management. State-of-the-art methods for fault detection and fault-tolerant control are briefly summarized. These methods are extended to a system for sensor fault detection and fault-tolerant state estimation, which is implemented and tested in a demonstrator vehicle.

The challenges in design and validation are further explored in **Chapter 4**. A review of state-of-the-art tools for ADAS development illustrates the problems that are associated with these tools. The simulation environment PreScan and the VeHIL laboratory are presented as a solution to these problems. VeHIL and PreScan are extended and integrated with the existing tools to complement the ADAS development process. Several examples illustrate the use of VeHIL for specific purposes, such as sensor calibration and system validation.

Chapter 5 reviews and extends state-of-the-art validation methods to form the methodological framework for this thesis. After a summary of randomized algorithms theory, a new efficient validation method using adaptive importance sampling is presented. The added value and validity of this method is illustrated with a case study involving a simple linear controller. Finally, the method is integrated with the test tools (PreScan, VeHIL, and test drives) in a methodological framework for the design and validation of ADASs.

In order to demonstrate the practical relevance of the methodological framework for various types of ADASs and various phases of the development process, it will be applied to several case studies. **Chapter 6** presents a case study involving a driver information and warning system for safe speed and safe distance. The focus of validation is on driver-related aspects, such as the timeliness and dependability of the warning function.

Chapter 7 presents a new algorithm for cooperative ACC in order to increase the driver comfort, traffic flow, and safety of automatic car-following. This control system also incorporates the fault-tolerant state estimation that was presented in Chapter 3. This control system is validated for string stability, and dependability issues, such as fault-tolerance.

Due to their safety-critical nature, pre-crash systems require a different approach for validation. In **Chapter 8** the methodological framework is therefore modified and applied to a pre-crash system. The focus in this chapter is on verification of the sensor processing software and validation of the dependability of the integrated system. The development

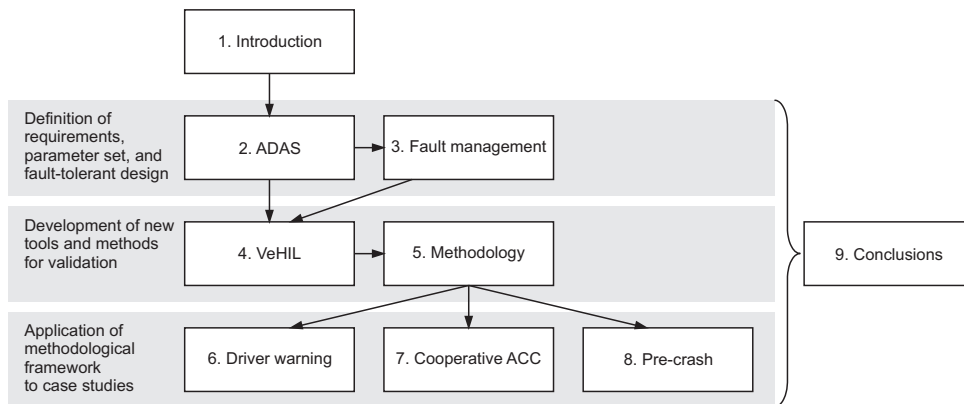


Figure 1.6: Overview and structure of this thesis.

process is also extended to benefit assessment for these types of safety systems.

In addition to the conclusions at the end of each chapter, **Chapter 9** presents the main conclusions of this thesis. The results of this thesis are critically reviewed, and both the added value and the limitations of the methodology are presented. This chapter also provides recommendations for future research.

1.6 Contributions of this thesis

This section summarizes the contributions of this thesis to the state of the art.

Definition of quantitative requirements and parameter set

An important result is the development of quantitative requirements and criteria for the validation of ADASs in terms of performance and dependability. Furthermore, the parameter set to which an ADAS is subjected is identified. For this purpose, a microscopic traffic model, including new models for inter-vehicle behavior, is developed. In addition, an accidentology analysis is performed to identify the most critical pre-crash scenarios.

Efficient tools and methods using adaptive importance sampling

An efficient methodological framework for the design and validation of ADASs is developed on the basis of the defined requirements and parameter set. The randomized algorithm that has been developed provides a guarantee on the required number of tests to obtain a desired accuracy and confidence level on the outcome of the validation process. Although this number is an upper bound and not minimal, the number of required tests is still considerably smaller than in a test program using conventional test methods.

The tools that are used in this methodology have been developed by TNO, while this thesis has focussed on extension and integration of these tools within the methodological framework. The simulation environment PreScan is extended for application of the methodological framework. Furthermore, the unique VeHIL laboratory is presented, which is specifically developed for testing ADASs. This facility is extended to include methods for

fault injection to analyze the capability of fault management systems. Due to the integration of these tools in the model-based automotive development process, VeHIL facilitates the transition from simulations to test drives. These test drives can be performed with a higher confidence and less risk, when the ADAS has already been thoroughly tested in VeHIL.

Fault-tolerant state estimation and control

A new method for fault-tolerant control of a cooperative ACC system is developed. State information from the host vehicle and target vehicles, obtained from environment sensors and vehicle-to-vehicle communication are fused to obtain a reliable estimate of the relative motion between vehicles. This analytical redundancy enables smooth and safe traffic flow in a string of vehicles, even in the event of safety-critical faults. The benefit of this fault-tolerant controller is shown with test results from an experimental demonstrator setup.

Practical relevance of the results

The development of tools and methods to address the challenges that are currently faced by the automotive industry has significant practical relevance. The methodology and associated tools allow car manufacturers and suppliers of ADASs to develop these systems in a more efficient way, and quantitatively validate the performance and dependability. In this way, it is expected that application of this methodology leads to lower development costs and a shorter time-to-market of ADASs. The added value for the automotive industry is illustrated with several prototype demonstrators. Furthermore, with the recent introduction of ADASs on the market, there is a desire for unambiguous evaluation criteria and standardized homologation procedures. A new experimental setup is therefore proposed for pre-crash testing in VeHIL, complementary to existing homologation procedures.

1.7 Publications by the author

Most of the material presented in this thesis has been previously presented at conferences, published in peer-reviewed journals, or submitted for future publication. The survey of ADASs and review of challenges in the development process in Chapter 2 is based on [82]. The material on forward collision warning algorithms in this chapter has been published in [168] and the traffic scenario modeling for ADASs is based on [241]. The method for fault detection in Chapter 3 has been submitted for publication in a journal [79].

The review of methods and tools for development of ADASs, as well as the VeHIL laboratory in Chapter 4 have been presented in [74]. The methods for fault injection have been published in [73, 83]. The application of VeHIL in various phases of the development process has been published in [44, 136]. The probabilistic approach for validation of ADASs in Chapter 5 has been presented at several conferences [71, 75, 84] and published in [76]. In addition, the extension of this approach to adaptive importance sampling was presented at a conference [77] and is also submitted for publication in a journal [78].

The application of this methodology to the case study in Chapter 6 has been published in [233]. The case study in Chapter 7 has been presented at a conference [190] and submitted for publication in a journal [80]. The PreScan simulations of the pre-crash system in Chapter 8 have been published in [85] and the MADYMO simulations in [146]. The corresponding VeHIL test results have been published in [81, 145].

Chapter 2

Advanced driver assistance systems

The previous chapter has outlined the scope of this thesis, which is further explored in this chapter. The goal of this chapter is to define quantitative requirements for design and validation of advanced driver assistance systems (ADASs), as was stated in Objective 1 on page 8. Sections 2.1 and 2.2 review the state of the art in ADASs and their enabling technologies, including the challenges this involves for controller design and validation. The resulting requirements for performance and dependability are formulated in Section 2.3. The ADAS should meet these requirements for a wide range of traffic scenarios, disturbances, and faults, subsequently discussed in Sections 2.4 and 2.5. Finally, Section 2.6 summarizes the chapter.

2.1 Overview of the state of the art in ADASs

Figure 2.1 provides an overview of the various types of ADASs and their future deployment paths. This roadmap distinguishes between systems for longitudinal and lateral assistance [46], and their gradual integration towards fully autonomous driving [217]. Since this field of research is very broad, this chapter only gives a brief summary of the ADAS control algorithms and technologies that are relevant for this thesis. The state-of-the-art overview is restricted to systems for longitudinal assistance, since these are quite generic from a functional point of view and are expected to have a significant safety potential [89]. For a more comprehensive overview the reader is referred to several references [18, 47, 228].

2.1.1 Forward collision warning systems

Forward collision warning systems (FCW) target a major crash type: rear-end collisions. FCW systems warn drivers of an imminent collision, such that the driver can take appropriate corrective actions in order to mitigate or to completely avoid a collision. Since its introduction by Nissan in 1988 [181], several FCW systems have been developed. The warning algorithms use distance-based, deceleration-based and time-based measures, as presented in [57, 126]. These algorithms are briefly summarized below.

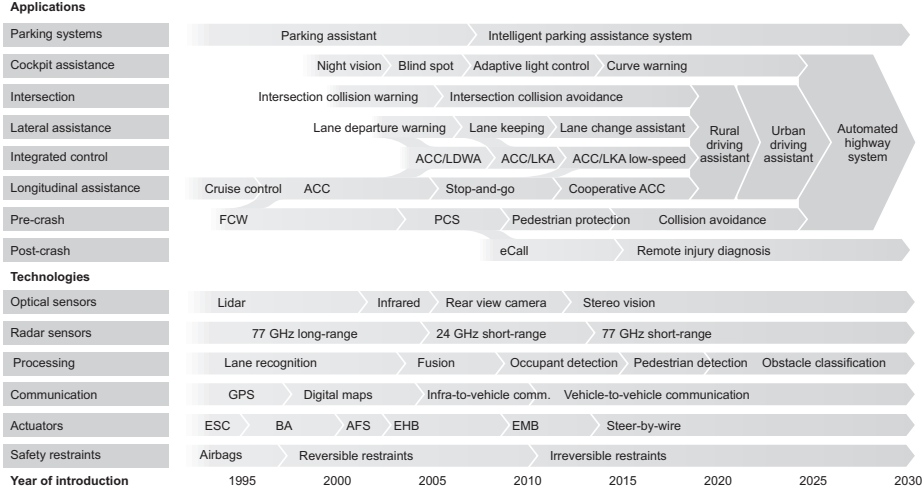


Figure 2.1: ADAS roadmap for applications and technologies. Refer to the Glossary and Figure 1.2 for an explanation of the terminology.

Critical braking distance

Most collision warning algorithms issue a warning when the current range to an object (the headway) x_r is less than the *critical warning distance* s_{warn} [28, 133, 176, 274]. The warning then allows the driver to stop or approach no closer than a designated distance s_0 behind a stopped or decelerating target vehicle. The calculation of s_{warn} is illustrated by Figure 2.2, which shows a host vehicle i and target vehicle $i-1$, each with state $\mathbf{x}_i = [x_i \ v_i \ a_i]^T$, where x_i is position, v_i velocity, and a_i acceleration of vehicle i . The figure also indicates the vehicle length L_i , the headway $x_{r,i} = x_{i-1} - x_i - L_i$, and the relative velocity $v_{r,i} = v_{i-1} - v_i$. Assuming constant values for the host vehicle maximum braking capability $a_{i,\text{max}}$, and for the accelerations a_{i-1} and a_i , three possible scenarios should be distinguished, due to the discontinuity in acceleration that occurs at standstill [264]: (a) an initially moving target vehicle stops prior to the host vehicle, (b) the target is still in motion when a potential collision would occur, or (c) the target is initially stopped. To determine which case applies, the target vehicle stopping time $t_{\text{stop},1}$ and host vehicle stopping time $t_{\text{stop},2}$ are calculated:

$$t_{\text{stop},1} = -\frac{v_1}{a_1}, \quad (2.1)$$

$$t_{\text{stop},2} = t_{\text{reac}} - \frac{v_2 + a_2 t_{\text{reac}}}{a_{2,\text{max}}}, \quad (2.2)$$

where t_{reac} is the driver reaction time. For each of these cases the critical warning distance s_{warn} can then be calculated as follows:

$$s_{\text{warn}} = \begin{cases} v_2 t_{\text{reac}} + \frac{1}{2} a_2 t_{\text{reac}}^2 - \frac{(v_2 + a_2 t_{\text{reac}})^2}{2a_{2,\text{max}}} + \frac{v_1^2}{2a_1} + s_0 & \text{if } t_{\text{stop},1} \leq t_{\text{stop},2} \text{ and } t_{\text{stop},1} \neq 0, \\ -v_1 t_{\text{reac}} - \frac{1}{2} a_1 t_{\text{reac}}^2 + \frac{(v_1 + a_1 t_{\text{reac}})^2}{2(a_1 - a_{2,\text{max}})} + s_0 & \text{if } t_{\text{stop},1} > t_{\text{stop},2} \text{ or } t_{\text{stop},1} = 0, \end{cases} \quad (2.3)$$

with $a_r = a_1 - a_2$ the relative acceleration.

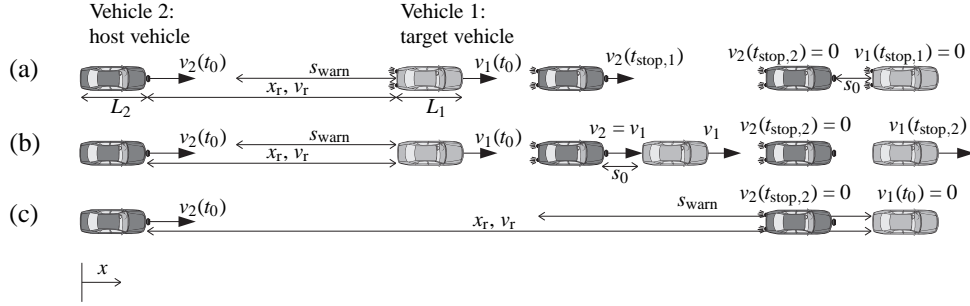


Figure 2.2: Development over time of possible scenario configurations for forward collision warning: (a) an initially moving target vehicle (light) stops prior to the host vehicle (dark), (b) the target is still in motion when a potential collision would occur, or (c) the target is initially stopped. The length of the bold arrows gives an indication of the absolute vehicle speed.

Using field-operational test drives with subject drivers, warning algorithm (2.3) has been empirically optimized for use in several commercial FCW systems [41, 208] to give warnings corresponding with natural driver behavior. Unfortunately, this type of algorithm will also warn drivers when they intend to perform a late lane-change maneuver, since the algorithm only considers longitudinal vehicle motion. As a result, drivers may find the system conservative and become less sensitive to future warnings. This illustrates the need for appropriate warnings to the driver.

Required deceleration

Instead of specifying a critical braking distance, a warning can be issued when a threshold of maximum braking capability a_{\min} is crossed by the required deceleration a_{ref} to prevent a collision [168]. To avoid a collision by a safety margin s_0 , a_{ref} is given by

$$a_{\text{ref}} = \begin{cases} \frac{a_1 v_2^2}{2a_1 (t_{\text{reac}} v_2 - x_r + s_0) + v_1^2} & \text{if } t_{\text{stop},1} \leq t_{\text{stop},2} \text{ and } t_{\text{stop},1} \neq 0, \\ \frac{a_1 (x_r - s_0) - \frac{1}{2} v_r^2}{t_{\text{reac}} (\frac{1}{2} t_{\text{reac}} a_1 + v_r) + x_r - s_0} & \text{if } t_{\text{stop},1} > t_{\text{stop},2} \text{ or } t_{\text{stop},1} = 0. \end{cases} \quad (2.4)$$

The critical warning distance algorithm (2.3) that was developed by the US National Highway Traffic Safety Administration (NHTSA) [28] has been transformed into an algorithm for required deceleration within the CAMP project [132]. An empirically optimized value for a_{ref} was established as

$$a_{\text{ref}} = \begin{cases} 0.685a_1 - 0.086(v_1 + a_2 t_{\text{reac}}) - 1.617 & \text{if } t_{\text{stop},1} \leq t_{\text{reac}}, \\ 0.685a_1 - 0.086(v_r + a_r t_{\text{reac}}) - 0.833 & \text{if } t_{\text{stop},1} > t_{\text{reac}}. \end{cases} \quad (2.5)$$

Time-to-collision

As investigated in [132], a less conservative algorithm can be obtained by using a time-based measure, such as the frequently used time-to-collision (TTC), denoted t_{TTC} . The TTC refers

to the time it would take for a collision to occur, considering the current vehicle motion. Assuming constant velocities, the basic TTC measure t_{TTC} is defined by

$$t_{\text{TTC}} = -\frac{x_{\text{r}}}{v_{\text{r}}}. \quad (2.6)$$

Alternatively, the acceleration of both vehicles can be included in the calculation, such that the TTC can be solved from $\frac{-v_{\text{r}}(t) \pm \sqrt{v_{\text{r}}(t)^2 - 2x_{\text{r}}(t)a_{\text{r}}(t)}}{a_{\text{r}}(t)}$. However, since the target vehicle acceleration may change unexpectedly and is difficult to estimate, this method produces a noisy signal. We therefore use (2.6) as the TTC measure.

A warning is issued when the TTC in (2.6) is below a threshold that allows the driver enough time to react. However, if the TTC is less than the driver reaction time t_{reac} , the driver will not benefit from warnings, and the system may activate automatic emergency braking [208]. The corresponding *critical braking distance* s_{br} is then

$$s_{\text{br}} = v_{\text{r}}t_{\text{reac}} + \frac{1}{2}a_2t_{\text{reac}}^2. \quad (2.7)$$

Time headway

Another time-based measure is the time headway

$$t_{\text{h}} = \frac{x_{\text{r}}}{v_2}, \quad (2.8)$$

where a warning can be issued when the time gap to a preceding vehicle is below a certain threshold value, *e.g.*, when a driver is *tailgating* behind another vehicle [28]. However, the criticality of a traffic situation is not always correlated to a specific time headway.

Probabilistic approach

Since the deterministic algorithms described above are based on a fixed structure, disagreement between the human drivers and the system response always exists. Deterministic algorithms may therefore be too conservative in some, or not responsive enough in other scenarios. Instead, a probabilistic approach can be used, where sensor data is provided to a Kalman filter [119]. This allows to estimate the probability for an evasive maneuver, and issue a warning at a certain collision probability. Alternatively, the collision probability can be estimated by an on-line Monte Carlo simulation of possible scenarios [131].

Safety potential

Evaluation of FCW systems with field-operational tests has demonstrated the safety potential and increased driver comfort [133]. However, the safety advantage of FCW is still limited, because of the little time available for the driver to react in case of a real collision threat. On the other hand, a more conservative warning algorithm will cause driver nuisance. Therefore, the market penetration of FCW systems in passenger cars has been low [93]. In the truck market on the other hand, FCW has been more successful [266], since professional drivers can be trained to respond more appropriately to a warning and tolerate any false alarms.

To further increase traffic safety, FCW can be extended with automatic braking to form a collision avoidance system, which only intervenes after the driver ignores a warning, as

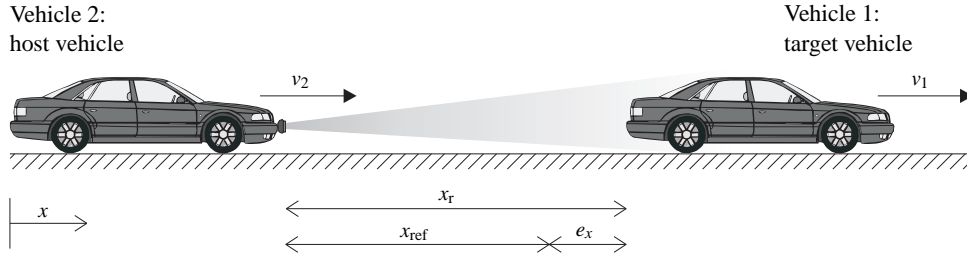


Figure 2.3: Schematic representation of an ACC system, which controls the ACC-equipped vehicle 2 to follow the target vehicle 1 with equal velocity at a desired safe distance.

in algorithm (2.7). However, in case a driver attempts an avoidance maneuver himself, automatic braking may interfere with the control by the driver. A more promising approach is therefore to integrate a warning function with adaptive cruise control, as discussed next.

2.1.2 Adaptive cruise control systems

Mitsubishi introduced adaptive cruise control (ACC) in 1995 as the first automatically intervening ADAS [259], soon followed by other car manufacturers [194, 209, 265]. There is a vast amount of literature on ACC systems (see [125] and the references therein), and only a short outline of the basic control configuration is given here.

Velocity control

The function of ACC is to maintain the cruise control velocity $v_{\text{ref, CC}}$, selected by the driver, where the desired acceleration a_{ref} is given by a simple proportional controller

$$a_{\text{ref}} = K_{\text{CC}}(v_{\text{ref, CC}} - v_2), \quad (2.9)$$

with $K_{\text{CC}} > 0$ the proportional gain for the velocity error $v_{\text{ref, CC}} - v_2$.

Distance control

ACC maintains the cruise control velocity, unless another vehicle in front is judged to be too close considering the current speed. As illustrated in Figure 2.3, the control objective of the ACC is to reach the same speed as the preceding vehicle at a desired safe distance x_{ref} . The ACC is designed to respond like an attentive human driver, in order to regulate both the spacing error $e_x = x_{\text{ref}} - x_r$ and the speed tracking error $e_v = v_{\text{ref}} - v_r$ to zero. This is usually accomplished by the combination of a control law that computes a reference acceleration a_{ref} and an acceleration controller that tracks this reference acceleration as well as possible. The desired acceleration a_{ref} is usually given by proportional feedback control of e_x and e_v :

$$a_{\text{ref}} = -K_2 e_v - K_1 e_x, \quad K_1, K_2 > 0. \quad (2.10)$$

Since the desired relative velocity v_{ref} is obviously equal to zero, (2.10) is rewritten as

$$a_{\text{ref}} = K_2 v_r + K_1 (x_r - x_{\text{ref}}). \quad (2.11)$$

The desired distance x_{ref} is often proportional to a time headway t_h , which can usually be set manually between 1 and 2 seconds. The feedback gains K_1 and K_2 are a nonlinear function of a number of state variables (usually x_r , v_2 , v_r , and t_h), tuned in order to achieve a natural following behavior [271]. This results in the constant time gap control law:

$$a_{\text{ref}} = K_2 v_r + K_1 (x_r - t_h v_2 - s_0). \quad (2.12)$$

In practice, gain scheduling is used to tune K_1 and K_2 for specific scenarios, and (2.12) is implemented using PID techniques [269]. In case a_{ref} cannot be achieved automatically, a warning is issued to the driver, indicating that the driver should overrule the brake control.

Advantages and challenges

A field-operational test by NHTSA of an integrated ACC/FCW system [176] has demonstrated increased levels of comfort and safety when using ACC systems in a highway environment. On the other hand, several authors (*e.g.*, [161]) show that ACC may have negative influence on traffic throughput, due to the increased headway between vehicles. Despite its safety potential, ACC is still marketed as a comfort system rather than a safety system, and the braking capacity is usually limited to around -3 m/s^2 [106]. This prevents a harsh deceleration, should a false alarm occur. ACC can therefore not be regarded as a collision avoidance system. In emergency situations, the driver has to take over, and remains fully responsible for the vehicle maneuvering. Depending on the tuning of (2.12), there is a significant difference in performance between various systems currently available on the market. In some situations the driving behavior does not seem natural, which may result in potentially dangerous situations, as shown by a recent benchmark of several commercially available ACC systems [201].

This also relates to the current range of ACC sensors of around 200m, which is not sufficient for timely recognition of an oncoming traffic jam or other potential danger further ahead. Furthermore, potential obstacles may not be detected by the environment sensor due to road curves or when they are blocked by other vehicles. Moreover, in high-density traffic (with short time headways), the ACC-advised time gap is relatively large, which can lead to many cut-ins and can be perceived as uncomfortable.

ACC is turned off automatically when the velocity drops below a threshold value (about 30 km/h). ACC systems are therefore extended with a stop-and-go function (sometimes referred to as ‘low-speed ACC’). Stop-and-go offers the possibility for automated longitudinal control in cluttered environments, such as traffic jams and city environments, and has recently been introduced on the market by Nissan [268]. Stop-and-go places high requirements on the capabilities of the sensorial platform, because of the complexity of low-speed traffic (*e.g.*, pedestrians, bicycles, short distances, crossings) [188, 248, 270].

2.1.3 Pre-crash systems

In some circumstances a collision cannot be avoided, even if the driver is assisted by an ADAS. In this case it makes sense for an intelligent vehicle control system to at least *mitigate* the collision. The objective of a pre-crash system (PCS) is to improve the effectiveness of safety restraints and subsequently minimize injury severity, by activating them *before* a collision occurs, in case this collision is assessed as imminent and unavoidable.

The concept of using radar for pre-crash sensing was already investigated in the 1970s [90], but was hampered by the low performance and reliability of early radar systems. The first PCS on the market therefore only used information from vehicle state sensors (such as ABS and roll-over detection sensors) [202]. But with the arrival of low-cost short-range environment sensors, PCSs have become an intensive research topic [4, 5] and the first radar-based PCS has been introduced on the market in 2003 [97]. A PCS usually includes a collision warning function, *e.g.*, by using vibration of the seat belt [128, 134].

When a collision becomes unavoidable, reversible *passive* restraints are used to mitigate the collision severity. An example is the activation of the belt pre-tensioner, such that the driver is brought into a better position for the airbag to work effectively. Pre-crash sensor data can also be used to optimize the activation of irreversible passive restraints, such as the deployment characteristics of an airbag (multi-stage, trigger time, etc.) [166].

The effects of the collision can also be mitigated by *active* safety measures, such as brake assist functions [65] or even autonomous emergency braking [258]. Such a system for collision mitigation by braking will not actually *prevent* the collision, but will still reduce the impact speed considerably. Furthermore, the activation of external airbags for pedestrian protection is currently under investigation [158].

PCSs are especially safety-critical systems to test, since this requires the recreation of a collision to validate the operation of the PCS. Furthermore, the safety benefit must be evaluated for a wide range of collision events, which places high demands on the accuracy and update rate of the sensor. In the next section we will therefore discuss some technical challenges that still lie ahead.

2.2 Enabling technologies for ADASs and their challenges

The previous section has shown the potential of ADASs for greatly increasing both driving comfort and vehicle safety. However, these systems rely on several enabling technologies that can be identified from the functional decomposition in Figure 2.4:

- *Sensors* to monitor the vehicle state and the surrounding traffic environment.
- *Controllers* to perform sensor fusion, obstacle identification, obstacle tracking, decision logic, and generation of command signals for the warning display and actuators.
- *Human-machine interface* to provide the interface between the decision logic of the ADAS and the driver.
- *Actuators*, such as engine and brake system, to carry out the desired vehicle motion.
- *Communication* to allow information exchange between all functional blocks.

In the following subsections these technologies will be discussed, as far as they are relevant for the thesis. The integration of these functions with mechanical and electronic components, together with machine intelligence is defined with the term *mechatronics* [112]. An ADAS typically is a mechatronic system in the sense that it integrates vehicle dynamics, engine, brake system, embedded control systems, and sensors, and interacts with several other vehicle control systems. The drawback of this increasing integration and interaction is that ever more complex control systems are created, which causes some considerable technical challenges [218].

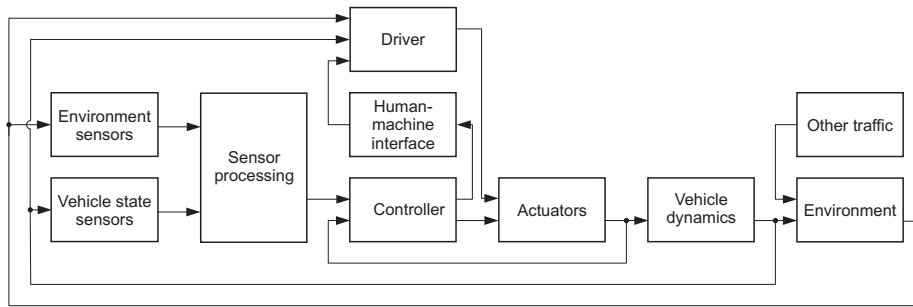


Figure 2.4: General functional decomposition of an ADAS.

2.2.1 Environment sensor technology

Modern passenger cars are already equipped with sensors for measuring the host vehicle state, such as wheel speed sensors (velocity v), gyroscopes (yaw rate $\dot{\psi}$), and accelerometers (acceleration a). In addition, ADASs use environment sensors to detect other road users and road infrastructure. State-of-the-art sensors are based on a variety of technologies, such as radar, lidar, machine vision, infrared, ultrasonic, and range imaging. This section gives only a brief overview of the basic characteristics and some technical challenges that are relevant for this thesis. For a more comprehensive overview of sensor technology, the reader is referred to [139, 227].

Radar systems

ADAS sensors are often based on radio detection and ranging (radar) technology. Pulse radars send short pulses of radar energy, and the reflections of these pulses are received by an antenna, such that the traveling time of the pulse, proportional to the traveled distance, is determined. The relative velocity is determined using the Doppler effect. In order to improve detection accuracy, continuous wave radars transmit a constant beam of radar energy by an illuminator. The reflected radar wave, received by a separate antenna, has a frequency that is slightly higher or lower than the original radar frequency due to the Doppler shift, from which the relative velocity to an object can be calculated. A frequency-modulated continuous wave (FMCW) radar is continuously transmitting a wave of which the frequency is modulated. By measuring the difference in frequency between the transmitted and received signal, the time between transmission and reception of the wave can be determined, and subsequently the range to the object. The reader is referred to [1] for a more detailed overview of automotive radar systems.

Long-range radar systems for use in ACC systems usually operate at the 77 GHz frequency. They have a range up to 200 m, but a poor short range detection, as well as a small beamwidth and limited angular resolution. For increased performance in stop-and-go applications, short-range radar has been developed, based on 24 GHz technology. For further improvement of short-range accuracy ultra-wide band technology can be used, for which the 24 GHz frequency has been allocated by the European Commission in 2004 [51]. Ultra-wide band technology will also be used at the 77 GHz frequency after 2013 [221].

The main trade-off in radar design occurs with respect to the beamwidth. Too wide of an azimuth or elevation angle will result in ‘ghost targets’ from side barriers or overhead

structures, whereas a small beamwidth will not allow to detect vehicles that cut-in in front of the host vehicle or vehicles further ahead when driving in curves. Another limitation of radar is that standing objects are difficult to detect, since they often disappear in the background noise. This effect may cause missed detections, and objects with zero speed are therefore usually neglected.

Lidar systems

Light detection and ranging (lidar) is a relatively inexpensive technology for obstacle detection. It is based on a scanning laser beam in the horizontal field that gives the distance and angle to all the objects in this plane. The main advantage is that it has a wide field-of-view and high accuracy [66]. On the other hand, lidar is more susceptible to adverse weather conditions than radar, although recent improvements have overcome this problem [102].

Vision systems

Vision systems are also increasingly used in ADASs. The most prominent application is lane detection for lateral control and warning systems. In addition, vision systems can be used for obstacle detection and classification, which is also interesting for longitudinal control systems, such as ACC.

The main challenge in machine vision is to provide reliable and real-time detection. Although this problem has been solved for lane detection, the computational load for obstacle detection and classification for safety-critical applications currently exceeds the power of existing electronic control units. The interested reader is referred to the work by Vlacic *et al.* [257] for a comprehensive overview of the various approaches for detection and classification of road markings and obstacles.

Vehicle navigation

As opposed to monitoring the *relative* position to objects, vehicles also require information on the *absolute* position for navigation purposes. The global positioning system (GPS) is a satellite navigation system that provides the global position of the vehicle. In combination with a digital map, GPS has been used in vehicle navigation systems for over a decade now.

Unfortunately, the standard accuracy and reliability of GPS (around 15 m) is insufficient for more advanced vehicle control applications based on navigation, such as curve speed warning [249]. Differential GPS (DGPS) uses differential corrections from base stations in the neighborhood and provides an accuracy of around 3 m at a sample rate of 1 Hz. However, the use of satellite navigation alone will remain unreliable for safety-critical ADASs [8].

2.2.2 Signal processing and control engineering

Environment sensors always exhibit measurement noise, and are limited in their capability to distinguish real targets (stopped vehicles, dropped loads) from infrastructure elements, such as overhead bridges, traffic signs, and guard rails. This complexity of the traffic environment, combined with a huge number of possible traffic scenarios, increases the demands on the sensorial platform. Post-processing is therefore necessary to perform tasks, such as state estimation, obstacle detection, obstacle tracking, and path prediction.

Sensor fusion

Since the requirements for safety-critical ADASs can often not be met by a single sensor, sensor fusion techniques are applied to merge the information provided by different sensors. The goal of sensor fusion in the current context is to provide a reliable environment model that is as robust as possible with respect to measurement noise, faults, and disturbances. Fusion strategies vary widely depending on the type of sensors and specific application. It has been a heavily researched topic, both within the industry, universities [64], research organizations [3], and EU funded projects, such as CARSENSE [140], ProFusion [227], and CHAMELEON [67]. A typical approach is to combine radar with vision, since radar has high distance accuracy, but low lateral resolution, whereas video has a low distance accuracy with high lateral resolution.

Modeling

The first step in the design of an ADAS controller is to develop an appropriate mathematical model for the vehicle dynamics, sensors and actuators. A reliable simulation model is necessary for synthesizing a controller with adequate performance. On the other hand, the model should also be not too complex to retain modeling insight and simulation speed. Modeling therefore always involves a trade-off between speed and transparency versus complexity and reliability. This issue will be further addressed in Chapters 4 and 5.

Control algorithms

Command signals for the warning display and actuators have to be calculated from the processed sensor data. Considering the wide range of applications, there is a large variety in control strategies and algorithms to achieve specific control objectives. The reader is referred to the books by Jurgen [124–126], and the references therein, for a comprehensive overview on the topic. Nevertheless, in the subsequent chapters we will extend and implement some of the algorithms discussed in the previous section to illustrate the development process of ADASs.

Challenges in control

Shladover [218] has identified many challenges in control that still lie ahead on the road to vehicle automation. These research needs are primarily related to the overall safety of the control system design, and include fault detection and isolation, fault-tolerant control, sensor fusion, and validation of the system safety. Correspondingly, the main challenge in control design addressed in this thesis is that the control system should exhibit a high level of dependability, which will be further discussed in Section 2.3.5.

2.2.3 Human-centered design

ADASs are inherently human-centered, since the control system must work together with the driver to achieve a safe and comfortable driving experience. The human-machine interface (HMI) provides the layer between the control system and the driver, as illustrated in Figure 2.4. The HMI is a very important factor in the ADAS design, because the HMI must be interpreted by the driver in a natural way, and should not reduce driver vigilance or

cause compensation effects. These human factors issues involve detailed research outside the scope of this thesis, and HMI system design and validation will therefore not be further investigated. See the work by Bishop [18] for an overview of the topic. On the other hand, the interaction with the human driver adds extra complexity to the system design, an issue that cannot be neglected in the system validation process. Unfortunately, it is often difficult to validate the HMI operation against qualitative criteria, because of the psychological elements of HMI interpretation.

2.2.4 Drive-by-wire technology

Similar to the rise of fly-by-wire technology in the aerospace industry, the development of drive-by-wire technology offers new possibilities for intelligent vehicle control, as summarized by Isermann *et al.* [115]. Drive-by-wire systems allow the driver to generate electrical commands for computer-controlled electromechanical actuators. The important advantage of brake-by-wire, throttle-by-wire, and steer-by-wire systems is that they offer higher performance in terms of reaction time, accuracy, flexibility, and adaptive control capabilities. Furthermore, the lower weight and volume improves systems integration. However, an important issue with drive-by-wire systems without mechanical backup is the dependability of the system [68].

2.2.5 Automotive communication systems

Reliable in-vehicle networking between sensors, processors, and actuators is an important prerequisite for dependable operation of an ADAS. A failure of the in-vehicle network poses a potential hazard for the vehicle, its passengers, and the vehicle's environment [12], and makes it a safety-critical factor in the ADAS design. Since its introduction in the 1980s, the controller area network (CAN) has established itself as a robust network for distributed control applications in the vehicle. CAN is essentially an event-triggered protocol, which means that its behavior is not fully deterministic in a real-time sense. However, in order to implement safety-critical drive-by-wire systems, the system behavior must be available in real-time. Several time-triggered protocols have therefore been developed in recent years for use in ADASs [210].

Another emerging technology is wireless communication between intelligent vehicle systems and infrastructure elements. Wireless communication with external services, such as travel information, infotainment, or electronic toll collection can significantly increase driver convenience. Furthermore, communication with the roadside through infrastructure-to-vehicle communication can provide locally relevant data, with applications such as curve speed warning and road condition warning. Dedicated short range communication is typically used as a communication protocol for infrastructure-to-vehicle communication.

Wireless communication also extends to vehicle-to-vehicle communication (VVC) between neighboring vehicles. VVC can be used for information functions, such as warning for local weather conditions (*e.g.*, fog, icy road) or oncoming traffic hazards (*e.g.*, accidents, traffic jams). In addition, VVC can be used to extend the sensor field-of-view beyond the directly preceding vehicle. VVC is expected to have great potential for increasing comfort and safety by augmenting environment sensors. In contrast to the command-response approach by dedicated short range communication, several EU projects (*e.g.*, CarTALK [167]) have explored the potential of ad-hoc local networks for VVC. The advantage is that ad-hoc net-

working does not need a central controller. However, challenges in VVC are the reliability of the local ad-hoc networks, and the accuracy of the transmitted data.

2.2.6 Summary of technological challenges

From this section we can summarize that an ADAS is a complex and safety-critical mechatronic system. The most important challenge in this respect is to achieve dependable environment sensing. Furthermore, satellite navigation and wireless communication networks are prone to disturbances, and need a high level of reliability or redundant systems for state estimation. Control algorithms need to be able to handle these disturbances to provide a safe and comfortable driving experience for the user. In addition, vehicle subsystems, in-vehicle networking, and actuators must remain operational, even in the presence of disturbances and faults, which underlines the need for fault tolerance of the complete system. Furthermore, the driver remains responsible for the vehicle maneuvering, whether or not he/she is supported by an ADAS. Therefore, designing an ADAS that can accommodate a wide range of human characteristics is non-trivial. These challenges mean that stringent requirements are demanded for ADASs regarding comfort, performance, and dependability.

2.3 Requirements and evaluation criteria for ADASs

The end user typically desires comfort, performance, and dependability for as low a price as possible. It is the task of the manufacturer to design and validate a product that meets these requirements. The European project RESPONSE [206] has focused on establishing an industry-wide methodology for the design and validation of ADASs, which is legally robust and valid from an engineering and human factors perspective. A Code of Practice has been developed that includes a set of requirements and procedures for the manufacturer to follow for ADAS design and testing [205]. Unfortunately, this Code of Practice and emerging ISO standards [106, 107] can only describe the development process at an abstract level. They do not provide quantitative requirements or validation procedures for ADASs. Therefore, the user requirements are extended in this section to quantitative measures that will be used in the remainder of this thesis.

2.3.1 Abstraction of the ADAS control configuration

The ADAS system architecture of Figure 2.4 can be replaced by the block diagram of Figure 2.5, where the ADAS is modeled as a nonlinear multivariable plant, consisting of the vehicle dynamics G_1 and actuator dynamics G_2 . The vehicle state is represented by the vector \mathbf{x} , with individual elements x_i representing position, velocity, acceleration, etc. The traffic environment can be regarded as a disturbance $\mathbf{d}_{\text{traffic}}$, such that we obtain the output state \mathbf{y} , representing the relative motion between the host and target vehicles.

Sensors measure the system output $\mathbf{y} = [\mathbf{y}_1^T \quad \mathbf{y}_2^T]^T$, consisting of variables representing the state of the vehicle \mathbf{y}_2 (using vehicle state sensors) and the state of the traffic environment \mathbf{y}_1 (using environment sensors). The input to the controller \mathbf{K}_1 is the error between the reference trajectory \mathbf{x}_{ref} and the measured output \mathbf{z}_1 , where $\mathbf{z}_1 = \mathbf{y}_1 + \mathbf{v}_{s,1} + \mathbf{f}_{s,1}$ with measurement noise $\mathbf{v}_{s,1}$ and sensor faults $\mathbf{f}_{s,1}$. The control objective is to manipulate the reference value for the actuator response \mathbf{u}_{ref} such that the control error $\mathbf{e} = \mathbf{y}_1 - \mathbf{x}_{\text{ref}}$ remains small in spite of disturbances $\mathbf{d}_{\text{traffic}}$ and vehicle faults \mathbf{f}_1 .

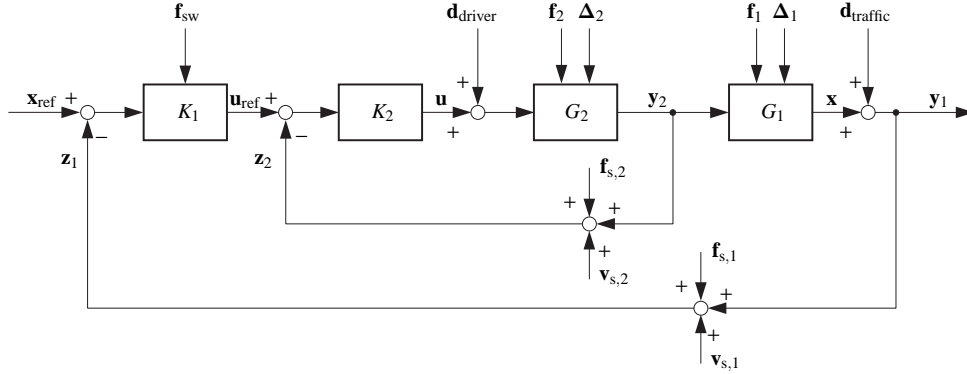


Figure 2.5: Block diagram of an ADAS under the influence of disturbances, faults, and model uncertainty.

Because an ADAS usually separates a high-level control law from low-level actuator control, we can define a similar control objective for the actuator controller \mathbf{K}_2 . Assuming acceleration control, the input to \mathbf{K}_2 is the error between the desired actuator response \mathbf{u}_{ref} and the measured vehicle state \mathbf{z}_2 , where $\mathbf{z}_2 = \mathbf{y}_2 + \mathbf{v}_{s,2} + \mathbf{f}_{s,2}$ with measurement noise $\mathbf{v}_{s,2}$ and measurement faults $\mathbf{f}_{s,2}$. The servo control objective is to manipulate the command signals \mathbf{u} for the vehicle actuators (e.g., throttle and brake in the case of an ACC system), such that desired response \mathbf{u}_{ref} is tracked in spite of disturbances by the driver $\mathbf{d}_{\text{driver}}$ and actuator faults \mathbf{f}_2 . Since there is always model uncertainty involved when modeling and simulating a system, the designed controllers are perturbed by model uncertainty in the actuator behavior Δ_2 and in the vehicle dynamics Δ_1 .

We can now use the signals \mathbf{u} , \mathbf{x} , and \mathbf{y} to define quantitative requirements for validation of the controllers \mathbf{K}_1 and \mathbf{K}_2 . We define the m -dimensional performance vector $\boldsymbol{\rho} = [\rho_1, \dots, \rho_k, \dots, \rho_m]^T$, where $\boldsymbol{\rho}$ consists of both Boolean and continuous signals. Unfortunately, the value of an individual measure ρ_k does not provide any information on how ‘good’ it is. Therefore, we also define evaluation criteria by a threshold value γ_k for every ρ_k , such that $\boldsymbol{\gamma} = [\gamma_1, \dots, \gamma_k, \dots, \gamma_m]^T$, where we define the k -th measure as satisfied when $\rho_k < \gamma_k$, unless indicated otherwise.

2.3.2 Stability requirements for the control system

Although collision warning systems simply involve feedforward control, many other ADASs, such as ACC, involve a feedback controller. An important requirement for any feedback control system is that it exhibits asymptotically stable behavior. We will therefore present stability requirements, both for the individual vehicles and for a string of vehicles.

Definition 2.1 Individual vehicle stability A vehicle following control law is said to provide individual vehicle stability for vehicle i if the spacing error e_i of the host vehicle i to the target vehicle $i-1$, as well as the speed tracking error \dot{e}_i converge to zero when the preceding vehicle is operating at constant speed:

$$a_{i-1} \rightarrow 0 \Rightarrow e_i \rightarrow 0 \text{ and } \dot{e}_i \rightarrow 0, \quad (2.13)$$

where $e_i = x_{\text{ref},i} - x_{r,i}$ and its derivative $\dot{e}_i = v_{\text{ref},i} - v_{r,i} = -v_{r,i}$. \square

Example 2.1 Individual vehicle stability. Since the ACC objective is to regulate e_i and \dot{e}_i to zero, the state of an ACC-equipped vehicle i is chosen as $\mathbf{x}_i = [x_{r,i} \quad v_{r,i} \quad v_i]^T$. Assuming a linear time-invariant system for the moment, we can use the state space representation

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{d} \quad (2.14)$$

with \mathbf{A} the system matrix, \mathbf{B} the input matrix, and \mathbf{E} the disturbance matrix. The ACC system is then described by

$$\dot{\mathbf{x}}_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{x}_i + \begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix} a_i + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} a_{i-1}, \quad (2.15)$$

where the acceleration of the host vehicle a_i is the control input, and the acceleration of the target vehicle a_{i-1} forms the disturbance to the system. The control law (2.12) can be rewritten as $\mathbf{u} = \mathbf{K}\mathbf{x}$ with $\mathbf{K} = [K_1 \quad K_2 \quad -K_1 t_h]$. Assuming that the acceleration can be controlled instantaneously, such that $\ddot{x}_i = a_{\text{ref},i}$, stability can be analyzed by computing the eigenvalues of the matrix $\mathbf{A} + \mathbf{B}\mathbf{K}$. From this follows that the poles lie in the left-half plane for all values $K_1 > 0$ and $K_2 > 0$ [197]. It is therefore concluded that the two-vehicle ACC system is always stable for any controller of the type (2.12).

Automated vehicles must not only exhibit stability individually, but the concept of stability can also be extended to multiple vehicles.

Definition 2.2 String stability is defined as an asymptotic stability of the system which is composed of a finite number of interconnected subsystems (*i.e.*, single vehicles) with the same or similar dynamics [230]. Practically, string stability requires that spacing errors and speed tracking errors are guaranteed not to amplify as they propagate upstream in a string of vehicles. A prerequisite for string stability is that each individual closed-loop vehicle controller is asymptotically stable, as defined in Definition 2.1. \square

String stability is an important prerequisite for obtaining safe longitudinal behavior without the risk of a collision. It was already recognized as an important control problem for automated highway systems in the 1960s [148]. With increasing market penetration of ACC, the chances increase that several ACC-equipped vehicles will drive behind each other. String stability for ACC has therefore been investigated by many authors [55, 150, 231, 272], and only a brief review of the theory is presented here.

A desirable characteristic for attenuation of propagating spacing errors is

$$\|e_i\|_\infty \leq \|e_{i-1}\|_\infty, \quad (2.16)$$

where e_{i-1} is the input spacing error (of the target vehicle $i-1$) and $e_i(t) = e_{i-1}(t) * \hat{h}(t)$ is the output spacing error (of the ACC-equipped host vehicle i) with $\hat{h}(t)$ the impulse response of the host. We therefore require that

$$\|e_{i-1} * \hat{h}\|_\infty \leq \|e_i\|_\infty. \quad (2.17)$$

Since $\|\hat{h} * e_i\|_\infty \leq \|\hat{h}\|_1 \|e_i\|_\infty$, a sufficient requirement for string stability is that

$$\|\hat{h}\|_1 \|e_{i-1}\|_\infty < \|e_{i-1}\|_\infty \iff \|\hat{h}\|_1 \leq 1. \quad (2.18)$$

For control system design and analysis it is however easier to use a transfer function instead of an impulse response. Swaroop [230] provides the proof that (2.18) can be replaced by the following two conditions:

$$\|H(s)\|_{\infty} = \left\| \frac{E_i(s)}{E_{i-1}(s)} \right\|_{\infty} \leq 1, \quad (2.19)$$

$$\hat{h}(t) > 0, \quad (2.20)$$

where $H(s)$ and $E(s)$ are the Laplace transforms of $\hat{h}(t)$ and $e(t)$, respectively. For practical reasons, condition (2.19) is selected as the performance measure for string stability. The constant headway controller is not string-stable, as shown next.

Example 2.2 String stability for constant headway policy. Consider the constant spacing control law (2.10) for vehicle i with spacing error $e_x = e_i$ and the speed tracking error its derivative: $e_v = \dot{e}_i$. The control law can then be rewritten as

$$a_{\text{ref}} = -K_1 e_i - K_2 \dot{e}_i. \quad (2.21)$$

Assuming the acceleration of the vehicle can be instantaneously controlled, such that $\ddot{x}_i = a_{\text{ref}}$, and implementing the same controller in the preceding vehicle $i-1$, we get:

$$\ddot{e}_i = \ddot{x}_i - \ddot{x}_{i-1} = -K_1 e_i - K_2 \dot{e}_i + K_1 e_{i-1} + K_2 \dot{e}_{i-1}, \quad (2.22)$$

which leads to the following closed-loop error dynamics:

$$\ddot{e}_i + K_2 \dot{e}_i + K_1 e_i = K_2 \dot{e}_{i-1} + K_1 e_{i-1}. \quad (2.23)$$

The propagation of spacing errors along the vehicle string is then described by

$$H(s) = \frac{E_i(s)}{E_{i-1}(s)} = \frac{K_2 s + K_1}{s^2 + K_2 s + K_1}. \quad (2.24)$$

All positive values of K_1 and K_2 guarantee that the spacing error of the i -th vehicle converges to zero when the spacing error of the $(i-1)$ -th vehicle is zero, thereby ensuring individual vehicle stability. However, there are no positive values K_1 and K_2 for which the magnitude of $H(s)$ in (2.24) can be guaranteed to be less than unity. Requiring that:

$$|H(j\omega)| = \left| \frac{K_2 j\omega + K_1}{-\omega^2 + K_2 j\omega + K_1} \right| = \frac{\sqrt{K_2^2 \omega^2 + K_1^2}}{\sqrt{K_2^2 \omega^2 + (K_1 - \omega^2)^2}} \leq 1, \quad (2.25)$$

and subsequent simplification leads to the requirement that

$$(K_1 - \omega^2)^2 > K_1^2, \quad (2.26)$$

which shows that spacing errors are magnified for frequencies below $\sqrt{2K_1}$. As a result

$$|H(j\omega)| = \begin{cases} = 1 & \text{if } \omega = 0, \\ < 1 & \text{if } \omega^2 > 2K_1, \\ \geq 1 & \text{otherwise.} \end{cases} \quad (2.27)$$

Unfortunately, low frequencies are the frequencies of interest in traffic dynamics, so this controller will result in an unstable string of vehicles.

The analysis of platoon string stability is complicated by the presence of nonlinearities. Although the string stability analysis in Example 2.2 is based on linear models, it is valid for small deviations around the operating conditions. In contrast to the constant spacing control law (2.10), it will be shown in Chapter 7 that the constant time headway control law (2.12) can be made string-stable, provided the time headway t_h is large enough. However, several field-operational tests have confirmed that in practice ACC is string-unstable due to sensor noise and nonlinear ACC algorithms [14, 58, 59]. Alternative vehicle control laws are therefore necessary to obtain a string-stable behavior, which will be further investigated in Chapter 7.

2.3.3 Functional performance requirements

Apart from stability, a controller must also exhibit adequate performance to satisfy the user. The performance of ADASs is often assessed by analyzing the added value in terms of safety potential and customer satisfaction, as discussed previously. This evaluation is based on analysis of questionnaires and expert panels [159]. The traffic impact of ADASs is expressed in *macroscopic* performance measures, such as traffic throughput, travel time, and average speed [98], and is usually investigated with simulations based on vehicles with similar characteristics and similar ADAS controller types [232]. Unfortunately, these measures do not give information on the performance and comfort of individual vehicles. We therefore need to define *microscopic* measures to define the *functional* performance of ADASs.

Relatively few references discuss the need for microscopic performance requirements and quantitative evaluation criteria. NHTSA has developed some requirements for FCW [133], but focusses on a description of traffic scenarios, rather than on quantitative criteria that are suitable for benchmarking. Some performance requirements have been discussed in [59] and [267], illustrated with an example of a simple ACC controller. For easy benchmarking it is desired to extend these requirements to quantitative performance measures. Controller performance is usually characterized in terms of response time, tracking performance, robustness, and control effort.

The system response time t_{delay} is the time difference between the start of a scenario (*e.g.*, a cut-in maneuver or an emergency braking maneuver of the preceding vehicle) and the first reaction of the ADAS.

Tracking error in terms of headway distance overshoot and undershoot of the host vehicle with respect to the target vehicle is an undesirable feature, not only for string stability, but also for performance perception by the driver. In the frequency domain, it can be defined by the \mathcal{H}_∞ norm of the transfer function from the headway distance command x_{ref} to the actual headway distance x_r , which gives the index

$$\left\| \frac{x_r(s)}{x_{\text{ref}}(s)} \right\|_\infty. \quad (2.28)$$

Although individual vehicle stability is guaranteed for any ACC controller of the type (2.12), the error will never settle exactly to zero, due to disturbances and faults in the range measurement. The resulting tracking error may cause undesirable acceleration and deceleration, which degrades passenger comfort and the perceived performance of the system. This robustness measure can be defined as the ℓ_2 norm of the sensor noise to the vehicle acceleration during following

$$\left\| \frac{a(s)}{x_{r,m}(s) - x_r(s)} \right\|_2. \quad (2.29)$$

There is always a trade-off between control performance and control effort, because of energy considerations and constraints on the actuator input. In the time domain, the control effort is defined in terms of the weighted ℓ_2 norm (*i.e.*, the root mean square (RMS) value) of throttle and brake usage:

$$\|u(t)\|_{\text{pow}}. \quad (2.30)$$

Often a number of the above-mentioned performance measures are combined in a cost function to synthesize an optimal controller, (see *e.g.*, [150]) where the tracking performance and control effort are used. However, these optimal controllers do not grasp the subtleties and nonlinearities of human driving behavior. This underlines the need to quantify desirable human driving behavior.

2.3.4 Driver interaction requirements

Driver comfort includes a wide range of issues, such as ease of operation, easy HMI interpretation, nuisance level, and physical comfort, which can often only be measured in a subjective way. In this thesis we quantify physical comfort in terms of the RMS value of the longitudinal acceleration:

$$\|a_{\text{long}}(t)\|_{\text{pow}}. \quad (2.31)$$

However, we also need a quantitative criterion to evaluate the performance of an automatic longitudinal vehicle controller compared to typical human driving behavior. We therefore define the RMS value of the error between the actual distance to a target vehicle x_r and the distance given by a reference model $x_{r,\text{ref}}$:

$$\|x_{r,\text{ref}}(t) - x_r(t)\|_{\text{pow}}. \quad (2.32)$$

The problem of realistic driver modeling is a heavily investigated research topic, and various types of driver models are available for traffic simulation, as shown in the historical review by Brackstone and McDonald [24]. One of the first simple approaches to driver modeling was the car-following model by Helly [95]. The acceleration is given by

$$a_i(t) = C_2 v_{r,i}(t - t_{\text{reac}}) + C_1 (x_{r,i}(t - t_{\text{reac}}) - x_{\text{ref},i}(t)), \quad (2.33)$$

where the desired distance is

$$x_{\text{ref},i}(t) = s_0 + t_{\text{h}} v_i(t - t_{\text{reac}}). \quad (2.34)$$

This model relates the driver's acceleration of vehicle i to the distance $x_{r,i}$ and relative velocity $v_{r,i}$ to a target vehicle, taking into account a driver reaction time t_{reac} . The desired following distance depends on the host velocity v_i and constant values C_1 and C_2 . Note the similarity between this model and the structure of the general ACC control law (2.12). Over the years, the complexity of driver models has increased. To include the driver reaction to n vehicles further ahead, multi-anticipative models have been developed [99], such as the Generalized Helly model

$$a_i(t) = \sum_{j=i-1}^n C_{2,j} v_{r,i,j}(t - t_{\text{reac}}) + \sum_{j=i-1}^n C_{1,j} (x_{r,i,j}(t - t_{\text{reac}}) - x_{\text{ref},i,j}). \quad (2.35)$$

However, benchmarking studies have revealed that even the most complex traffic flow models were not able to predict car-following behavior more accurately than the relatively simple ones [25]. Nevertheless, Lee and Peng [142] showed that a good approximation of single-leader car-following is provided by the Gipps model.

The driver model developed by Gipps [86] belongs to an often used class of *safety distance models*. The objective of the first part of this model is to maintain a desired velocity v_{ref} assuming there is no preceding vehicle, the so-called *free-flow* mode. The aim of the second part is to maintain a safe following distance, *i.e.*, the *car-following* mode. Both parts give the velocity of the vehicle after a driver reaction time t_{reac} , and the resulting velocity is the minimum of the two:

$$v_i(t) = \min \left\{ v_i(t-t_{\text{reac}}) + 2.5a_{\text{max}}t_{\text{reac}} \left(1 - \frac{v_i(t-t_{\text{reac}})}{v_{\text{ref},i}} \right) \sqrt{0.025 + \frac{v_i(t-t_{\text{reac}})}{v_{\text{ref},i}}}, \right. \\ \left. \hat{a}_{\text{min},i}t_{\text{reac}} + \sqrt{(\hat{a}_{\text{min},i}t_{\text{reac}})^2 - \hat{a}_{\text{min},i} \left(2x_r(t-t_{\text{reac}}) - v_i(t-t_{\text{reac}})t_{\text{reac}} - \frac{v_{i-1}^2(t-t_{\text{reac}})}{\hat{a}_{\text{min},i-1}} \right)} \right\}. \quad (2.36)$$

The Gipps model has the advantage that it has a low degree of freedom. That is, only a small number of parameters needs to be calibrated: the maximum allowable acceleration a_{max} , the estimate of the most severe braking deceleration that the driver of vehicle i wishes to undertake $\hat{a}_{\text{min},i}$, and the ratio between the desired velocity and the initial velocity:

$$\lambda_{v,i} = \frac{v_{\text{ref},i}}{v_i(0)}. \quad (2.37)$$

Because of the simplicity and wide applicability of the Gipps model (2.36), we will use it here to compute the reference distance $x_{r,\text{ref}}$ in (2.32) to judge the car-following behavior of a longitudinal control system.

For collision warning systems we must also evaluate the *timeliness* of the warning, which reflects whether the warning is on time, too early, or too late. The distribution of the error $e_{\text{time}} = t_{\text{warn}} - t_{\text{ref}}$ between the time instant t_{ref} when an alarm should have been given and the time instant t_{warn} of the actual alarm gives an empirical mean and variance, which can be used for validation. As was shown in [144], the NHTSA/CAMP algorithm, presented in (2.5) performs very good overall, and will therefore be used for creating a ‘reference’ warning signal when the required acceleration a_{ref} in (2.5) crosses a threshold value a_{min} . The requirement for timely alarms is strongly correlated with the notion of reliability that the driver perceives. This can be extended to the more general notion of dependability, as discussed next.

2.3.5 Dependability requirements

Apart from the performance and comfort requirements, drivers expect an ADAS to meet stringent requirements regarding *dependability*, which covers several complementary and partly overlapping properties, such as availability, reliability, safety, confidentiality, integrity, maintainability, security, and fault tolerance [141]. For ADASs this term can be reformulated as follows:

Definition 2.3 Dependability is the property of the ADAS to give appropriate alarms and to take correct actions at the right moment with a high confidence level, for a wide set of operating conditions, in spite of the occurrence of disturbances and failure modes. \square

The dependability of a safety-critical ADAS is not only important for the driver, but also for the manufacturer and regulating authorities, *e.g.*, for certification and type approval [205]. The most relevant attributes for ADASs are reliability, safety, and fault tolerance, as defined in the following.

Definition 2.4 Reliability is the probability of a component, subsystem, or complete system, functioning correctly over a given period of time under a given set of operating conditions [226]. \square

Although failure of mechanical or electronic vehicle components only accounts for a small percentage of accidents [243], there is an increasing hazard severity of failure with an increasing degree of automation. Furthermore, as accidents by human failure will decrease, the percentage of equipment failure related accidents will relatively grow.

Software reliability is an extensive research topic on its own, and the reader is referred to the book by Leveson [147] for a comprehensive introduction to the subject. *Hardware reliability* is often assessed through a bottom-up approach using standardized procedures, as will be briefly discussed in Chapter 4. These procedures rely on statistical failure probabilities for separate components, from which the reliability of subsystems and eventually of the complete ADAS can be assessed [239]. However, the underlying reliability properties can be uncertain, and the outcome will not likely reflect the overall system reliability. Since the scope of this thesis is on the validation of the ADAS on system level, we will investigate reliability from a system perspective.

Instead of focussing on software and hardware reliability separately, a human-centered approach for reliability analysis is used. The system reliability that the driver experiences in practice is associated with the capability of the sensor post-processing and warning algorithms to provide reliable warnings. In other words, the user expects the controller to only return *true positives* (that indicate correct activation when needed), and *true negatives* (that indicate correct suppression of the safety device)¹.

Correspondingly, system reliability is associated with a low number of *false positives*. This includes false alarms due to an untimely or incorrect decision by the ADAS controller (*e.g.*, in case of a late lane-change). It also involves alarms that cannot be attributed to the scenario, but are caused by out-of-path objects or other disturbances (so-called nuisance alarms). Similarly, *false negatives* may occur, which include late detections and missed alarms due to other disturbances. Missed and false alarms may occur when driving in curves, on hilly roads, when another obstacle is blocking the line-of-sight of the sensor, or when the sensor picks up an infrastructure element as a target, as illustrated in Figure 2.6.

False alarms are usually expressed as an occurrence rate per driven distance. The NHTSA field-operational test has demonstrated a false alarm rate for an FCW system around $2 \cdot 10^{-3}/\text{km}$, and for an ACC around $10^{-5}/\text{km}$ [176]. The reliability requirement will likely become stricter with a higher degree of automation and criticality of the system, *e.g.*, the false alarm rate of a pedestrian protection system was measured at $10^{-8}/\text{km}$ [204], and the false alarm rate of a PCS activating irreversible restraints is projected at $10^{-9}/\text{km}$. Unfortunately, the expression of these discrete alarm events per driven distance (regardless of the alarm duration) creates ambiguity in the validation process. Consider for instance the occurrence of one very long continuous alarm or many short intermittent alarms, both of which will cloud the estimation of the false alarm rate.

¹In this thesis the term ‘positive’ relates to the presence of a specific condition, such as an alarm situation, whereas ‘negative’ represents the default null hypothesis. These terms do *not* have any qualitative connotation.

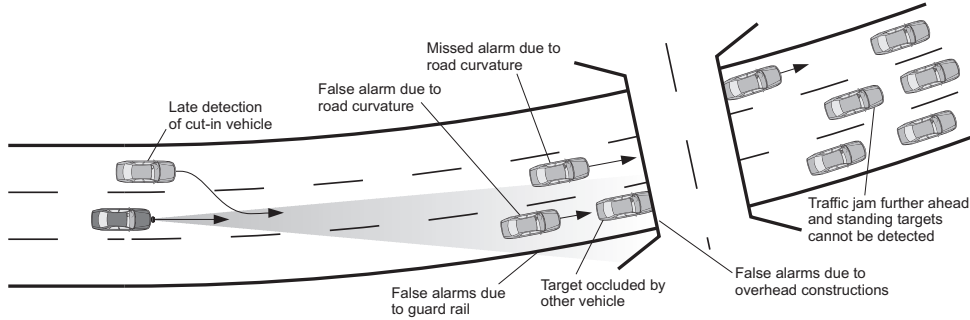


Figure 2.6: Overview of false and missed alarms due to infrastructure layout.

To provide an unambiguous measure for all types of false and missed alarms, they can best be characterized by a sample-based measure. Each data sample is therefore classified based on a comparison of the actual situation and the prediction of the ADAS, as shown in Table 2.1. Based on this table, the mathematical formulation of these reliability measures is shown in Table 2.2 and illustrated in Figure 2.7.

Although it is desired to have the false and missed alarm rate close to zero, it does not need to be, since there is always an inherent trade-off between a low missed alarm rate and a low false alarm rate. When instructed about the limitations of the system, a driver may still accept false and missed (or late) alarms, as long as they occur seldom and there is still the possibility to intervene. If the positive data set ($N_{TP} + N_{FN}$) and the negative data set ($N_{TN} + N_{FP}$) have similar size, the *accuracy* p_{accuracy} will be a good reliability index, indicating the overall rate of correct decisions. However, in case of statistical unbalance between positive and negative sets, low values for p_{FP} and p_{FN} may still result in a bad *precision* p_{CP} . This precision reflects the *correct alarm rate*, which is the *a posteriori* probability of a true alarm, as illustrated in Example 2.3. A more suitable reliability measure is the geometric mean of the precision p_{CP} and the true positive rate p_{TP} [144]. Therefore, decision thresholds for ADASs must be set not just for low false and missed alarm rates, but also for relatively high values of the *a posteriori* true alarm probability. Considering the reliability rates of state-of-the-art systems, average velocity, and typical alarm duration, approximate criteria for these rates can be estimated, as shown in Table 2.3 on page 36.

Example 2.3 Reliability measures. Consider the set of consecutive traffic scenarios depicted in Figure 2.7. During the approach maneuver, the warning system predicts a possible collision and issues a warning in the time interval [5.1, 6.1], which is considered a false alarm, since the driver makes a lane-change. Then in the inter-

Table 2.1: Prediction matrix with the number of samples, categorized as true negatives N_{TN} , true positives N_{TP} , false negatives N_{FN} , or false positives N_{FP} . Adapted from [144].

		Actual data	
		Negative (safe)	Positive (threat)
Prediction	Negative (safe)	N_{TN}	N_{FN}
	Positive (threat)	N_{FP}	N_{TP}

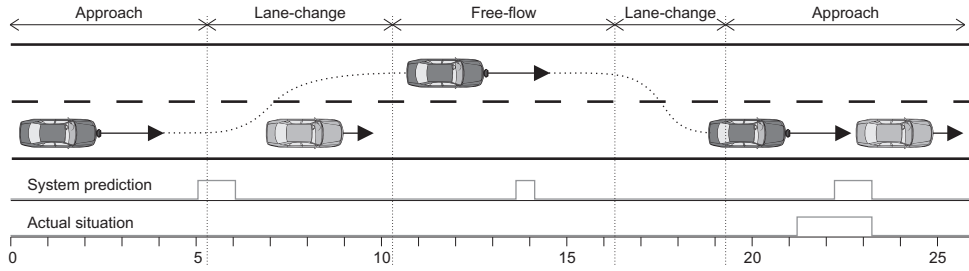


Figure 2.7: Sample-based analysis of false and missed alarms.

val [13.6, 14.2] the system gives a nuisance alarm due to detection of the guard rail. Finally, the vehicle cuts in behind a slower vehicle, causing an alarm in the interval [22.2, 23.2]. However, this critical cut-in would have required an alarm at 21.2 s. Correspondingly, the timeliness of this last warning is +1.0 s. The corresponding reliability rates are included in Table 2.2.

Another example is a (hypothetical) collision warning algorithm that never issues a warning, so the false alarm rate p_{FP} is zero. Since normal driving data is predominately safe (the real value of p is very low), such an algorithm might still enjoy an extremely high accuracy $p_{accuracy}$.

A more realistic example that illustrates the problem of unbalanced sets, considers a situation with the following number of data samples: $N_{TN} = 10^{10}$, $N_{FP} = 10^5$, $N_{TP} = 10^5$, and $N_{FN} = 1$. The resulting false alarm rate p_{FP} and missed alarm rate p_{FN} are both 10^{-5} , which may seem quite acceptable. However, the resulting correct alarm rate p_{CP} is only 0.5, due to the low occurrence rate of the dangerous event $p \approx 10^{-5}$. This means that half the alarms are incorrect!

However, we must note that these numbers may not be representative, as they are often *a priori* estimates and differ between drivers and operating conditions. Obviously, warning algorithms may have been designed based on different philosophies: avoiding all collisions, collision mitigation, or mimicking the behavior of an average driver. These considerations may not have been incorporated in the reference warning. However, regardless of the design philosophy, these algorithms will be subject to the driver's acceptance in practice. To provide an objective validation of the ADAS reliability, we must obtain a model to investigate

Table 2.2: Reliability measures calculated from the prediction matrix Table 2.1.

Rate	Definition	Example 2.3
Real occurrence rate p	$(N_{FN} + N_{TP}) / (N_{TN} + N_{FP} + N_{FN} + N_{TP})$	0.072
Accuracy $p_{accuracy}$	$(N_{TN} + N_{TP}) / (N_{TN} + N_{FP} + N_{FN} + N_{TP})$	0.906
Precision p_{CP}	$N_{TP} / (N_{FP} + N_{TP})$	0.385
True positive rate p_{TP}	$N_{TP} / (N_{FN} + N_{TP})$	0.500
False negative rate p_{FN}	$N_{FN} / (N_{FN} + N_{TP})$	0.500
True negative rate p_{TN}	$N_{TN} / (N_{TN} + N_{FP})$	0.938
False positive rate p_{FP}	$N_{FP} / (N_{TN} + N_{FP})$	0.063
Reliability p_{rel}	$\sqrt{N_{TP}^2 / (N_{FP} + N_{TP})(N_{FN} + N_{TP})}$	0.439

whether the actual scenario is ‘safe’ or ‘threatening’. For that reason, the next section will further discuss the modeling of human drivers and traffic scenarios.

Another problem with testing reliability is the difficulty in validating these probabilistic measures, due to the low occurrence rate of the event. This makes *exhaustive testing* of ADASs impossible in practice. Reliable procedures for statistical evaluation and extrapolation from field test data or extensive simulations are therefore required for the validation of system reliability. This is an issue which has been stressed by several authors (e.g., [31]), but not yet resolved. It will therefore be considered as an important topic in this thesis. A related dependability attribute is the *safety* of the system, as defined next.

Definition 2.5 *Safety* is the property of a system that it will not endanger human life or the environment [226]. \square

Safety is a very important requirement for any transportation mode. Apart from the safety potential of ADASs on a macroscopic scale, as discussed in Chapter 1, the safety of the individual system itself should also be validated. In the aerospace industry, safety standards typically require that the probability per hour of a hazard with catastrophic consequences (loss of life) be less than 10^{-9} [182]. Considering the trend towards automated vehicle control, an ADAS should also be expected to exhibit such a high level of safety [91]. However, the environment of an airplane is less complex than every-day traffic, whereas the occurrence rate of potential hazards is much greater for a car than for an airplane [219]. This implies that safety is a critical issue for ADAS validation, because of the complex and safety-critical nature. In the automotive industry the criticality of a system is usually quantified using safety integrity levels [121], and system safety is evaluated using tools that rely on failure probabilities, such as fault tree analysis (FTA) and failure modes, effects, and criticality analysis (FMECA) [7].

In addition, we need to validate the safety of an ADAS, based on the complexity of the traffic environment. Associating the notion of safety with the absence of a collision, the safety of a particular scenario can then be defined with a Boolean value $\rho_{\text{coll}} \in \{0, 1\}$, where $\rho_{\text{coll}} = 0$ means ‘no collision’, and $\rho_{\text{coll}} = 1$ means that the situation would require an intervention by the driver to prevent a collision (or that an actual collision occurs). In this way the safety of the controller can be defined as a collision probability per hour $p \in [0, 1]$.

However, a Boolean safety measure cannot distinguish in the level of severity between different situations for which $\rho_{\text{coll}} = 1$ or $\rho_{\text{coll}} = 0$. Instead of a Boolean safety measure in terms of a collision indicator, we therefore require a continuous safety measure. The minimum TTC value $t_{\text{TTC}, \min}$ indicates how imminent a potential collision has been during a scenario within the time interval $[0, t_{\text{end}}]$.

$$t_{\text{TTC}, \min} = \min_{t \in [0, t_{\text{end}}]} \left(-\frac{x_r(t)}{v_r(t)} \right). \quad (2.38)$$

The TTC is a good indication of the safety of the system, as perceived by the driver. It is commonly recognized as a rule of thumb that drivers start to pay close attention to the lead vehicle when the TTC falls below 9 or 10 s [57], and start to apply braking around $t_{\text{TTC}} = 6$ s.

Since the use of TTC only makes sense during closing situations, but not for tailgating situations, we also define the minimum value for the time headway $t_{\text{h}, \min}$ that occurs during a particular traffic scenario. In case a collision does occur, the values for $t_{\text{TTC}, \min}$ and $t_{\text{h}, \min}$ are both equal to zero. We therefore define continuous measures for the collision case using the crash velocity v_{coll} , e.g., for use in validation of a PCS.

Reliability and safety are closely related from the viewpoint of the driver, who perceives safety in terms of the failure of the system to adequately respond to a hazardous situation, *e.g.*, a missed alarm. However, the terms safety and reliability are not equivalent, since a safe system may still be unreliable, and uncovered hazards in a very reliable system may have hazardous consequences. Although safety and reliability sometimes have conflicting requirements, one aspect that contributes to both is fault tolerance, which can be defined as follows.

Definition 2.6 **Fault tolerance** is the ability of a controlled system to maintain its control objectives despite the occurrence of a fault, where a degradation of control performance may be accepted [20]. \square

The fault tolerance of the system can be validated as the ability of the control system to manage faults that otherwise would lead to failures on system level:

$$p_{\text{FT}} = 1 - \frac{N_{\text{failures}}}{N_{\text{faults}}}, \quad (2.39)$$

where N_{faults} are the number of faults that occur in the system and N_{failures} are the number of failures that remain and affect the operation of the system. This fault terminology is described further on in Section 2.5.1. The fault tolerance of a safety-critical system can be improved through a variety of methods, as will be further investigated in Chapter 3.

2.3.6 Compatibility with operating conditions

The requirements presented in this section are summarized in Table 2.3. Unfortunately, these stringent requirements are often in contradiction to the increasing complexity of an ADAS controller and its environment. Usually the environmental compatibility is defined in terms of environmental disturbances and ambient conditions that the system should be able to withstand. However, we would like to define the complexity of the operating conditions in a more quantitative way in relation to the above-mentioned performance measures.

The operating conditions can be grouped in a multi-dimensional parameter set \mathcal{Q} , which is an n -dimensional set of all possible parameter combinations $\mathbf{q} \in \mathcal{Q} \subset \mathbb{R}^n$. The vector \mathbf{q} consists of initial conditions, scenario parameters, driver input, disturbances, and failure modes. Since the performance is often validated by a probabilistic measure p , depending on the occurrence rate of scenarios, we also need to define the probability distribution of \mathcal{Q} . The subsets of \mathcal{Q} and their (multivariate) probability functions $f_{\mathcal{Q}}$ will be investigated next.

2.4 Scenario definition by microscopic traffic modeling

The performance measures ρ are mainly affected by the traffic environment, *i.e.*, the motion of other vehicles. Since this motion enters the system through the environment sensor, they are grouped in an input disturbance vector $\mathbf{d}_{\text{traffic}}$, as was illustrated in Figure 2.5.

2.4.1 Single-lane traffic modeling

Traditional *macroscopic* traffic models treat vehicles as homogenous flows that obey simple speed-flow relationships (see *e.g.*, Tampère [232]). However, this approach is inadequate for

Table 2.3: Summary of requirements.

Measure	Type	Symbol or Definition	Criterion (if available)
<i>Stability</i>			
Individual vehicle stability	Boolean	$\lim_{a_{i-1} \rightarrow 0} e_i, \lim_{a_{i-1} \rightarrow 0} e_i$	$= 0$
String stability	Continuous	$\ H(s)\ _\infty$	≤ 1
<i>Performance</i>			
Response time	Continuous	t_{delay}	
Tracking error	Continuous	$\left\ \frac{x_r(s)}{x_{\text{ref}}(s)} \right\ _\infty$	
Robustness	Continuous	$\left\ \frac{a(s)}{x_{r,m}(s) - x_r(s)} \right\ _2$	
Control effort	Continuous	$\ u(t)\ _{\text{pow}}$	
<i>Comfort</i>			
Naturalistic driving behavior	Continuous	$\ x_{r,\text{ref}}(t) - x_r(t)\ _{\text{pow}}$	
Longitudinal acceleration	Continuous	$\ a_{\text{long}}(t)\ _{\text{pow}}$	
Warning timeliness	Continuous	e_{time}	FCW: $\text{std}\{e_{\text{time}}\} < 1 \text{ s}$
<i>Dependability</i>			
False alarm rate	Boolean	p_{FP}	FCW: 10^{-4} , ACC: 10^{-6}
Missed alarm rate	Boolean	p_{FN}	10^{-3}
Correct alarm rate	Boolean	p_{CP}	0.99
Reliability	Boolean	p_{rel}	0.99
Collision rate	Boolean	p_{coll}	10^{-9}
Fault tolerance	Boolean	p_{FT}	0.99
Minimum time-to-collision	Continuous	$t_{\text{TTC}, \text{min}}$	$\leq 6 \text{ s}$
Minimum time headway	Continuous	$t_{\text{h}, \text{min}}$	$\leq 0.5 \text{ s}$
Crash velocity	Continuous	v_{coll}	

the system-level validation of ADASs, as it requires accurate knowledge of the interactions between individual vehicles. Since we are concerned with the performance on *microscopic* level, we concentrate on the development of a modeling environment for microscopic traffic scenarios, such that ADASs can be validated for a representative set of traffic scenarios. Therefore, the macroscopic traffic flow is split up into distinctive subscenarios that are representative of the scenarios that a longitudinal control algorithm should handle.

A human driver looks only several vehicles ahead (usually two or three [99]), and longitudinal control and warning systems only consider target vehicles that are either in, entering into, or leaving the host vehicle's lane, as illustrated by Figure 2.8. We therefore focus the identification of the parameter set on *single-lane* scenarios, which are defined as follows.

Definition 2.7 A **single-lane scenario** is a particular configuration of up to three vehicles and their behavior in a single lane over a predetermined period of time. \square

2.4.2 Subscenarios in single-lane traffic

In each scenario the host vehicle is denoted as the i -th vehicle. An (optional) preceding vehicle (*i.e.*, the closest vehicle in front of the host in the longitudinal sense, though not necessarily in the same lane) is denoted as the target vehicle $i-1$. In case of n preceding

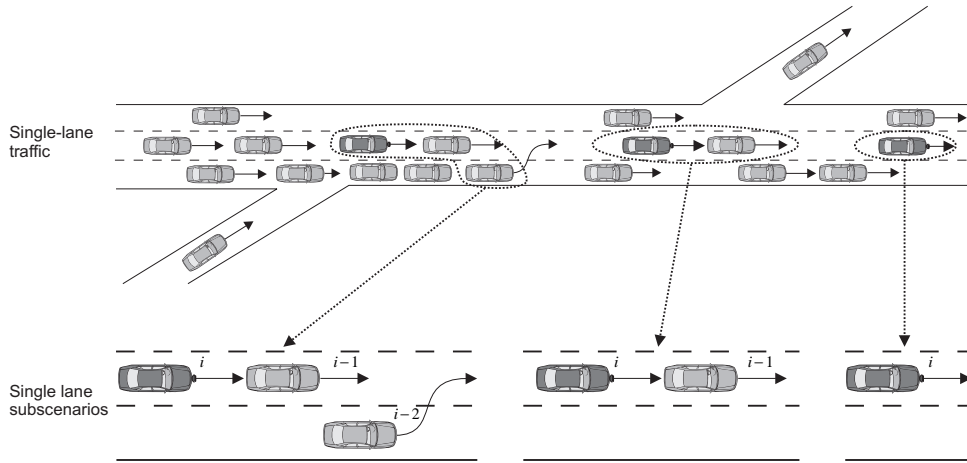


Figure 2.8: Top: definition of single-lane traffic. Bottom: examples of single-lane subscenarios. From left to right: car-following with cut-in; car-following; and free-flow.

vehicles, additional (optional) preceding vehicles are denoted by $i-2, \dots, i-n$, where the order i depends on how close the target vehicle is to the host in longitudinal direction.

Usually ADAS control system analysis distinguishes between either free driving or car-following. However, we would like to obtain a more complete overview of the scenarios that an ADAS should handle. The single-lane traffic scenario is therefore divided into several subscenarios, which are recurrent behavior patterns of vehicles. For two vehicles in single-lane traffic the following seven subscenarios can be identified:

- *Free-flow*: Vehicle i has no preceding vehicle.
- *Car-following*: Vehicle i is following vehicle $i-1$ with a constant velocity.
- *Cut-in*: Vehicle $i-1$ moves in front of vehicle i from an adjacent lane.
- *Cut-out*: Vehicle $i-1$ moves out of the lane of vehicle i to an adjacent lane.
- *Lane-change*: Vehicle i changes lane and enters another subscenario in an adjacent lane.
- *Approach*: Vehicle i drives towards vehicle $i-1$ in the same lane.
- *Separate*: Vehicle $i-1$ drives away from vehicle i in the same lane.

Figure 2.9 shows a top view of the subscenarios together with two graphs of parameters, which contain the vehicle's velocity profiles (left) and the relative distance (right) during the subscenarios. For example, an approach maneuver typically leads to a car-following situation, after which the host vehicle may undertake a lane-change or the target vehicle a cut-out. The interaction between the different subscenarios is illustrated in Figure 2.10. The subscenarios are considered as individual modes, where the transitions between those modes are depicted by the arrows.

The following subscenario configurations are possible in case of zero, one, or two preceding vehicles. The most basic configuration for a single-lane scenario is in case of a single

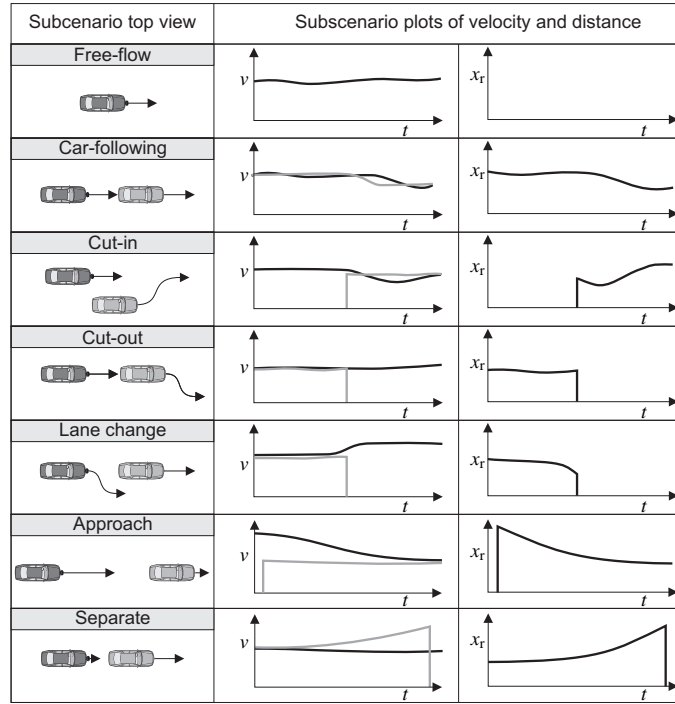


Figure 2.9: Overview of the subscenarios in single-lane traffic. On the right the velocity profiles of the target vehicle (grey line) and host vehicle (black line), as well as the distance between the vehicles, during the scenario are shown. Note that when no target is present, the states v and x_r are zero.

vehicle. The only two possible subscenarios for single-lane scenario with one vehicle are free-flow and lane-change. For the single-lane scenario, consisting of two vehicles (i and $i-1$), five configurations are possible: *car-following*, *cut-in*, *cut-out*, *lane-change*, and *approach*². The configurations for single-lane scenarios, consisting of three vehicles (i , $i-1$, and $i-2$) are slightly more complex. Therefore the 3-vehicle configurations are described in two steps. First, the possible subscenarios between vehicles i and $i-1$ are *car-following*, *approach* and *lane-change*. Secondly, there are five possible subscenarios between vehicles $i-1$ and $i-2$: *car-following*, *cut-in*, *cut-out*, *lane-change*, and *approach*, resulting in a total of 15 configurations for single-lane scenarios with three vehicles.

2.4.3 Modeling and calibration of single-lane scenario parameters

In order to derive probabilistic performance measures, a representative probabilistic model should be investigated. The subscenarios are characterized by initial conditions, such as the initial vehicle velocities $v_i(0)$ and the initial distance $x_r(0)$. Another relevant scenario parameter is the *duration* T_j of the subscenario j , *i.e.*, the time frame over which to evaluate the performance of the ADAS, after which a transition to a next subscenario occurs. This

²In the remainder, the subscenario *separate* is integrated in the *car-following* subscenario.

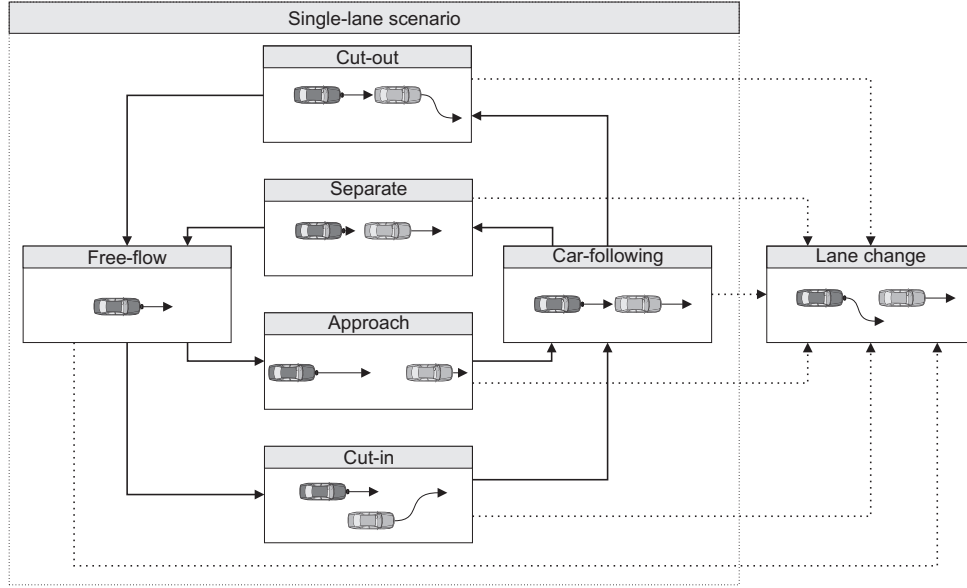


Figure 2.10: The relations between the different subscenarios in single-lane traffic.

duration is determined by the duration of the subscenario that applies to the host vehicle of the configuration. As an example, the scenario durations in Figure 2.7 are 5.3 s, 5.0 s, 6.0 s, 3.0 s, and 6.7 s respectively. Furthermore, the *occurrence rate* of the configuration represents how often a particular configuration occurs, and is determined by the multiplication of the occurrence rates of the subscenarios in the configuration.

The scenario parameters of the above-mentioned subscenarios are modeled by probability density functions (PDFs). In case a parameter is uncorrelated, the PDF of a normal, log-normal, or Laplace distribution will be used. The normal distribution of a variate X is given by

$$f(X) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(X-\mu)^2}{2\sigma^2}}, \quad (2.40)$$

where μ is the mean, and σ is the standard deviation of X . The log-normal distribution of a variate $X > 0$ is given by

$$f(X) = \frac{1}{\sigma\sqrt{2\pi}X} e^{-\frac{(\ln X - \mu)^2}{2\sigma^2}}, \quad (2.41)$$

where μ and σ are the mean and standard deviation of the variate's logarithm. Finally, the Laplace distribution of X is given by

$$f(X) = \frac{1}{2\zeta} e^{-\frac{|X-\mu|}{\zeta}}, \quad (2.42)$$

where μ is a location parameter, and ζ is a scale parameter.

In a next step these parameter models must be calibrated with test data in order to form a representative set for microscopic traffic simulation. Several traffic databases are available, each containing a considerable amount of useful data for traffic modeling. Examples are the

Table 2.4: Calibration results for the occurrence rate of each subscenario.

Subscenario	Occurrence rate per hour	Total duration [%]
Free-flow	16	13.7
Car-following	88	67.0
Cut-in	21	2.1
Cut-out	29	3.3
Approach	15	8.2
Lane-change	42	5.8

CAMP database [170–172] and the SAVME database [143]. Unfortunately these databases do not distinguish between different types of scenarios, nor do they contain information on the occurrence rate and initial conditions of each subscenario.

We therefore use raw data obtained from an instrumented car of the California PATH program [36]. This database contains measurements of relevant vehicle states, such as position, velocity, acceleration, steering angle, throttle angle, and brake use during hundreds of hours of highway driving by several subject drivers. Surrounding traffic is monitored by radar to record the distance and relative velocity to preceding vehicles. Furthermore, video recordings of the driver and the forward scene allow to categorize the scenarios and assess driver behavior.

Unfortunately, the large amount of raw data requires post-processing before it can be used for validation of ADASs in single-lane scenario simulations. First, the data has been assigned to one of the subscenario categories, as was illustrated in Figure 2.7. Next, the calibration of scenario parameters has been performed using a maximum likelihood estimation on the parameters of the PDFs in (2.40)-(2.42). We refer to [241] for further details on the data-extraction and calibration procedures that were used.

Table 2.4 shows the calibration results for the occurrence rate of subscenarios. Secondly, the calibration results for the subscenario duration are provided in Figure 2.11. The distribution of the initial relative velocity $v_r(0)$ is shown in Figure 2.12. Note that no fit was performed for free-flow, due to the absence of a target vehicle, leaving $v_r(0)$ undefined. Instead, the velocity parameter to be estimated for free-flow is the ratio between the velocity $v(0)$ and the speed limit v_{limit} , as shown in Figure 2.13. For subscenarios other than free-flow, the initial distance $x_r(0)$ is a function of the initial velocity $v(0)$. We therefore need a bivariate normal PDF, which, in case of correlated variates X_1 and X_2 , is defined as

$$f(X_1, X_2) = \frac{1}{2\pi\sqrt{\det(\mathbf{S}_{X_1, X_2})}} e^{-\frac{1}{2} \left(\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} - \begin{bmatrix} \mu_{X_1} \\ \mu_{X_2} \end{bmatrix} \right)^T (\mathbf{S}_{X_1, X_2})^{-1} \left(\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} - \begin{bmatrix} \mu_{X_1} \\ \mu_{X_2} \end{bmatrix} \right)}, \quad (2.43)$$

where \mathbf{S}_{X_1, X_2} is the covariance matrix for X_1 and X_2 , μ_{X_1} is the mean of X_1 , and μ_{X_2} is the mean of X_2 . In order to fit (2.43) to the raw vehicle data with $X_1 = \log(x_r(0))$ and $X_2 = v(0)$, the parameters \mathbf{S}_{X_1, X_2} , μ_{X_1} , and μ_{X_2} are calibrated. The results are given in Figure 2.14.

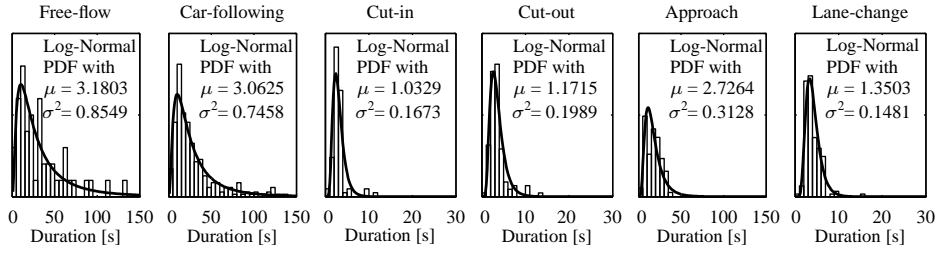


Figure 2.11: Calibration results for the subscenario duration.

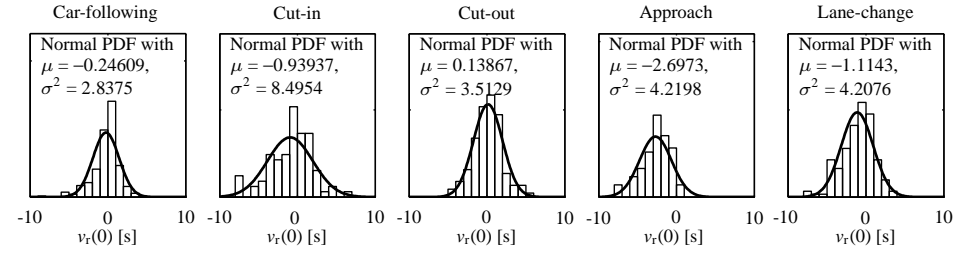


Figure 2.12: Calibration results for the initial relative velocity $v_r(0)$.

2.4.4 Calibration of the Gipps driver model

Apart from initial conditions for the traffic scenarios, the behavior of vehicles during the scenarios must be modeled. If the host vehicle is controlled by a longitudinal control system, it does not require a driver model. However, a driver model is required for simulation of preceding vehicles, as well as for providing a reference for collision warning systems. We therefore calibrate the parameters of the Gipps driver model for the individual subscenarios. Recall from Section 2.3.4 that the Gipps model is characterized by:

- The maximum allowable acceleration of the host vehicle a_{\max} .
- The estimate of the most severe braking deceleration $\hat{a}_{\min,i}$ that the driver of vehicle i wishes to undertake.
- The ratio between the desired velocity and the initial velocity $\lambda_{v,i} = v_{\text{ref},i}/v_i(0)$ that indicates whether the driver would like to accelerate or decelerate.
- The driver reaction time t_{reac} .

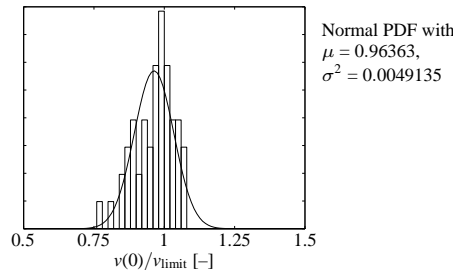


Figure 2.13: Calibration of the ratio of initial free-flow velocity and the speed limit.

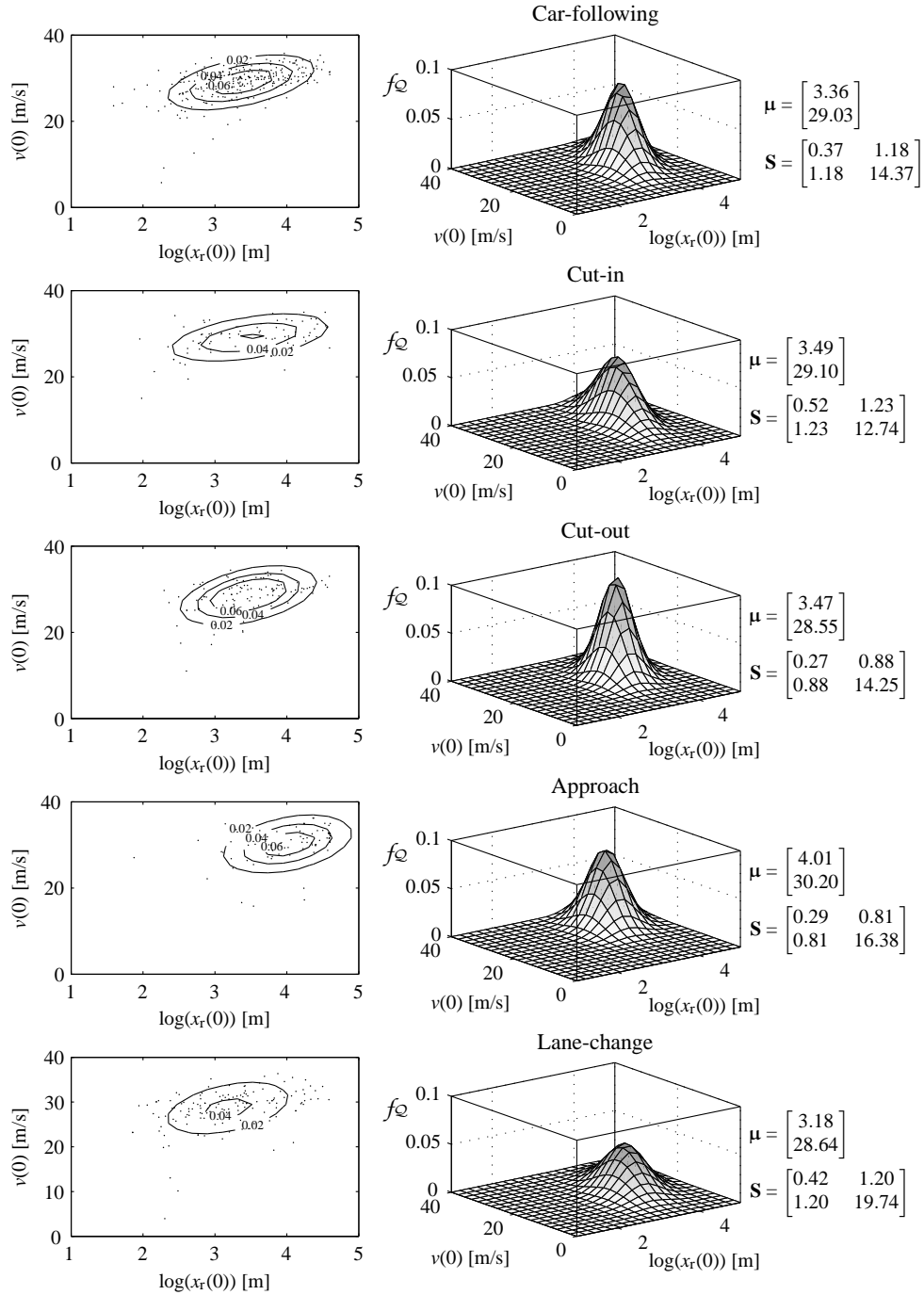


Figure 2.14: Calibration results of each subscenario for the parameters $x_r(0)$ and $v(0)$. Left: Top view of the data points and the contour lines of the fitted distribution. Right: three-dimensional plot of the fitted bivariate normal distribution.

Table 2.5: Calibration of the ratio $\lambda_{v,i}$ for vehicles 1 and 2, using the Laplace PDF (2.42).

		Subscenarios				
		Car-following	Cut-in	Cut-out	Approach	Lane-change
Vehicle 1	Mean μ	1.03	0.99	1.01	1.09	1.01
	Scaling ς	0.09	0.12	0.08	0.11	0.11
Vehicle 2	Mean μ	1.01	0.99	1.00	0.98	1.01
	Scaling ς	0.03	0.01	0.01	0.05	0.02

From the experimental data the maximum allowable acceleration a_{\max} is fitted by a log-normal distribution

$$a_{\max} \sim \log\text{-}\mathcal{N}(0.44, 0.17). \quad (2.44)$$

This model is valid for all relevant subscenarios in the single lane, since the allowable acceleration of a vehicle is considered independent of the subscenarios. According to Gipps [86] the most severe deceleration for the host vehicle i is related to (2.44) and given by

$$\hat{a}_{\min,i} = -2a_{\max}. \quad (2.45)$$

Furthermore, the maximum deceleration by vehicle $i-1$, as estimated by the driver of vehicle i , is given by

$$\hat{a}_{\min,i-1} = \frac{\hat{a}_{\min,i} - 3}{2}. \quad (2.46)$$

The distribution of the driver reaction time t_{reac} is given by [56] as

$$t_{\text{reac}} \sim \log\text{-}\mathcal{N}(-0.31, 0.16), \quad (2.47)$$

which corresponds to the experimental data. Finally, the ratio between the desired velocity and the initial velocity $\lambda_{v,i} = v_{\text{ref},i}/v_i(0)$ are calibrated from the experimental data, as shown in Table 2.5.

2.5 Impact of disturbances and faults on dependability

The previous section has presented the set of traffic scenarios that forms the input to the ADAS control algorithm. Unfortunately, as indicated in Figure 2.5, the ADAS is also susceptible to uncontrolled and unknown, but bounded, disturbances. The system is affected by parametric uncertainty in the form of measurement noise \mathbf{v}_s affecting the measurement vector \mathbf{z} . These are disturbances originating from environmental and ambient conditions, such as temperature, rain, snow, light, vibration, electromagnetic disturbances, and fog. The measurement noise is usually Gaussian distributed, as will be shown later in Chapter 3.

In addition, the driver forms an important source of disturbances, such as distraction, over-reacting, panicking, and ignorance. However, the influence of these psychological driving attributes on the system dependability are difficult to quantify and will not be investigated in this thesis. Instead, driver behavior is modeled by PDFs of specific driving style parameters (as discussed above), which enters the system as a vector $\mathbf{d}_{\text{driver}}$.

Furthermore, the simulated system G_{sim} that is used for controller synthesis and analysis, always exhibits parametric uncertainty due to modeling errors and variable system

parameters, as well as unstructured uncertainty due to unmodeled dynamics Δ_1 . All this uncertainty is grouped in the vector $\Delta \in \mathcal{D}$.

One of the main requirements for any control system, is that it has robust stability and robust performance in the presence of the disturbance vector Δ . The issue of robust performance and robust stability for ACC-equipped vehicles has been investigated by several authors [154, 267], but only for the linear case, with a simple controller, and for individual disturbances. Instead, it is of interest to investigate the robustness of a complex ADAS controller for a multivariate disturbance set.

Some perturbations on the system cannot be taken into account as ‘normal’ disturbances, but can better be regarded as *faults*. These faults are unpredictable perturbations that may seriously influence the performance of the ADAS.

2.5.1 Fault terminology

The potential decrease of controller dependability under the influence of faults has been well researched since the 1970s, as summarized in the historical overview by Isermann [113]. First we provide definitions for the fault terminology, as developed by Patton [186]:

Definition 2.8 A **fault** in a system is a non-permitted and unpredictable deviation of a characteristic property of the system that leads to the inability to fulfill the intended purpose of that system. \square

Under certain conditions, a fault can manifest itself as an *error*, *i.e.*, a deviation between a measured value (of an output variable) and the true value. Although a fault remains localized in the affected code, circuitry, or subsystem, multiple errors can originate from one fault site and propagate throughout the system. Although a fault may be a tolerable malfunction, an error may induce a *failure*, *i.e.*, a deviation from the appropriate system behavior and consequently partial or complete breakdown of a component or function. The various ways in which failures occur are called *failure modes*. A failure forms a *hazard* if it can lead to undesired consequences, such as an accident, given certain environmental conditions beyond the control of the system designer.

2.5.2 Classification of faults in ADASs

Faults appear in many forms and can be categorized according to the location within the system, the time of occurrence and duration, and the form of the fault. The starting point for a dependable ADAS design is knowledge about faults that may occur in a system. The functional decomposition of an ADAS, as shown in Figure 2.4, can support the identification of faults that can occur during the design and operation of the system, and the hazards associated with them using FTA and FMECA tools.

Figure 2.5 illustrates the occurrence of faults in both hardware and software. However, software faults \mathbf{f}_{sw} are always the consequence of incorrect specification or design, and are usually systematic. Hardware faults include faults in sensors \mathbf{f}_s , actuators \mathbf{f}_2 , mechanical component failures \mathbf{f}_1 in the vehicle itself, controller hardware, and communication busses.

Faults are also classified according to their time behavior as abrupt, incipient or intermittent faults. Abrupt faults occur instantaneously and have a step-like behavior. Incipient faults have a drift-like behavior and have a slowly parametric change. Intermittent faults appear and disappear repeatedly and have a temporary effect.

2.5.3 Modeling the form of a fault

Looking closer at the effect of faults on the system, we can distinguish between *additive* and *multiplicative* faults, as described by Patton [183]. In practical applications, some faults have the effect of deviations on the system's dynamic parameters, as incorporated in the system matrix \mathbf{A} . These are effectively multiplicative faults that manifest themselves as a multiplication of state or control signals with parameter deviations. Other faults have an additive effect upon the system inputs and/or outputs and therefore affect these signals additively. We can refer to these as additive fault signals.

Faults in environment sensors

Sensor faults are discrepancies between the measured and actual values of individual plant variables. These faults are usually considered as additive (independent of the measured magnitude), though some sensor faults (such as complete failure resulting in zero signal) may be better characterized as multiplicative.

Although vehicle state sensors are very reliable, a failure in a wheel speed sensor, accelerometer or yaw rate sensor can be safety-critical. Furthermore, GPS satellite navigation is a relatively unreliable source of position information and is therefore usually combined with information from inertial sensors (accelerometers and yaw sensors) and wheel speed sensors. Environment sensors are even more safety-critical, because they provide inter-vehicle information, which is crucial for longitudinal control. A number of sources describe the faults that may occur in radar, lidar, and vision systems [2, 152]. These faults are typically caused by multi-path reflections, 'ghost' objects, weather conditions, vibration, or hardware failure.

Networking faults

Nowadays, in-vehicle networking is primarily performed using the CAN protocol, and CAN bus failure can be a very critical source of errors [12]. VVC is a safety-critical factor because it passes the speed and acceleration of preceding vehicles to the vehicles behind. The performance of VVC largely depends on its power, signal processing techniques and other algorithms, inter-vehicle distance, the environment, and other interferences. For example, a large obstacle or a building between two vehicles may cause signal loss. Signal loss is typically a multiplicative fault and has an abrupt form.

Actuator faults

Actuator faults are discrepancies between the input command of an actuator and its actual output. Actuator faults are usually handled as additive, although some faults (complete failure) may better be described as multiplicative. Actuator faults in longitudinal vehicle control systems relate to the engine and the braking system and are very safety-critical. The most hazardous actuator fault is that an electrohydraulic brake system fails to energize, which may prevent the vehicle from slowing down.

Process faults

Additive process faults are unknown inputs acting on the plant, which are normally zero and which, when present, cause a change in the plant outputs independent of the known inputs.

Multiplicative process faults are changes (abrupt or gradual) in some plant parameters. They cause changes in the plant outputs, which depend also on the magnitude of the known inputs. Process faults can occur in any mechanical or electrical subsystem of the vehicle, *e.g.*, a flat tyre. However, these are usually prevented by regular maintenance, and fall outside the scope of this thesis.

System modeling

Additive and multiplicative faults enter differently in the state space model (2.14). The system is subjected to disturbances $\mathbf{d} \in \mathcal{D}$, actuator and sensor faults $\mathbf{f} \in \mathcal{F}$, process noise \mathbf{w} , and measurement noise \mathbf{v} . Considering the fault and disturbance vectors as purely additive, such a system is modeled as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}_2\mathbf{d}_2 + \mathbf{F}_2\mathbf{f}_2 + \mathbf{w}, \quad (2.48)$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{E}_1\mathbf{d}_1 + \mathbf{F}_1\mathbf{f}_1 + \mathbf{v}. \quad (2.49)$$

where \mathbf{E}_1 and \mathbf{E}_2 are the unknown input matrices, and \mathbf{F}_1 and \mathbf{F}_2 the fault entry matrices. The vectors \mathbf{w} and \mathbf{v} are Gaussian white-noise processes.

2.6 Summary

Since the research field concerning ADASs is very broad, this chapter has briefly outlined different types of ADASs, such as driver warning systems, longitudinal control systems, and pre-crash systems, as well as the relevant enabling technologies. The application of these ADASs and technologies will be illustrated with three case studies later on in this thesis.

The discussion of these systems and technologies has also highlighted some challenges regarding the development of ADASs. The main challenge is to satisfy stringent performance and dependability requirements, regarding string stability, naturalistic driving behavior, reliability, safety, and fault tolerance. The dependability requirements are particularly relevant, because of the increasing complexity of the ADAS, the vehicle, and its traffic environment. Apart from usual perturbations, such as traffic scenarios, there are also disturbances which have to be taken into account during the operation of the system. In addition, a major safety-critical factor is the occurrence of faults, especially in environment sensors, GPS, and VVC. The controller then receives incorrect information about its environment, in case no backup system is available.

A consequence of the safety-critical operation is that the sensorial platform and controller will be required to exhibit a very high level of fault tolerance. In the next chapter we will therefore present a fault-tolerant state estimation system using sensor fusion techniques. However, these fault management features make the ADAS even more complex and difficult to thoroughly understand. This also underlines the need for a thorough design and validation methodology, which will be investigated in Chapters 4 and 5.

Chapter 3

Fault-tolerant state estimation

The previous chapter has shown that an advanced driver assistance system (ADAS) has to combine adequate performance with high dependability, that is, to give appropriate alarms and to take correct actions at the right moment for a wide set of operating conditions, even in the presence of safety-critical disturbances and failure modes. Nominal and robust performance can usually be guaranteed by robust control techniques. The question then arises how faults, if they occur, can be managed in order to achieve a dependable system. This chapter will therefore investigate methods for fault management, as defined in Objective 2 on page 8. These methods are illustrated by the design of a fault-tolerant state estimation system for two demonstrator vehicles. Section 3.1 starts with a general introduction to the topic of fault management and further outlines the goal of this chapter. Based on a review of the theory and state-of-the-art on model-based fault detection in Section 3.2, a suitable configuration for sensor fault management is selected. Section 3.3 introduces the two demonstrator vehicles and the sensor faults that should be managed. In Section 3.4 a vehicle state estimator is developed for the nominal case. Subsequently, in Sections 3.5 and 3.6 fault management systems are developed to handle faults in vehicle state sensors and in environment sensors, respectively. Finally, Section 3.7 concludes the chapter.

3.1 Introduction to fault management

Since faults may significantly degrade the dependability of an ADAS, much effort is placed on *fault prevention* in the design process and *fault removal* during the testing of a control system [141]. However, a dependable design should also anticipate the occurrence of faults during system operation, making *fault tolerance* an important system attribute. Fault tolerance can be achieved by a combination of *robustness* and *redundancy* on various system levels: on the architectural level, in hardware components, and in the control system itself.

3.1.1 Fault-tolerant system architecture

Fault tolerance for automotive mechatronic systems is increasingly provided on system level by choosing a suitable distributed control architecture [68]. The dependability of a distributed system is increased by allocating subfunctions to logically or physically separate subsystems that cooperate to achieve the required system functionality. In addition, it may

provide more flexibility in design and increase the performance of the system by load sharing and more computing power. However, distributed control also introduces additional complexity, costs, weight, and packaging problems [135].

3.1.2 Robustness and redundancy for component fault tolerance

Obviously, the dependability of a system can be increased by improving the hardware *robustness* of its components. However, hardware is likely to fail eventually and increased robustness is associated with exponentially increasing costs, such that faults can never be completely avoided, in which case the system must still exhibit some form of fault tolerance.

A more sensible option would therefore be to implement *hardware redundancy*, where dependability is provided by a backup component. In the aerospace industry *static redundancy* is often implemented, where three or more modules perform the same task on an input signal in parallel. A voting mechanism compares their output signals and decides by majority which signal value is the correct one. *Dynamic redundancy* on the other hand, requires only one module in operation, and if it fails, a backup unit takes over. An example is a hydraulic backup in current electromechanical brake systems [122]. Again, some sort of fault detection is necessary to observe whether the primary component has failed.

Unfortunately, hardware redundancy is often not acceptable for automotive systems, due to constraints on costs and available space. Furthermore, faults are not always caused by faulty hardware, but may also result from specific operating conditions. Instead of adding extra hardware, we will improve fault tolerance by *analytical redundancy*, that is, by implementing a mechanism for fault detection, state estimation, and control reconfiguration.

3.1.3 Fault tolerance for control systems by analytical redundancy

Passive fault tolerance

In a feedback control system, small additive or multiplicative faults in the system are covered by usual robustness properties of the controller [276]. Robust control is often referred to as *passive* fault tolerance. The impaired system continues to operate with the same controller with the original control objective, but this comes at the cost of a trade-off between the level of robustness and nominal system performance. In case of large faults, the controlled system may even become unstable, which underlines the need for an *active* approach to fault tolerance.

Active fault tolerance

In order to provide an active fault-tolerant approach by analytical redundancy, the occurrence of a fault must be detected, diagnosed, and handled. This *fault management* process consists of the following steps, as illustrated in Figure 3.1:

- *Fault detection*: the indication that a fault has occurred.
- *Fault isolation*: the assessment of the exact location of the fault.
- *Fault identification*: the identification of the type and magnitude of the fault.
- *Control reconfiguration*: the accommodation of the effect of the fault by changing the control parameters or control structure.

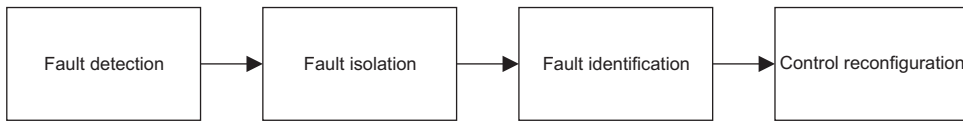


Figure 3.1: The consecutive steps in fault management.

The first two steps are often grouped together as fault detection and isolation (FDI). When the fault identification function is included along with FDI, the process is referred to as fault detection and diagnostics.

Depending on the type of fault, the reconfiguration may change the structure and/or parameters of the controller, in order to achieve *fault-tolerant control* (FTC). *Supervision* is thus required to monitor whether the control objective is met. If not, the supervisor should modify the control structure and/or parameters that make the failed closed-loop system meet the objective [20]. In order to accommodate a fault, *graceful degradation* of the system functionality may also be necessary, where the supervisor calculates a revised control objective with a degraded (but still acceptable) performance. Whilst supervision is essential for active FTC, this research area has only received little attention [19].

Fault-tolerant control is a multi-disciplinary field of research, where a variety of issues, such as reliability, redundancy, reconfiguration, robustness and supervision, all contribute to the overall fault tolerance. Especially the integration of FDI and FTC methods is necessary for achieving a truly dependable system for safety-critical applications. The objective of this chapter is therefore to integrate several methods for FDI and FTC with the aim of fault-tolerant state estimation for ADASs.

3.2 Model-based fault detection and isolation

Currently, fault detection methods for automotive applications are usually based on model-free *plausibility checking*, where a sensor measurement is simply compared to a threshold. In case of a fault (*e.g.*, an ABS sensor fault), the system goes into a fail-safe mode, and a warning is communicated to the driver (*e.g.*, by lighting up a warning display). The advantage is its simplicity, requiring little implementation effort. However, only large faults (*e.g.*, complete failure) can be detected and more detailed fault diagnosis is not possible. Furthermore, with increasing automation of the driving task, reliable fault detection and analytical redundancy will be necessary for safety-critical vehicle control functions.

Considering the complexity of the operating environment, *model-based* methods for FDI will be more suitable. Many different approaches for FDI have been developed since the 1970s, as summarized in several textbooks [11, 32, 70, 186], in order to achieve fast and robust detection of faults. These methods have been applied in a variety of safety-critical industries [113], such as the chemical, aerospace, and nuclear industry. From the 1990s, FDI has also been increasingly researched for automotive control systems [111]. This section gives a brief overview of these FDI methods and the possibilities for their application to the fault types that were mentioned in Chapter 2.

Quantitative model-based FDI methods use measured input signals $\mathbf{u}_m(t)$ and measured output signals $\mathbf{z}(t)$ of the process to derive characteristic values of the process, such as model parameters or state estimates $\hat{\mathbf{x}}(t)$. These estimates can be compared to the corresponding

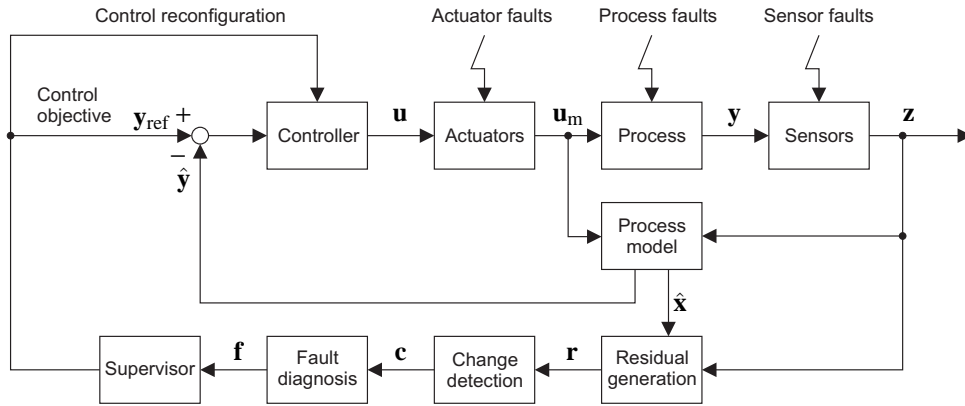


Figure 3.2: Schematic diagram of model-based fault management.

measurements and the resulting differences are the *residuals* $\mathbf{r}(t)$ of the fault detection process. The deviation of residuals from the ideal value of zero is the combined result of noise, modeling errors, and faults. By *change detection* methods these residuals are analyzed and changes $\mathbf{c}(t)$ can be interpreted as faults $\mathbf{f}(t)$, as illustrated in Figure 3.2. Residuals can be generated using a variety of methods, depending on the type and location of the fault.

3.2.1 Parameter estimation methods

Parameter estimation can be used to detect process faults if the faults are reflected in process parameters, as discussed in the tutorial paper by Isermann [110]. The comparison of reference parameters of input-output models with the actual measured parameters can indicate the appearance of the fault. Generally speaking, the information gained from parameter estimation is very high, especially for multiplicative process faults. A further advantage is that they are less sensitive to noise than the other methods presented in this section. However, as the aim of this chapter focuses on sensor fault detection, we will further investigate other model-based fault detection methods.

3.2.2 Dedicated state estimators

For an observable system the system state can be estimated from sensor measurements using a variety of observers. Dedicated observers excited by one measurement z_i can be used as a baseline solution. From this observer input the other outputs can be reconstructed in the vector $\hat{\mathbf{y}}$ and compared with the corresponding measurements \mathbf{z} . This allows the detection of single sensor faults, as presented by Patton and Chen [185]. The applicability of each of the observer types depends heavily on the specific problem, especially on the degree of analytical redundancy provided by the process measurements.

In case of a stochastic process, a Kalman filter can be used to estimate the state of a linear dynamic system that has been perturbed by Gaussian white noise [252]. The stochastic innovations of a Kalman filter can indicate a change in the internal state of the process and are useful for fault detection purposes [185].

Since dynamic systems often exhibit nonlinear behavior, a nonlinear observer or an extended Kalman filter (EKF) may also be applied [252]. The EKF linearizes the system around the current state by calculating the Jacobian of the process and measurement models. EKF-based sensor fusion is often used for state estimation applications, *e.g.*, in vehicle navigation systems [60]. Although the EKF is an effective solution, there are drawbacks to this filter. One is that, if the system is highly nonlinear, the linearization might introduce substantial errors in the estimation and might lead to divergence of the EKF. Furthermore, if the noise distribution is not Gaussian, this could produce additional errors in the estimate. These robustness issues should therefore be taken into account in the design of the filter.

In the previous chapter, a linear continuous-time representation of the system was used for reasons of transparency, see (2.14). In this chapter we recognize that a vehicle system has nonlinear dynamics with discrete-time measurements. We therefore consider a nonlinear discrete-time representation, using a general stochastic difference equation for the process

$$\mathbf{x}(k) = f(\mathbf{x}(k-1), \mathbf{u}(k-1), \mathbf{w}(k-1)) \quad (3.1)$$

and for the measurement

$$\mathbf{z}(k) = \hat{h}(\mathbf{x}(k), \mathbf{u}(k), \mathbf{v}(k)). \quad (3.2)$$

The nonlinear function f relates the state $\mathbf{x}(k)$ at the current time step k to the state $\mathbf{x}(k-1)$, the control input $\mathbf{u}(k-1)$, and the noise $\mathbf{w}(k-1)$ at the previous time step $k-1$. The nonlinear function \hat{h} relates the measurement $\mathbf{z}(k)$ to the state $\mathbf{x}(k)$, the input $\mathbf{u}(k)$, and the measurement noise $\mathbf{v}(k)$.

The variables $\mathbf{w}(k)$ and $\mathbf{v}(k)$ are assumed to be Gaussian distributed with zero mean and a covariance represented by the covariance matrices \mathbf{W} and \mathbf{V} , respectively:

$$\mathbf{w} \sim \mathcal{N}(0, \mathbf{W}), \quad (3.3)$$

$$\mathbf{v} \sim \mathcal{N}(0, \mathbf{V}). \quad (3.4)$$

Two kinds of estimation errors can be defined, the *a priori* estimation error $\mathbf{e}(k|k-1)$ and the *a posteriori* estimation error $\mathbf{e}(k)$. They are described by

$$\mathbf{e}(k|k-1) = \mathbf{x}(k) - \hat{\mathbf{x}}(k|k-1), \quad (3.5)$$

$$\mathbf{e}(k) = \mathbf{x}(k) - \hat{\mathbf{x}}(k), \quad (3.6)$$

where $\hat{\mathbf{x}}(k|k-1)$ is the *a priori* state estimate based on knowledge of the process up to step $k-1$ and $\hat{\mathbf{x}}(k)$ is the *a posteriori* state estimate based on measurements $\mathbf{z}(k)$ up to step k . The covariances of these estimation errors are defined by

$$\mathbf{P}(k|k-1) = \mathbb{E}\left\{\mathbf{e}(k|k-1)(\mathbf{e}(k|k-1))^T\right\}, \quad (3.7)$$

$$\mathbf{P}(k) = \mathbb{E}\left\{\mathbf{e}(k)(\mathbf{e}(k))^T\right\}, \quad (3.8)$$

where $\mathbf{P}(k|k-1)$ is the *a priori* estimate error covariance and $\mathbf{P}(k)$ the *a posteriori* estimate error covariance.

The first step in the *time update stage* of the EKF algorithm is to project the state and error covariance ahead by

$$\hat{\mathbf{x}}(k|k-1) = f(\hat{\mathbf{x}}(k-1), \mathbf{u}(k-1)), \quad (3.9)$$

$$\mathbf{P}(k|k-1) = \mathbf{A}(k)\mathbf{P}(k-1)\mathbf{A}^T(k) + \mathbf{W}(k-1), \quad (3.10)$$

where $\mathbf{A}(k)$ is the Jacobian of the system model equations described by

$$A_{i,j}(k) = \frac{\partial f_i}{\partial x_j} (\hat{\mathbf{x}}(k-1), \mathbf{u}(k-1), 0). \quad (3.11)$$

In the *measurement update stage*, we first compute the Kalman gain as

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{C}^T(k) \left(\mathbf{C}(k)\mathbf{P}(k|k-1)\mathbf{C}^T(k) + \mathbf{V}(k) \right)^{-1}, \quad (3.12)$$

where $\mathbf{K}(k)$ is the Kalman gain that minimizes the *a posteriori* estimate error covariance $\mathbf{P}(k)$, and $\mathbf{C}(k)$ is the Jacobian of the measurement function \hat{h} according to

$$C_{i,j}(k) = \frac{\partial \hat{h}_i}{\partial x_j} (\hat{\mathbf{x}}(k|k-1), \mathbf{u}(k), 0). \quad (3.13)$$

The state estimate and the error covariance are then updated with the measurement

$$\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k) \left(\mathbf{z}(k) - \hat{h}(\hat{\mathbf{x}}(k|k-1), \mathbf{u}(k)) \right), \quad (3.14)$$

$$\mathbf{P}(k) = (\mathbf{I} - \mathbf{K}(k)\mathbf{C}(k))\mathbf{P}(k|k-1). \quad (3.15)$$

The residual

$$\mathbf{r}(k) = \mathbf{z}(k) - \hat{h}(\hat{\mathbf{x}}(k|k-1), \mathbf{u}(k)) \quad (3.16)$$

reflects the error between the predicted measurement by the function \hat{h} and the real measurement from $\mathbf{z}(k)$. In case of a fault in one of the inputs u_j or measurements z_i , the residual r_i will diverge from its nominal value around zero, such that a fault can be detected. The change \mathbf{c} in the residual vector \mathbf{r} can then be compared to known *fault signatures* \mathbf{f} to diagnose the type and location of the fault. Fault diagnosis will be further discussed in Section 3.2.8.

3.2.3 Fault-detection filters

In order to facilitate this fault diagnosis process, fault-detection filters can be designed that yield multi-dimensional residuals that point in a particular direction, depending on the type of fault. These filters have a special choice of the observer feedback gain, such that the residual vector has certain directional properties [33]. It places the reachable subspace of each fault into invariant subspaces that do not overlap. Then, when a nonzero residual is detected, a fault can be announced and identified by projecting the residual onto each invariant subspace.

3.2.4 Unknown input observers

In case a fault occurs in a system input, an *unknown input observer* may be used, where one of the inputs u_j is taken into account as an augmented state [252]. The generation of $u_j(k)$ is usually modeled after a *random walk* process as

$$u_j(k) = u_j(k-1) + w_{u_j}(k-1), \quad (3.17)$$

with $w_{u_j}(k)$ a white noise sequence, representing the uncertainty in the input. In case of an unknown input u_j , we can rewrite (3.1) as

$$\mathbf{x}'(k) = f(\mathbf{x}'(k-1), \mathbf{u}'(k-1), \mathbf{w}'(k-1)), \quad (3.18)$$

where $\mathbf{x}' = [\mathbf{x}^T \ u_j]^T$ is the augmented state vector, $\mathbf{u}'(k)$ is the vector of known inputs, and $\mathbf{w}'(k) = [\mathbf{w}^T \ w_{u_j}]^T$ the augmented process noise vector. Correspondingly, the process covariance matrix (3.3) and the Jacobians (3.11) and (3.13) have to be modified. With regard to FDI, the unknown input observer is very useful in detecting faults in actuators or other system inputs.

3.2.5 Generalized observer scheme

In addition to using a single observer, a bank of observers can be used, excited by all outputs. This is useful if the faults inflict changes on the internal states of the process. With a bank of observers, a *dedicated observer scheme*, each driven by a different output, redundant state estimates become available, and even the detection of multiple sensor faults becomes possible. FDI using a bank of dedicated observers has been successfully applied to longitudinal vehicle control [101]. However, it is often not possible to obtain an observable system using only one input.

A *generalized observer scheme* provides a more suitable solution, which implements a bank of observers with each observer i driven by all but the i -th measured variable, as illustrated by the example in Figure 3.3(a). The residual vector \mathbf{r}_i from the i -th observer is then sensitive to faults in all but the i -th sensor. Subsequently, all but one observer (the i -th) will respond to a fault in the i -th sensor. The generated residual set will therefore be less sensitive to modeling errors and unknown disturbances, which improves the robustness of the FDI system and reduces the false alarm rate. However, the cost for this robustness is the ability to detect only a single fault at a time in one of the sensors of the system.

3.2.6 Parity equations

Parity equations are a relatively simple and straightforward approach to construct residuals that carry fault information, independent of system operating conditions and system inputs under nominal operating conditions. Parity equations require the knowledge of a fixed parameterized model that serves as a reference for the measured behavior, such that the residual is constructed by

$$\mathbf{r} = \mathbf{z} - \hat{\mathbf{y}}. \quad (3.19)$$

Under nominal (fault-free) conditions these residuals fluctuate around zero below the thresholds γ . These parity equations are violated in the event of faults that effect the measurement \mathbf{z} . A fault can be detected when one or more residuals r_i cross a threshold: $|r_i| > \gamma_i$. Parity equations are especially suitable for sensor faults [70, 184]. However, they are sensitive to noise and the residuals are always temporal, since parity equations are open-loop by nature, as illustrated in Figure 3.3(b).

Usually, application of a parity equation is combined with a state estimator, which provides the reference signal $\hat{\mathbf{y}}$. The application of parity equations for fault detection in automated longitudinal control was demonstrated by California PATH in [33] (in combination with fault detection filters) and in [101] (in combination with a dedicated observer scheme).

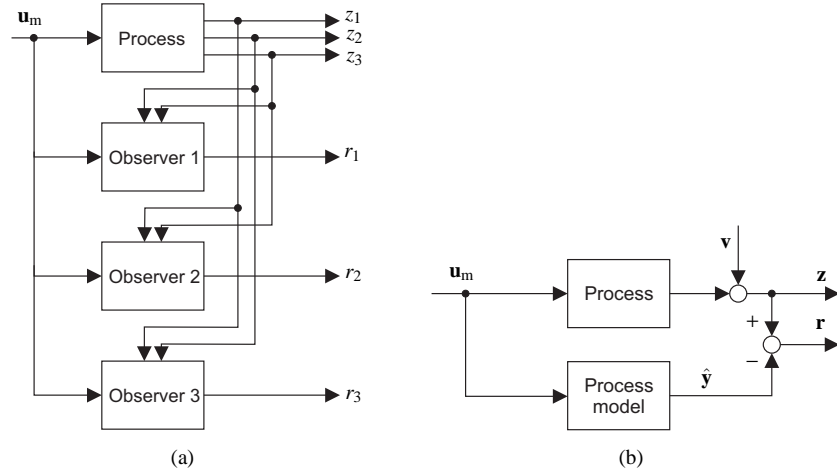


Figure 3.3: Examples of model-based FDI: (a) generalized observer scheme; (b) parity equation.

3.2.7 Change detection methods

After the generation of a residual, a fault should be detected by change detection methods. Basseville [15] discusses several methods to derive a Boolean detection vector \mathbf{c} . An important performance requirement for effective residual evaluation is the *detectability* of a fault. A basic solution is to detect a fault when the residual crosses a preassigned threshold γ . However, due to model uncertainty and measurement noise, the residual will never be exactly zero in the fault-free case, requiring a non-zero threshold. Considering a generalized observer scheme with a sensor fault in the j -th sensor, the residuals r_{ij} of the i -th observer will satisfy the following detection logic:

$$c_{ij} = \begin{cases} 0 & \text{if } |r_{ij}(k)| \leq \gamma_{ij}, \quad i = j \\ 1 & \text{if } |r_{ij}(k)| > \gamma_{ij}, \quad i \neq j \end{cases}, \quad (3.20)$$

where γ_{ij} are isolation thresholds.

As an example, Figure 3.4(a) illustrates a scenario, where μ_0 is the mean value of the residual before the change (zero), and μ_1 the *a priori* unknown mean after the change. Unfortunately, a fixed threshold may cause false and missed alarms, when it is either set too sensitive (γ_1) or too unresponsive (γ_2). Instead, adaptive thresholds can be used, depending on the operating conditions. Another option may be to filter the residual using a moving average filter, in order to reduce the signal noise. Unfortunately, all these solutions require *a priori* knowledge of the operating conditions and the residual behavior in case of a fault.

Alternatively, a statistical test can be used to check whether the residual has a zero mean and unchanged variance, assuming that the EKF residual is a white-noise signal. Since the mean of the residual after the fault occurrence is usually unknown, an appropriate test is the generalized likelihood ratio test [15]. Like most statistical detection algorithms, it uses the

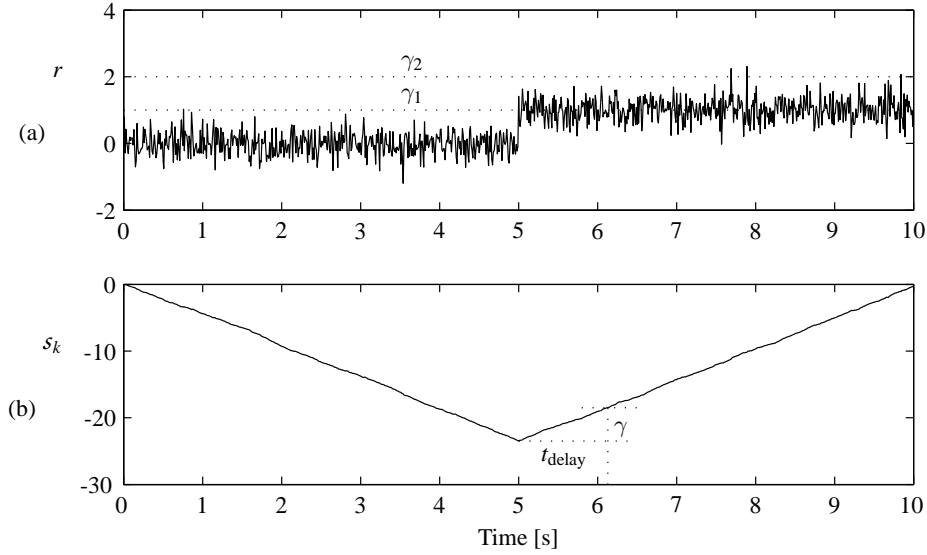


Figure 3.4: Strategies for change detection, illustrated for a change in residual mean from $\mu_0 = 0$ to $\mu_1 = 1$ at $t = 5$ s: (a) using fixed threshold γ_1 and γ_2 ; (b) using log-likelihood ratio testing, where the difference function (3.22) crosses a threshold γ after a delay t_{delay} .

log-likelihood ratio s_j^k for observations of the residual r_i from time j up to time k :

$$s_j^k = \sum_{i=j}^k \ln \frac{p_{\mu_1}(r_i)}{p_{\mu_0}(r_i)}. \quad (3.21)$$

The statistical approach is to use the maximum likelihood estimate for μ , based on observations r_i , and estimate the corresponding probability distributions p_{μ_0} and p_{μ_1} , respectively. The log-likelihood s_k is then a measure for the likelihood that the mean value has changed from μ_0 to μ_1 . A fault is detected when the difference function

$$g_k = s_k - \min_{1 \leq j \leq k} s_j \quad (3.22)$$

crosses a threshold γ , as illustrated in Figure 3.4(b). Obviously, such a test introduces a detection delay t_{delay} to allow robust detection of the residual change.

3.2.8 Fault diagnosis

After fault detection is performed, the *isolability* of faults can be defined as the ability to distinguish (isolate) specific faults [184]. A straightforward approach is to use a voting scheme to pinpoint the location of the fault using prior knowledge of particular residual changes in response to that fault. However, isolation and identification of faults is a more difficult problem than fault detection, since each fault must have a unique effect upon the residuals for correct isolation and identification. Although a single residual signal is sufficient to *detect* faults, a set of residuals (or a residual vector) is required for fault *isolation*.

Table 3.1: Example of a generalized residual set for fault isolation.

Sensor in demonstrator	\mathbf{f}	Residuals \mathbf{r}							
		r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8
Longitudinal accelerometer	f_1	0	1	1	1	1	1	1	1
Lateral accelerometer	f_2	1	0	1	1	1	1	1	1
Gyroscope (yaw rate sensor)	f_3	1	1	0	1	1	1	1	1
Wheel speed sensor 1L	f_4	1	1	1	0	1	1	1	1
Wheel speed sensor 1R	f_5	1	1	1	1	0	1	1	1
Wheel speed sensor 2L	f_6	1	1	1	1	1	0	1	1
Wheel speed sensor 2R	f_7	1	1	1	1	1	1	0	1
Steering angle sensor	f_8	1	1	1	1	1	1	1	0

To facilitate the isolation problem for model-based FDI, a structured residual set \mathbf{r} is usually generated, where each residual r_i is designed to be sensitive to a subset of faults while remaining insensitive to other faults. This can be a dedicated residual set, where each residual is sensitive to only one fault, but with this isolation method it is hard to achieve robustness to model uncertainty [185].

Instead, a generalized observer scheme uses a generalized residual set, where each residual is sensitive to all faults but one. This results in a unique fault signature with respect to the occurrence of each fault, which subsequently can be isolated by decision logic. In Table 3.1, each row represents a fault, where a “1” on the j -th row and i -th column implies that fault f_j affects the residual r_i of the i -th observer. Since each residual is sensitive to all but one sensor fault, there exists a unique combination of the residual response for each fault, as can be seen in each row in Table 3.1. Using such a residual table, any single sensor fault, f_j can be uniquely detected and isolated.

3.2.9 Considerations for model-based FDI for ADASs

Large faults are relatively easy detected by the diagnosis methods described above. However, it is also of interest to know what fault size is necessary to trigger the change detection module. In addition to isolability and detectability, an important performance indicator for FDI is therefore the fault *sensitivity*, *i.e.*, the ability of the method to detect faults of a reasonably small size [70], in order to obtain an FDI system with a low missed alarm rate.

Simultaneously, the FDI system must have a certain level of *robustness* to noise, disturbances, and modeling error, in order to have a low false alarm rate. Several authors, see the survey by Gertler [70], present methods for increasing the robustness of FDI systems, either by adaptive methods, or by accounting for the uncertainty in residual generation directly. Kanev [130] has developed a method for fault-tolerant control that deals with inaccurate information coming from the FDI scheme through the use of randomized algorithms. This highlights the importance of developing the FDI and FTC systems in an integrated approach.

In any case, an important requirement in highly dynamic systems (such as automobiles) is rapid detection and isolation of faults, *i.e.*, a small value for the detection delay t_{delay} , when using the generalized likelihood ratio test. A rapid detection will prevent instability of the system during the fault detection and possible reconfiguration of the control system. The supervisor in an FTC system must therefore maintain stability during the fault detection and reconfiguration process [229] and has to avoid large transients or discontinuities [117].



Figure 3.5: The two Smart vehicles at the test track around the VeHIL laboratory in Helmond, the Netherlands.

Since all methods have their advantages and disadvantages, a combination of different methods may lead to a better *fault detection coverage*, *i.e.*, a larger subset of \mathcal{F} that can be detected. This will improve the dependability of the fault management system and of the system as a whole. We therefore combine the generalized observer scheme and parity equations in the design of a fault-tolerant state estimation system.

3.3 Demonstrator vehicles

A demonstrator setup has been built for implementation of a fault-tolerant ADAS. This demonstrator consists of two automated Smart Fortwo vehicles, depicted in Figure 3.5.

3.3.1 Prototype instrumentation

The Smarts are used as experimental platforms for rapid control prototyping purposes, and are instrumented with electronically controlled actuators. The throttle angle is controlled by providing an electronic interface to the motor management system. The standard hydraulic brake system has been replaced by an electrohydraulic brake system, which provides a fast and accurate response and higher control bandwidth. The six-speed sequential gearbox is electronically controlled, and a steering motor is implemented for ADAS applications that require automatic steering, *e.g.*, lane keeping. In addition, communication with the driver is provided by a control lever to receive driver inputs, and a human-machine interface (HMI) to transmit information and warnings to the driver. Signal processing and control algorithms are implemented on ControlCIT, which is a real-time Linux-based, PC/104 industrial computer system, specifically developed for real-time control prototyping of ADAS applications [191]. The interfacing between ControlCIT, sensors, and actuators is implemented by a dedicated vehicle CAN bus. The instrumentation is schematically depicted in Figure 3.6.

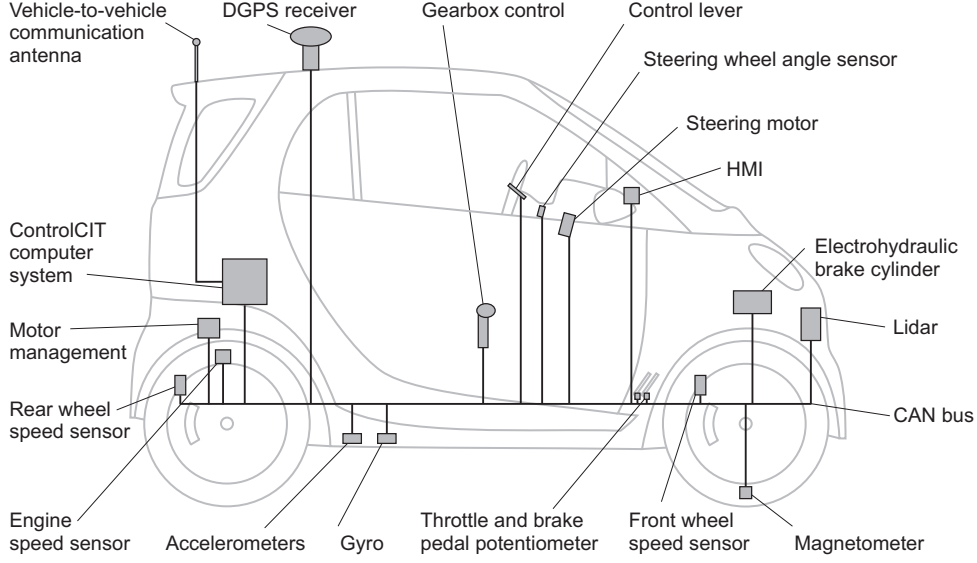


Figure 3.6: Schematic diagram of the instrumentation of the Smart.

The vehicles are equipped with several sensor systems for state estimation and environment detection. A triaxial accelerometer measures the acceleration a of the vehicle in the x , y , and z direction. A wheel encoder on each wheel ij measures the angular wheel speed $\omega_{\text{wheel},ij}$, with $i \in \{1, 2\}$ indicating the front and rear axle, and $j \in \{L, R\}$ the left and right wheel, respectively. A gyroscope is used to measure the yaw rate $\dot{\psi}$. The actuator positions are measured by sensors that give the steering wheel angle δ_{st} , throttle pedal position s_{th} , brake pedal position s_{br} , and gearbox position s_{gear} . A magnetometer is used to detect magnetic markers embedded in the road surface as an absolute position reference. Although magnets have been used for automatic vehicle control [214], they are used here merely for validation purposes. Instead, we rely on differential GPS (DGPS) to provide global position information, such as latitude ζ , longitude η , and heading ϑ in the World Geodetic System WGS84. The vehicle motion is defined in the global coordinate frame $\{G\}$, which is a right-handed Cartesian coordinate system (x, y) . These axes x and y are aligned with the longitude and latitude directions (pointing north and east respectively), and centered on a local datum (ζ_0, η_0) on the Earth. The orientation of the vehicle is indicated by the yaw angle ψ . Assuming a spherical projection, an approximate conversion from global position in the WGS84 coordinate system to the frame $\{G\}$ is given by [48]

$$x = \frac{2\pi R_{\text{earth}} (1 - e_{\text{earth}}^2) (\cos(\pi\zeta_0/180))}{360 (1 - e_{\text{earth}}^2 (\sin(\pi\zeta_0/180))^2)^{1.5}} \eta, \quad (3.23)$$

$$y = \frac{2\pi R_{\text{earth}} (1 - e_{\text{earth}}^2)}{360 (1 - e_{\text{earth}}^2 (\sin(\pi\zeta_0/180))^2)^{1.5}} \zeta, \quad (3.24)$$

$$\psi = \text{mod} \left(\frac{-(\vartheta - 90)2\pi}{360}, 2\pi \right), \quad (3.25)$$

Table 3.2: Relevant sensor characteristics of the Smart demonstrator vehicles.

Sensor	Signal	Unit	Range		Resolution	Update frequency [Hz]	Variance ^a
			Min.	Max.			
DGPS	ζ	$^\circ$	-90	90	$1.0 \cdot 10^{-7}$	1	10.0m^2
	η	$^\circ$	-180	180	$1.0 \cdot 10^{-7}$	1	10.0m^2
	ϑ	$^\circ$	0	360	0.0078125	1	0.4rad^2
Accelerometer	a_{long}	m/s^2	-19.62	19.62	0.0485	100	$2.3 \cdot 10^{-2}$
	a_{lat}	m/s^2	-19.62	19.62	0.0485	100	$3.6 \cdot 10^{-2}$
Gyroscope	$\dot{\psi}$	rad/s	-1.745	1.745	0.000244	100	$1.0 \cdot 10^{-6}$
Wheel encoder	ω_{ij}	m/s	-40	40	0.0146	100	$2.5 \cdot 10^{-4}$
Steer angle encoder	δ_s	$^\circ$	-780	780	0.1	100	$1.0 \cdot 10^{-6}$
	$\dot{\delta}_s$	$^\circ/\text{s}$	0	1016	4.0	100	$5.0 \cdot 10^{-3}$
Lidar	r_{lidar}	m	0	100	0.1	11	$3.0 \cdot 10^{-2}$
	ϕ_{lidar}	$^\circ$	-25.4	25.4	0.2	11	$3.0 \cdot 10^{-2}$
	\dot{r}_{lidar}	m/s	-60	20	0.1	11	$1.0 \cdot 10^{-2}$
VVC	\mathbf{x}		0	400		50	
Magnetometer	x, y	m			$8.0 \cdot 10^{-4}$	500	$5.0 \cdot 10^{-5}$

^a The unit of the variance is the square of the parameter unit, unless indicated otherwise.

where $R_{\text{earth}} = 6378137\text{ m}$ is the Earth radius and $e_{\text{earth}} = 0.08181919$ is the eccentricity of the Earth.

The host vehicle state measurements are defined in a frame $\{G'\}$ with its origin fixed at the origin of frame $\{G\}$, but its direction vectors (denoted as ‘lat’ and ‘long’) aligned with the central principal axes of the vehicle chassis¹, as illustrated in Figure 3.7. Considering only vehicle motion in the horizontal plane, the transformation from $\{G'\}$ to $\{G\}$ can be obtained by pre-multiplication of ${}^G\mathbf{x}$ with the transformation matrix

$${}^G\mathbf{R} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix}. \quad (3.26)$$

For the purpose of environment sensing the vehicles are equipped with a lidar sensor, which provides the range r , the range rate \dot{r} , and the angle ϕ to an object, measured in a vehicle-fixed polar coordinate system $\{P\}$. The lidar can track up to five targets simultaneously. In addition, the vehicles are equipped with wireless local area network modules that are able to receive and to transmit information to other vehicles within a range of several hundreds of meters, depending on environmental conditions. Table 3.2 provides some basic characteristics for the above-mentioned sensor systems.

3.3.2 Vehicle modeling

A vehicle model is required to implement a system for vehicle state estimation and model-based FDI. Since we require a reliable model for robust FDI, a nonlinear two-track vehicle model is set up, including engine, driveline, chassis, body, and tire dynamics. Considering

¹Note that the direction vectors in the frame $\{G'\}$ refer to the *lateral* and *longitudinal* direction of the vehicle, and should not be confused with the terms *latitude* and *longitude* that are used in the WGS84 frame. Refer to the Glossary for more information on the coordinate systems used in this thesis.

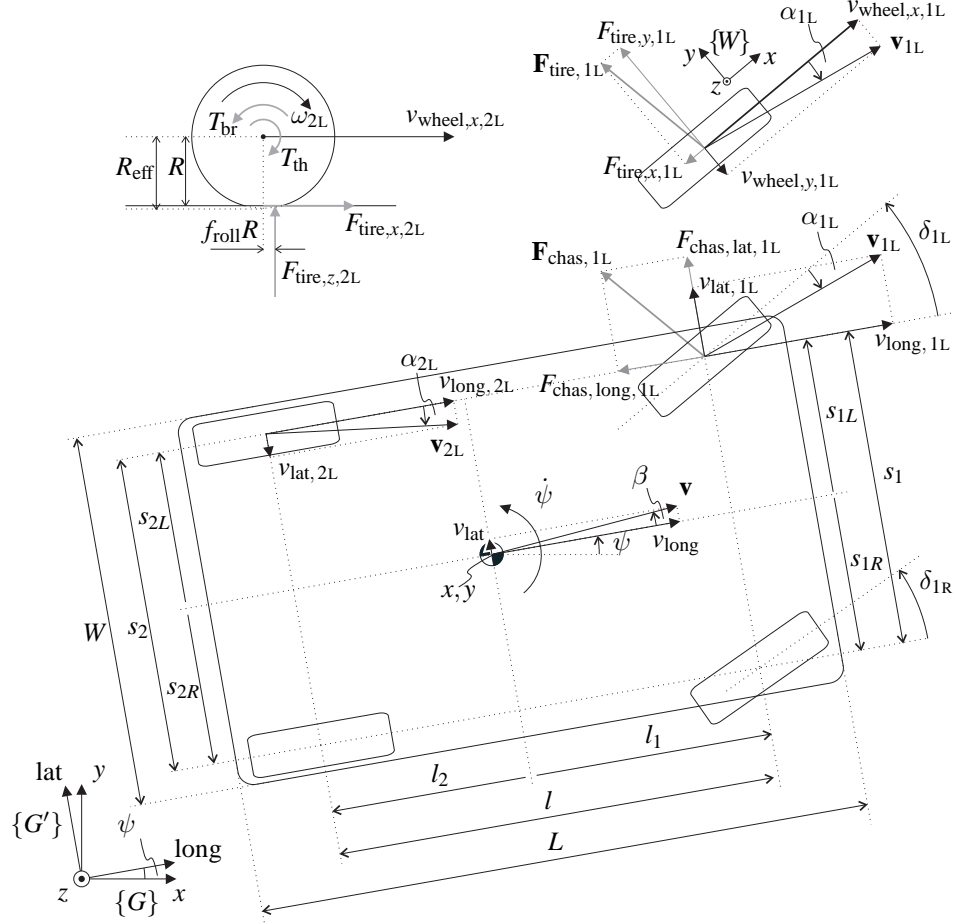


Figure 3.7: Definition of the coordinate frames: (top left) side view of the (driven) rear left wheel; (top right) top view of the (steered) front left wheel; (bottom) top view of the vehicle chassis. Black bold arrows indicate velocities (or rotational velocities), grey bold arrows indicate forces (or torques), and black thin arrows indicate dimensions (or angles).

only motion in the horizontal plane, the discrete-time equations of motion are² [177]:

$$m \dot{v}_{\text{long}}^G(k) - m \dot{\psi}^G(k) \dot{v}_{\text{lat}}^G(k) = \sum \dot{F}_{\text{long}}^G(k) - \dot{F}_{\text{air, long}}^G(k) - \dot{F}_{\text{grav, long}}^G(k), \quad (3.27)$$

$$m \dot{v}_{\text{lat}}^G(k) + m \dot{\psi}^G(k) \dot{v}_{\text{long}}^G(k) = \sum \dot{F}_{\text{lat}}^G(k), \quad (3.28)$$

$$I_z \ddot{\psi}^G(k) = \sum \dot{M}_z^G(k), \quad (3.29)$$

²For the ease of notation we will denote the finite difference approximation of velocity and acceleration signals by their time derivative, e.g., the longitudinal component of the vehicle's acceleration is denoted by \dot{v}_{long} . With sample time t_s , we then have $\dot{v}_{\text{long}}(k) = \frac{v_{\text{long}}(kt_s) - v_{\text{long}}((k-1)t_s)}{t_s}$, where the left-hand side is the sampled signal, and the right-hand side refers to continuous-time signals.

Table 3.3: Smart vehicle characteristics.

Parameter	Symbol	Value	Unit
Vehicle length	L	2500	mm
Vehicle width	W	1515	mm
Wheelbase	l	1812	mm
Distance front axle to center of gravity	l_1	1022	mm
Distance rear axle to center of gravity	l_2	-790	mm
Height of center of gravity	h_0	620	mm
Track width front axle	s_1	1286	mm
Track width rear axle	s_2	1354	mm
Vehicle mass (empty)	m	820	kg
Moment of inertia around z axis	I_z	609	kgm ²
Wheel moment of inertia	I_{wheel}	1.4	kgm ²
Gravitational acceleration	g	9.81	m/s ²
Loaded tire radius	R	285.0	mm
Effective tire radius	R_{eff}	290.8	mm
Maximum engine power (rear-wheel drive)	P_{eng}	40	kW
Maximum engine torque	T_{eng}	80	Nm
Maximum engine speed	n_{eng}	6000	rpm
Aerodynamic drag coefficient	C_w	0.37	-
Frontal area	A_x	2.32	m ²
Air density	ϱ_{air}	1.202	kg/m ³

where m and I_z are the vehicle mass and moment of inertia, g the gravitational acceleration, and $\sum {}^G F_{\text{long}}(k)$, $\sum {}^G F_{\text{lat}}(k)$, and $\sum {}^G M_z(k)$ are the combined tire forces and moments in the vehicle chassis frame $\{G'\}$ at step k . The longitudinal components of the air and gravitational resistance forces are

$${}^G F_{\text{air, long}}(k) = 0.5 \varrho_{\text{air}} C_w A_x {}^G v_{\text{long}}^2(k), \quad (3.30)$$

$${}^G F_{\text{grav, long}}(k) = mg \sin({}^G \alpha_{\text{road}}(k)), \quad (3.31)$$

with ϱ_{air} the air density, C_w the aerodynamic drag coefficient, A_x the frontal area of the vehicle, and ${}^G \alpha_{\text{road}}(k)$ the road inclination. Coast-down tests provide rolling and air resistance coefficients, such that the parameters involved in ${}^G F_{\text{air, long}}(k)$ and ${}^G F_{\text{grav, long}}(k)$ can be calculated, see Table 3.3.

Equations (3.27)-(3.29) can be solved using the following model. The vehicle velocities (${}^G v_{\text{long}}(k-1)$, ${}^G v_{\text{lat}}(k-1)$, and yaw rate ${}^G \dot{\psi}(k-1)$), the wheel angular velocities ${}^w \omega_{ij}(k)$, and the vertical tire forces $F_{\text{tire, } z, ij}(k-1)$ are used at time step k as initial conditions for the model. The velocity components at each wheel location are then

$${}^G v_{\text{long, } ij}(k) = {}^G v_{\text{long}}(k-1) - {}^G \dot{\psi}(k-1) {}^L s_{ij}, \quad (3.32)$$

$${}^G v_{\text{lat, } ij}(k) = {}^G v_{\text{lat}}(k-1) + {}^G \dot{\psi}(k-1) {}^L l_i, \quad (3.33)$$

with ${}^L l_i$ and ${}^L s_{ij}$ being the longitudinal and lateral distance from the vehicle center of gravity, as measured in the vehicle-fixed frame $\{L\}$ ³. With wheel angles δ_{ij} the velocities in the

³For notational convenience the indication of the sample step k and coordinate frame have been omitted in the remainder of this section, unless the discrete time step is different from k .

wheel frame $v_{\text{wheel},x,ij}$ and $v_{\text{wheel},y,ij}$ are then

$$\begin{bmatrix} v_{\text{wheel},x,ij} \\ v_{\text{wheel},y,ij} \end{bmatrix} = \begin{bmatrix} \cos \delta_{ij} & \sin \delta_{ij} \\ -\sin \delta_{ij} & \cos \delta_{ij} \end{bmatrix} \begin{bmatrix} v_{\text{long},ij} \\ v_{\text{lat},ij} \end{bmatrix}. \quad (3.34)$$

For a tire with effective radius R_{eff} , the tire slip angles \varkappa and α are then given by

$$\varkappa_{ij} = -\frac{v_{\text{wheel},x,ij} - \omega_{ij}R_{\text{eff}}}{v_{\text{wheel},x,ij}}, \quad (3.35)$$

$$\alpha_{ij} = \tan^{-1} \left(\frac{v_{\text{wheel},y,ij}}{v_{\text{wheel},x,ij}} \right). \quad (3.36)$$

For longitudinal and lateral tire slip angles \varkappa_{ij} and α_{ij} , camber angles v_{ij} , and vertical tire forces $F_{\text{tire},z,ij}$, Pacejka's Magic Formula [177] gives

$$F_{\text{tire},x,ij}, F_{\text{tire},y,ij}, M_{\text{tire},z,ij} = \text{Magic Formula}(\varkappa_{ij}, \alpha_{ij}, v_{ij}, F_{\text{tire},z,ij}), \quad (3.37)$$

where we assume constant road surface conditions. The reader is referred to [278] for an in-depth discussion of tire force and road friction estimation. These tire forces and moments for each wheel are then transformed to the chassis frame by

$$\begin{bmatrix} F_{\text{chas, long},ij} \\ F_{\text{chas, lat},ij} \end{bmatrix} = \begin{bmatrix} \cos \delta_{ij} & -\sin \delta_{ij} \\ \sin \delta_{ij} & \cos \delta_{ij} \end{bmatrix} \begin{bmatrix} F_{\text{tire},x,ij} \\ F_{\text{tire},y,ij} \end{bmatrix}. \quad (3.38)$$

The equations of motion (3.27)-(3.29) can then be solved by summing the chassis forces and moments:

$$\sum F_{\text{long}} = \sum_{i,j} F_{\text{chas, long},ij}, \quad (3.39)$$

$$\sum F_{\text{lat}} = \sum_{i,j} F_{\text{chas, lat},ij}, \quad (3.40)$$

$$\sum M_z = \sum_{i,j} (-F_{\text{chas, long},ij}s_{ij} + F_{\text{chas, lat},ij}l_i + M_{\text{tire},z,ij}). \quad (3.41)$$

Introducing a wheel moment of inertia I_{wheel} , drive torque T_{th} , brake torque T_{br} , and rolling resistance coefficient f_{roll} , the motion equation for the wheel is obtained as

$$I_{\text{wheel}}\dot{\omega}_{ij} = T_{\text{th},ij} - T_{\text{br},ij} - F_{\text{tire},x,ij}R - F_{\text{tire},z,ij}f_{\text{roll}}R, \quad (3.42)$$

which gives ω_{ij} for the next integration time step $k+1$ with sample time t_s :

$$\omega_{ij}(k+1) = \omega_{ij}(k) + \dot{\omega}_{ij}(k)t_s. \quad (3.43)$$

The corresponding characteristics of the driveline and brake dynamics are also summarized in Table 3.3. The vertical tire forces for the next integration time step can then be found by

$$F_{\text{tire},z,1L}(k+1) = \frac{l_2}{2l}mg + \frac{(F_{\text{chas},y,1L}(k) + F_{\text{chas},y,1R}(k))h_1}{s_1} - \frac{h_0}{2l}m\dot{v}_{\text{long}}(k), \quad (3.44)$$

$$F_{\text{tire},z,1R}(k+1) = \frac{l_2}{2l}mg - \frac{(F_{\text{chas},y,1L}(k) + F_{\text{chas},y,1R}(k))h_1}{s_1} - \frac{h_0}{2l}m\dot{v}_{\text{long}}(k), \quad (3.45)$$

$$F_{\text{tire},z,2L}(k+1) = \frac{l_1}{2l}mg + \frac{(F_{\text{chas},y,2L}(k) + F_{\text{chas},y,2R}(k))h_2}{s_2} + \frac{h_0}{2l}m\dot{v}_{\text{long}}(k), \quad (3.46)$$

$$F_{\text{tire},z,2R}(k+1) = \frac{l_1}{2l}mg - \frac{(F_{\text{chas},y,2L}(k) + F_{\text{chas},y,2R}(k))h_2}{s_2} + \frac{h_0}{2l}m\dot{v}_{\text{long}}(k), \quad (3.47)$$

where the first term is the static load distribution, the second term the load transfer due to roll, and the third term the load transfer due to pitch (neglecting suspension characteristics). Here, l is the wheelbase, h_0 is the height of the center of gravity, and h_i the height of the roll center at the i -th axle. Values for these parameters are also given in Table 3.3.

3.4 State estimation by extended Kalman filtering

Based on the nonlinear vehicle model, as presented in the previous section, we will now derive an EKF in order to estimate the vehicle state.

3.4.1 Model equations

The vehicle state is characterized by the position, velocity, and acceleration in three directions of the vehicle frame $\{G'\}$ (x , y , and rotation around z). In addition, the inertial sensors (accelerometers and gyro) are susceptible to bias, due to changes in orientation (*e.g.*, road banking). The state vector \mathbf{x} is therefore defined to contain the following components:

$G'x$	x position,
$G'y$	y position,
$G'\psi$	yaw angle,
$G'v_{\text{long}}$	velocity in longitudinal direction,
$G'v_{\text{lat}}$	velocity in lateral direction,
$G'\dot{\psi}$	yaw rate,
$G'\dot{v}_{\text{long}}$	differentiated velocity in longitudinal direction,
$G'\dot{v}_{\text{lat}}$	differentiated velocity in lateral direction,
$G'\ddot{\psi}$	yaw acceleration,
$b_{\text{acc, long}}$	bias state for longitudinal accelerometer,
$b_{\text{acc, lat}}$	bias state for lateral accelerometer,
b_{gyro}	bias state for gyro.

Since it is difficult to derive the engine and brake torque in a reliable way, the use of (3.42)-(3.43) is not recommended for deriving the wheel dynamics. Instead the measurements of the four wheel speeds ${}^W\omega_{\text{wheel},m,i,j}$ are used as input. In addition, the wheel angles $G'\delta_{i,j}$ can be derived directly from the steering wheel sensor measurement $\delta_{\text{st},m}$. Together with estimated values for the vertical tire forces ${}^WF_{\text{tire},z,i,j}$, the filter's input vector \mathbf{u} then contains the following components:

$G'\hat{\delta}_{1L,m}$	measured wheel angle for wheel 1L,
$G'\hat{\delta}_{1R,m}$	measured wheel angle for wheel 1R,
${}^W\omega_{\text{wheel},1L,m}$	measured wheel speed for wheel 1L,
${}^W\omega_{\text{wheel},1R,m}$	measured wheel speed for wheel 1R,
${}^W\omega_{\text{wheel},2L,m}$	measured wheel speed for wheel 2L,
${}^W\omega_{\text{wheel},2R,m}$	measured wheel speed for wheel 2R,
${}^WF_{\text{tire},z,1L}$	estimated vertical tire force for wheel 1L,
${}^WF_{\text{tire},z,1R}$	estimated vertical tire force for wheel 1R,
${}^WF_{\text{tire},z,2L}$	estimated vertical tire force for wheel 2L,
${}^WF_{\text{tire},z,2R}$	estimated vertical tire force for wheel 2R.

Based on the assumption of Gaussian distributed white noise, we can now derive the following nonlinear discrete-time state equations:

$$\begin{bmatrix} {}^G x(k) \\ {}^G y(k) \\ {}^G \psi(k) \\ {}^{G'} v_{\text{long}}(k) \\ {}^{G'} v_{\text{lat}}(k) \\ {}^G \dot{\psi}(k) \\ {}^{G'} v_{\text{long}}(k) \\ {}^{G'} v_{\text{lat}}(k) \\ {}^G \dot{\psi}(k) \\ b_{\text{acc, long}}(k) \\ b_{\text{acc, lat}}(k) \\ b_{\text{gyro}}(k) \end{bmatrix} = \begin{bmatrix} {}^G x(k-1) + t_s {}^{G'} v_{\text{long}}(k-1) \cos({}^G \psi(k-1)) \\ \quad - t_s {}^{G'} v_{\text{lat}}(k-1) \sin({}^G \psi(k-1)) + w_x(k-1) \\ {}^G y(k-1) + t_s {}^{G'} v_{\text{long}}(k-1) \sin({}^G \psi(k-1)) \\ \quad + t_s {}^{G'} v_{\text{lat}}(k-1) \cos({}^G \psi(k-1)) + w_y(k-1) \\ {}^G \psi(k-1) + t_s {}^G \dot{\psi}(k-1) + w_\psi(k-1) \\ {}^{G'} v_{\text{long}}(k-1) + t_s {}^{G'} v_{\text{long}}(k-1) + w_{v_{\text{long}}}(k-1) \\ {}^{G'} v_{\text{lat}}(k-1) + t_s {}^{G'} v_{\text{lat}}(k-1) + w_{v_{\text{lat}}}(k-1) \\ {}^G \dot{\psi}(k-1) + t_s {}^G \dot{\psi}(k-1) + w_{\dot{\psi}}(k-1) \\ f(\mathbf{x}(k-1), \mathbf{u}(k-1)) + w_{v_{\text{long}}}(k-1) \\ f(\mathbf{x}(k-1), \mathbf{u}(k-1)) + w_{v_{\text{lat}}}(k-1) \\ f(\mathbf{x}(k-1), \mathbf{u}(k-1)) + w_{\dot{\psi}}(k-1) \\ b_{\text{acc, long}}(k-1) + w_{\text{acc, x}}(k-1) \\ b_{\text{acc, lat}}(k-1) + w_{\text{acc, y}}(k-1) \\ b_{\text{gyro}}(k-1) + w_{\text{gyro}}(k-1) \end{bmatrix}, \quad (3.48)$$

where $f(\cdot)$ represents the model (3.27)-(3.47).

Using the available sensors, the following measurement signals are defined:

${}^G x_{\text{GPS}}$	x coordinate obtained from GPS longitude, according to (3.23),
${}^G y_{\text{GPS}}$	y coordinate obtained from GPS latitude, according to (3.24),
${}^G \psi_{\text{GPS}}$	yaw angle ψ obtained from GPS heading, according to (3.25),
${}^{G'} a_{\text{long, m}}$	longitudinal acceleration obtained from accelerometer,
${}^{G'} a_{\text{lat, m}}$	lateral acceleration obtained from accelerometer,
${}^G \dot{\psi}_{\text{m}}$	yaw rate obtained from gyroscope.

The measurement equations are defined as follows⁴:

$$\begin{bmatrix} {}^G x_{\text{GPS}}(k) \\ {}^G y_{\text{GPS}}(k) \\ {}^G \psi_{\text{GPS}}(k) \\ {}^{G'} a_{\text{long, m}}(k) \\ {}^{G'} a_{\text{lat, m}}(k) \\ {}^G \dot{\psi}_{\text{m}}(k) \end{bmatrix} = \begin{bmatrix} {}^G x(k - N_{\text{GPS}}) + v_{x_{\text{GPS}}}(k) \\ {}^G y(k - N_{\text{GPS}}) + v_{y_{\text{GPS}}}(k) \\ {}^G \psi(k - N_{\text{GPS}}) + v_{\psi_{\text{GPS}}}(k) \\ {}^{G'} v_{\text{long}}(k) - {}^{G'} v_{\text{lat}}(k) {}^G \dot{\psi}(k) + b_{\text{acc, long}}(k) + v_{\text{acc, x}}(k) \\ {}^{G'} v_{\text{lat}}(k) + {}^{G'} v_{\text{long}}(k) {}^G \dot{\psi}(k) + b_{\text{acc, lat}}(k) + v_{\text{acc, y}}(k) \\ {}^G \dot{\psi}(k) + b_{\text{gyro}}(k) + v_{\text{gyro}}(k) \end{bmatrix}. \quad (3.49)$$

Due to GPS signal processing, GPS has an inherent latency of around 0.12 s, which is included in the measurement equations by the parameter N_{GPS} . With a time step t_s of 0.01 s, N_{GPS} is thus equal to 12. Another property of GPS is that the measurement updates once every 1 second, whereas the other vehicle state sensors have an update rate of at least 100 Hz, as shown in Table 3.2. In case of updates from the accelerometers and gyro, (3.49) reduces to the last three lines. This requires careful implementation of the filter, as discussed next.

3.4.2 Observability, implementation, and tuning of the EKF

A dynamic system is said to be observable if it is possible to uniquely reconstruct the state information, based on the model of a system given the inputs and outputs of the system.

⁴The reader should not confuse velocity signals v and elements v of the noise vector \mathbf{v} in (3.49) and (3.57).

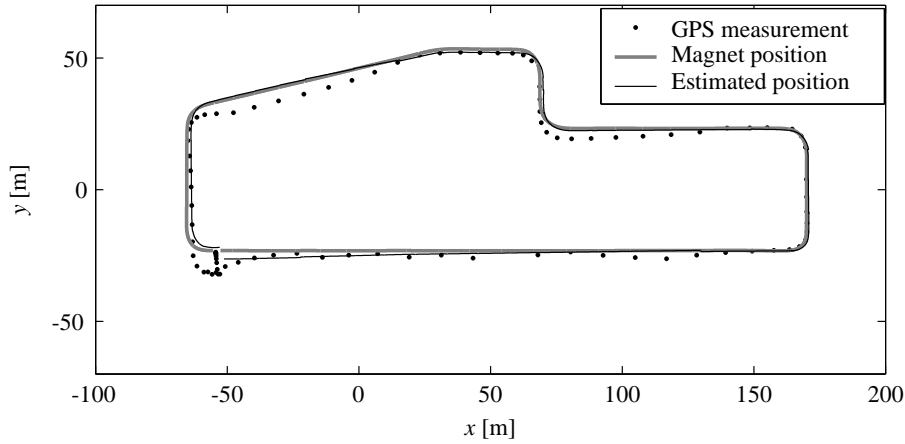


Figure 3.8: Comparison of real and estimated position for one lap around a test track.

For a linear time-invariant state space model (2.14) with n states the observability can be proven by requiring that the observability matrix $\mathbf{O} = [\mathbf{C}^T \ ; \ \mathbf{A}^T \mathbf{C}^T \ ; \ \dots \ ; \ (\mathbf{A}^T)^{n-1} \mathbf{C}^T]^T$ has rank n . However, for the nonlinear vehicle model (3.27)-(3.47) \mathbf{O} cannot be calculated in a straightforward way, and global observability is difficult to prove. The model can therefore be linearized at each time step k of the EKF operation. Subsequently, the observability matrix \mathbf{O} , is calculated for each linearization in some test runs of the EKF. It is shown in [96] that \mathbf{O} has full rank at each step k , in case the vehicle is moving, and the linearized model is thus locally observable.

The EKF is implemented in ADVANCE, which is a MATLAB/SIMULINK toolbox for simulation and analysis of combined vehicle dynamics and powertrain systems [45]. Tuning of the EKF requires knowledge about the covariances of the model disturbances. The measurement covariance matrix \mathbf{V} is implemented as a diagonal matrix with the diagonal elements set equal to the variance of the measurement noise, as indicated in Table 3.2, *i.e.*, $\mathbf{V} = \text{diag}(10, 10, 0.4, 2.3 \cdot 10^{-2}, 3.6 \cdot 10^{-2}, 1.0 \cdot 10^{-6})$. The process noise covariance matrix \mathbf{W} is tuned by adjusting the variance of individual states proportional to the variance of corresponding measurements. As an example, the process noise of the position states (x, y) is chosen much smaller than the GPS noise variance. On the other hand, the process noise of the yaw rate $\dot{\psi}$ is set larger than the variance of the gyro, since the gyro is a very accurate sensor. Therefore we choose $\mathbf{W} = \text{diag}(1.0 \cdot 10^{-4}, 1.0 \cdot 10^{-4}, 1.0 \cdot 10^{-5}, 1.0 \cdot 10^{-7}, 1.0 \cdot 10^{-7}, 6.0 \cdot 10^{-7}, 1.0 \cdot 10^{-2}, 1.0 \cdot 10^{-3}, 1.0 \cdot 10^{-3}, 1.0 \cdot 10^{-12}, 1.0 \cdot 10^{-8}, 1.0 \cdot 10^{-8})$.

To initialize the EKF, all states are set to zero, and an initialization sequence of 60 s is applied, where the process noise w_i of the position states in \mathbf{W} is chosen at a much larger value. This allows for the global position estimate to converge to a reliable value. Different values for the \mathbf{V} matrix are used for vehicle state sensor updates and GPS updates. In addition, since the tire dynamic model does not work properly at low speed, a simple low speed odometric model is used for speeds below 2 m/s.

Table 3.4: Accuracy of the state estimation in terms of the error covariance.

State x_i	Error covariance $P_{i,j=i}$
$\sqrt{c_x^2 + c_y^2}$	1.6
c_ψ	$1.0 \cdot 10^{-3}$
$c'_{v_{\text{long}}}$	$6.6 \cdot 10^{-5}$
c'_ψ	$1.0 \cdot 10^{-6}$
$c'_{a_{\text{long}}}$	$1.9 \cdot 10^{-2}$
$c'_{a_{\text{lat}}}$	$1.2 \cdot 10^{-3}$
$b_{\text{acc, long}}$	$1.9 \cdot 10^{-4}$
$b_{\text{acc, lat}}$	$1.5 \cdot 10^{-3}$
b_{gyro}	$1.0 \cdot 10^{-14}$

3.4.3 Validation of the state estimation

The performance of the state estimation has been evaluated on a test track in Helmond, the Netherlands (see Figure 3.5), where the road surface is instrumented with magnetic markers with known coordinates. Figure 3.8 shows the results of the estimated position for one lap, compared to the actual position, as obtained from the magnetic markers with an accuracy of ± 0.04 m. Figure 3.9 shows the vehicle states estimated by the EKF, as well as the corresponding measurements.

The quality of these estimates is identified by comparing them to a ground truth reference. Table 3.4 shows the variance of the estimation error $\mathbf{y} - \hat{\mathbf{y}}$, where the estimates are given by (3.49) and the reference state \mathbf{y} is provided by the magnet measurements. Compared to the noise variance of the measurements \mathbf{z} in Table 3.2, the EKF provides more accurate state information. Especially the error in the estimate for the global position (c_x, c_y) has decreased significantly, which is important for navigation purposes. Furthermore, previously unavailable vehicle states such as the lateral velocity v_{lat} , tire forces $\mathbf{F}_{\text{tire},ij}$, and tire slip angles α_{ij} can be estimated, which is useful for vehicle stability control systems. The acceleration estimates have also improved, due to the bias estimation. However, they still show some high-frequency noise, due to the noisy system inputs (the wheel speed measurements) and the feedback of the noisy acceleration measurements. The signals \hat{a}_{long} and \hat{a}_{lat} are therefore post-filtered with a low-pass Butterworth filter to give a satisfactory result.

Figure 3.9 shows that the update of the GPS heading lags behind. In addition, Figure 3.8 shows that at some instances during the test drive errors occur in the GPS measurement, due to the disturbances discussed in Chapter 2. Although this error is not due to mechanical failure, it is considered as a fault, since the error in the GPS measurement increases beyond what is considered normal noise variance. This could be solved by increasing the GPS-related variance in the measurement covariance matrix \mathbf{V} , but this could also cause the estimated position to diverge over time. Since GPS faults can be quite large and sudden (e.g., when driving into a tunnel), the inherent trade-off between robustness and accuracy of the EKF cannot be solved using a fixed measurement covariance matrix. In the next section, a method for fault detection of sensor faults is therefore applied that also provides analytical redundancy, in case sensors are temporarily unavailable.

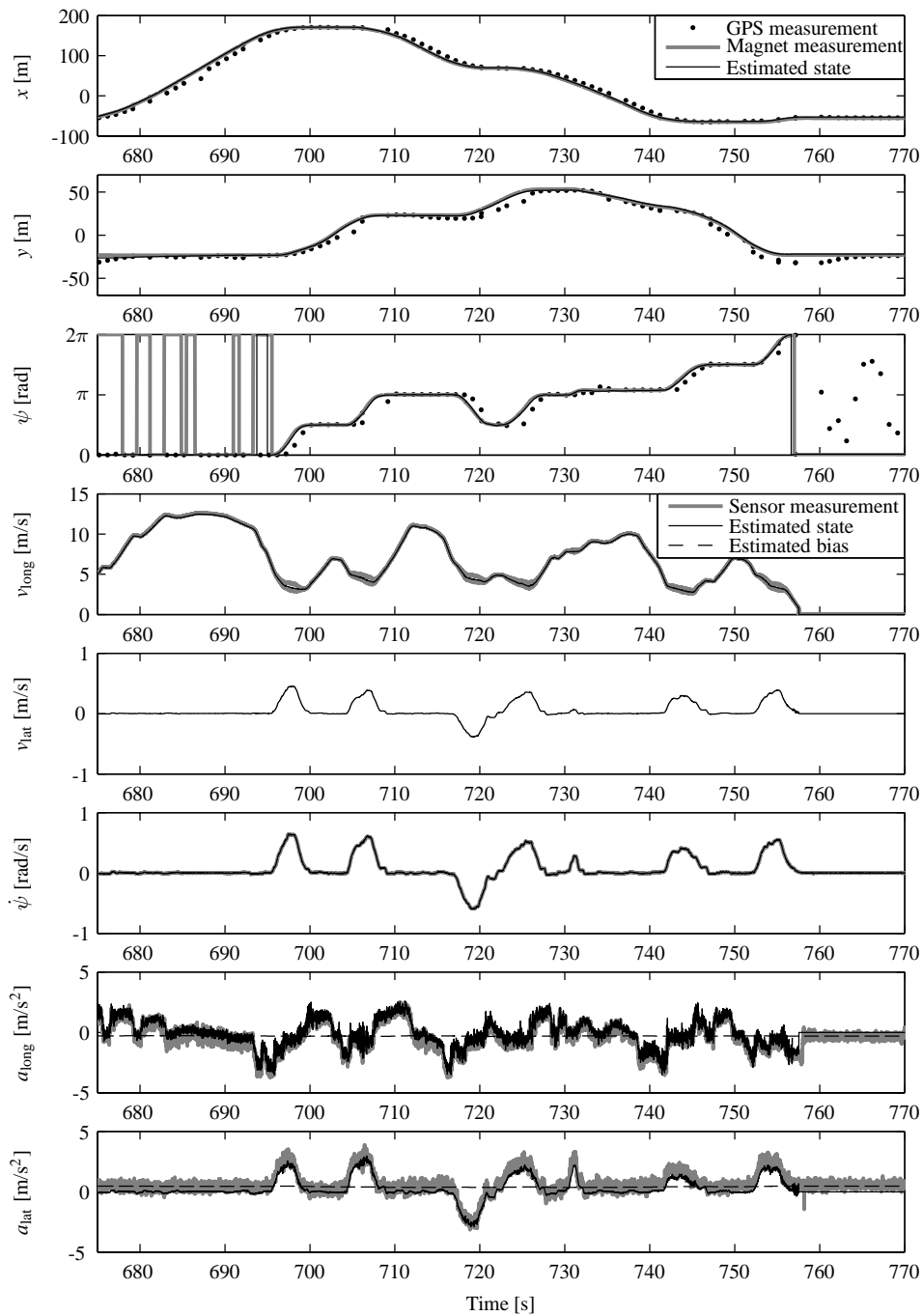


Figure 3.9: Measurements, ground truth magnet position, and estimated states. From top to bottom: position ${}^c x$, position ${}^c y$, heading ${}^c \psi$, longitudinal velocity ${}^c v_{\text{long}}$, lateral velocity ${}^c v_{\text{lat}}$, yaw rate ${}^c \dot{\psi}$, longitudinal acceleration ${}^c a_{\text{long}}$, and lateral acceleration ${}^c a_{\text{lat}}$.

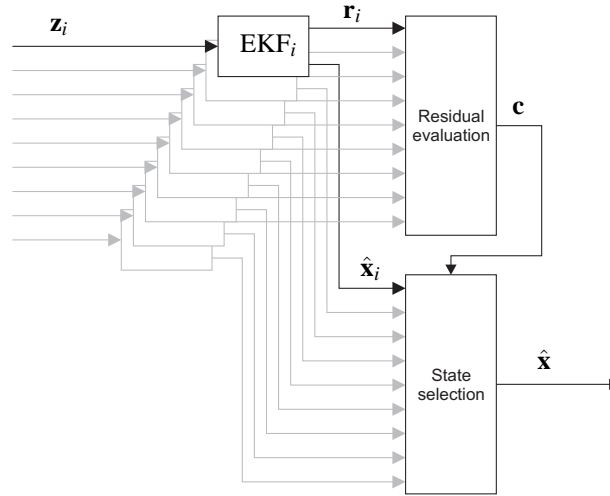


Figure 3.10: Generalized observer scheme for the sensor fault management system.

3.5 Fault management of vehicle state sensor faults

Although estimation of position by GPS is prone to faults, vehicle state sensors, such as wheel speed sensors, are generally quite reliable. However, with increasing automation, faults should be quickly diagnosed and managed. In this section we therefore present a model-based FDI system for sensor fault diagnosis that is robust to the system's disturbances. Furthermore, analytical redundancy is provided by a reliable estimate for the faulty sensor signal to maintain fault-operational behavior of the ADAS. With regard to the fault occurrence, we assume that, apart from GPS, only a single sensor fault occurs at any time instant, due to the high hardware reliability. Furthermore, all sensor faults can be modeled additively.

3.5.1 Generalized observer scheme of EKFs

A generalized observer scheme of EKFs is developed, where each EKF is driven by all but one sensor measurement, based on the EKF developed in the previous section. Since the Smart demonstrator vehicle is instrumented with eight vehicle state sensors (two accelerometers, a gyro, four wheel speed sensors, and a steering angle sensor), a generalized observer scheme of nine EKFs is developed, as illustrated in Figure 3.10. The first filter EKF_0 uses all sensors in the nominal case, whereas the other eight filters EKF_1 to EKF_8 each use all but one sensor. EKF_1 to EKF_3 are filters of the type presented in Section 3.2.2, whereas filters EKF_4 to EKF_8 are unknown input observers, as presented in Section 3.2.4. In case of a fault in sensor i , the output from the residual evaluation block will indicate which sensor fault has occurred and subsequently select the state estimate from the corresponding EKF_i (*i.e.*, the one that uses all but the faulty sensor measurement) as the fault-free output.

For GPS faults, separate fault detection logic is applied, since these faults occur much more often, and the residuals have a different behavior from that of the generalized observer scheme. The residuals are created using parity equations by comparison of the estimated

position $[\hat{x} \ \hat{y} \ \hat{\psi}]^T$ and the GPS measurement vector $[x_{\text{GPS}} \ y_{\text{GPS}} \ \psi_{\text{GPS}}]^T$. The resulting residuals

$${}^G \mathbf{r}_{\text{GPS}} = \begin{bmatrix} {}^G x_{\text{GPS}} \\ {}^G y_{\text{GPS}} \\ {}^G \psi_{\text{GPS}} \end{bmatrix} - \begin{bmatrix} {}^G \hat{x} \\ {}^G \hat{y} \\ {}^G \hat{\psi} \end{bmatrix} \quad (3.50)$$

are then transformed to the vehicle frame $\{G'\}$:

$${}^{G'} \mathbf{r}_{\text{GPS}} = {}_G^{G'} \mathbf{R} {}^G \mathbf{r}_{\text{GPS}}. \quad (3.51)$$

Subsequently, fault detection is performed using velocity-dependent thresholds. In case a GPS fault occurs, a redundant position estimate is provided by the EKF, using only measurement updates from the accelerometers and gyroscope by considering the last three lines in (3.49). The result of this analytical redundancy can be seen in Figure 3.8. Due to multi-path reflections near buildings, the GPS measurement suddenly changes at the lower-left corner of the track, which is subsequently recognized and handled by the GPS fault management mechanism.

3.5.2 Sensor fault detection and isolation

A significant change in the residuals \mathbf{r}_{ij} for each EKF i and each sensor j is detected with the generalized likelihood ratio test (3.22). The change detection c_{ij} , a Boolean value, then indicates the change of a residual vector for EKF i . In case of a sensor fault in sensor j , there is a unique pattern of the generalized residual set and therefore this sensor fault can be accurately isolated, as was shown in Table 3.1. A sensor fault will be declared only if a change is detected in *all* residuals that are sensitive to this sensor fault. Therefore, a generalized observer scheme demonstrates good robustness to noise and disturbances. Conversely, in order to improve the sensitivity of the FDI process, the generalized residual set in Table 3.1 may be altered, when a specific observer i is insensitive to a fault in sensor j .

3.5.3 Reconfiguration of state estimation

Since each EKF i in the generalized observer scheme uses all but one sensor signal, there is an estimate generated from each EKF i for the sensor measurement $j = i$ it does not use, providing analytical redundancy for this sensor. A fault management system is integrated into the designed FDI system for recognizing and handling the diagnosed fault. The basic scheme of this fault management system is shown in Figure 3.10. From the nine EKFs, nine estimated state vectors $\hat{\mathbf{x}}_0, \dots, \hat{\mathbf{x}}_i, \dots, \hat{\mathbf{x}}_8$ are obtained, where $i = 0$ represents the nominal case, and $i = 1, \dots, 8$ correspond to the eight sensors from Table 3.1. The change detection vector \mathbf{c} from the residual evaluation block is then used in the state selection block to switch to $\hat{\mathbf{x}}_i$ in case of a fault in the j -th sensor, with $i = j$.

3.5.4 Validation results

The performance of the fault management system is demonstrated during test drives, in which actual faults occur. In addition, several faults are injected in data from test drives that was replayed during off-line simulations. As an example, Figure 3.11 shows the results for a simulated fault in the longitudinal accelerometer.

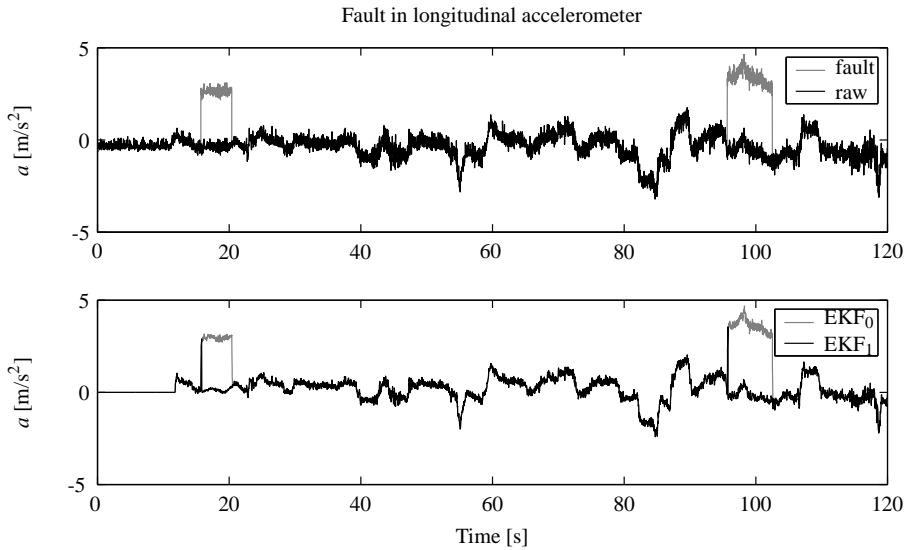


Figure 3.11: Results of host vehicle sensor FDI and fault-tolerant state estimation.

From the results it can be concluded that the fault management system performs adequate for the presented scenarios. However, before this system can be implemented in a production vehicle, more elaborate testing must assure that the sensitivity and robustness of the fault management system are above a desired level. Unfortunately, considering the wide variety of operating conditions and failure modes, it is difficult to obtain representative values for these performance measures and recreate the fault conditions. Nevertheless, since the focus of this thesis is precisely on this validation problem, the current system design is used for the remainder of this thesis.

3.6 Fault management for relative motion estimation

Apart from host vehicle information, an ADAS especially requires reliable information on other road users from environment sensors. However, due to noise, disturbances, and faults in individual sensors, fusion of data from multiple sensors is often necessary to obtain more complete and more accurate information of the traffic environment. In this section we therefore design a separate fault management system for environment sensing, in order to obtain accurate and reliable information on the relative motion between the host and target vehicles.

3.6.1 Definition of relative motion

Information on the relative motion between two vehicles is characterized by the distance x_r and y_r in longitudinal and lateral direction, relative velocities $v_{r,x}$ and $v_{r,y}$, and relative angle ψ_r , defined in a vehicle-fixed frame $\{L\}$, as illustrated in Figure 3.12. In the demonstrator vehicles two methods are available for obtaining this relative motion. The lidar directly measures the range r_{lidar} , angle ϕ_{lidar} , and range rate \dot{r}_{lidar} to an object in the polar frame of

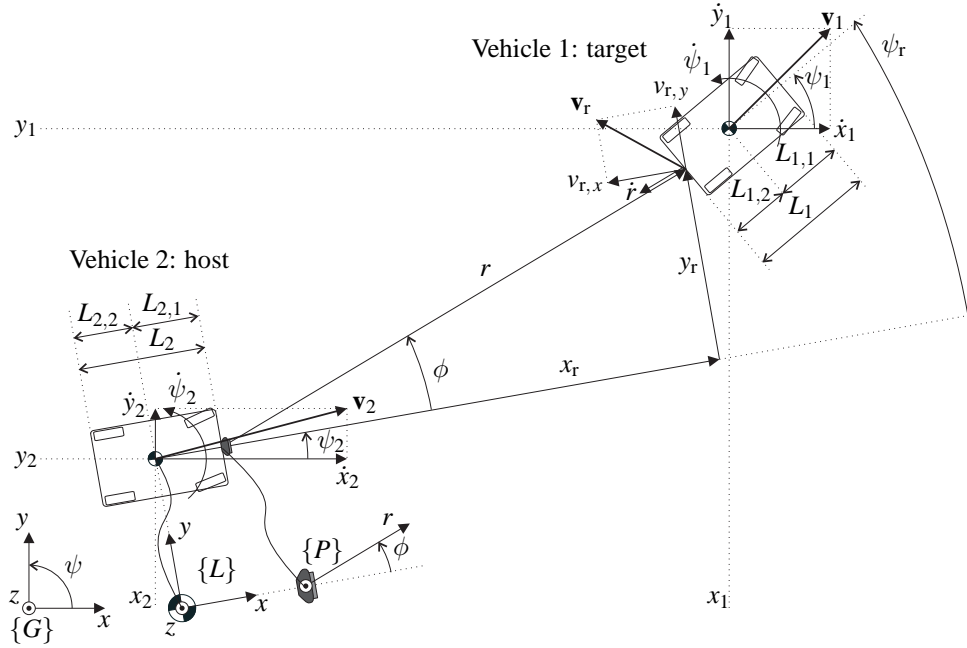


Figure 3.12: Model of the inter-vehicle motion between the host and target vehicle.

the sensor $\{P\}$. In addition, the relative state information can be derived from communication of the vehicle states in frame $\{G\}$, as estimated by the fault management system of the previous section. The first step in sensor fusion, is to represent the sensor measurements ${}^P\mathbf{z}_{\text{idar}}$ and vehicle states ${}^G\mathbf{x}$ in a common frame. The transformation of communicated data from $\{G\}$ to $\{P\}$ in order to obtain the relative motion ${}^P\mathbf{x}_{r, \text{comm}}$ is then given by:

$${}^P r_{\text{comm}} = \sqrt{({}^G x_1 - {}^G x_2)^2 + ({}^G y_1 - {}^G y_2)^2} - (L_{1,2} + L_{2,1}), \quad (3.52)$$

$${}^P \phi_{\text{comm}} = \arctan_2 \left(\frac{{}^G y_1 - {}^G y_2}{{}^G x_1 - {}^G x_2} \right) - {}^G \psi_2, \quad (3.53)$$

$${}^P \dot{r}_{\text{comm}} = ({}^G \dot{x}_1 - {}^G \dot{x}_2) \cos({}^P \phi_{\text{comm}} + {}^G \psi_2) + ({}^G \dot{y}_1 - {}^G \dot{y}_2) \sin({}^P \phi_{\text{comm}} + {}^G \psi_2), \quad (3.54)$$

where $L_{1,2}$ and $L_{2,1}$ are the vehicle lengths to take into account the translation from the origin of $\{L\}$ to $\{P\}$ (see Figure 3.12). The four-quadrant arc tangent operator \arctan_2 takes x and y as arguments and maps them into the full circle $(-\pi, \pi]$.

3.6.2 Data association for sensor fusion

Figure 3.13 gives a schematic diagram of the environment sensing FDI system, which involves sensor fusion. The first step in this sensor fusion for inter-vehicle state estimation, is to combine the data obtained from the lidar sensor and the data obtained through communication with other vehicles. Since the lidar system can detect up to five different objects, it is necessary to match a communicated target vehicle i to the associated detected object j . One possibility to accomplish this data association is to use the Mahalanobis distance \mathbf{s} , which is

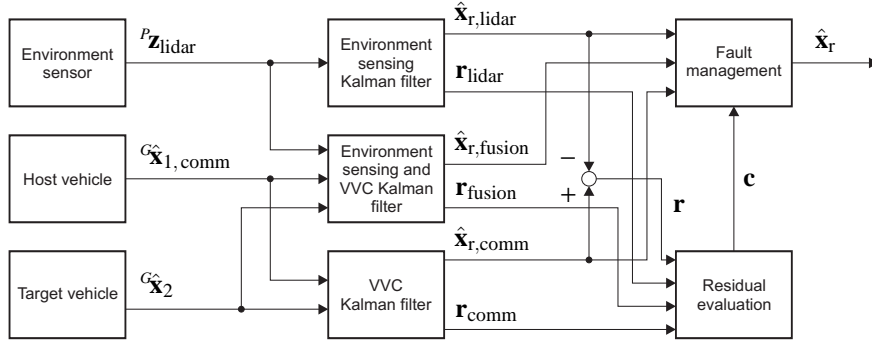


Figure 3.13: Fault management system for lidar and communication signals.

the Euclidean distance normalized by the variance of each variable. This distance provides an unbiased measure for the equivalence of two sets of measurements. Each element of \mathbf{s} is defined as

$$s_{i,j} = \sqrt{(\mathbf{z}_{\text{comm},i} - \mathbf{z}_{\text{lidar},j})^T \mathbf{V}^{-1} (\mathbf{z}_{\text{comm},i} - \mathbf{z}_{\text{lidar},j})}, \quad (3.55)$$

where $\mathbf{z}_{\text{comm},i} = [{}^p r_{\text{comm},i} \quad {}^p \phi_{\text{comm},i} \quad {}^p \dot{r}_{\text{comm},i}]^T$ is the state vector communicated by vehicle i , modified according to (3.52)-(3.54) and $\mathbf{z}_{\text{lidar},j} = [{}^p r_{\text{lidar},j} \quad {}^p \phi_{\text{lidar},j} \quad {}^p \dot{r}_{\text{lidar},j}]^T$ is the measurement vector of the j -th lidar target. To associate the correct object i with the correct target j , the Mahalanobis distance is calculated for each combination of object i and target j . Those objects and targets with the smallest distance $s_{i,j}$, taking into account a maximum threshold value, are associated. Figure 3.14 shows the target association according to (3.55) during a typical test run.

3.6.3 Sensor fusion by Kalman filtering

In case two corresponding measurements from lidar and communication are available, they are combined in a linear Kalman filter to provide an estimate for the inter-vehicle states.

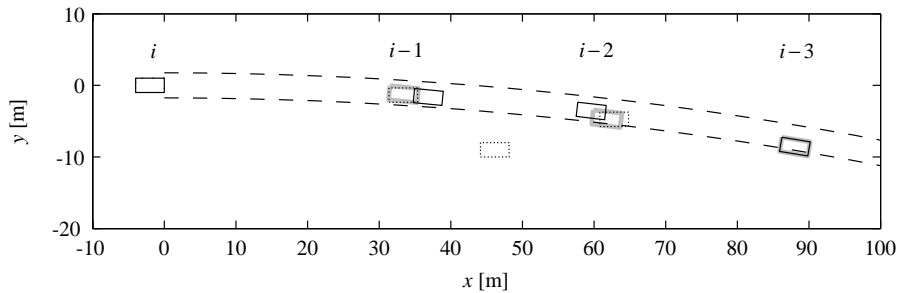


Figure 3.14: Visualization of target association, with objects received through communication (\square), targets obtained with the environment sensor (\square), and the resulting associated and relevant targets (\square). In addition, the path prediction for the host vehicle is indicated ($- -$), which illustrates that the vehicle is driving in a curve.

The relative state vector is defined as $\mathbf{x}_r = [x_r \ y_r \ v_{r,x} \ v_{r,y}]^T$, and the discrete-time state equations as:

$$\begin{bmatrix} x_r(k) \\ y_r(k) \\ v_{r,x}(k) \\ v_{r,y}(k) \end{bmatrix} = \begin{bmatrix} x_r(k-1) + t_s v_{r,x}(k-1) + w_{x_r}(k-1) \\ y_r(k-1) + t_s v_{r,y}(k-1) + w_{y_r}(k-1) \\ v_{r,x}(k-1) + w_{v_{r,x}}(k-1) \\ v_{r,y}(k-1) + w_{v_{r,y}}(k-1) \end{bmatrix}. \quad (3.56)$$

In case data from both sources is available, the discrete-time measurement equations are

$$\begin{bmatrix} x_{r,\text{lidar}}(k) \\ y_{r,\text{lidar}}(k) \\ v_{r,\text{lidar},x}(k) \\ v_{r,\text{lidar},y}(k) \\ x_{r,\text{comm}}(k) \\ y_{r,\text{comm}}(k) \\ v_{r,\text{comm},x}(k) \\ v_{r,\text{comm},y}(k) \end{bmatrix} = \begin{bmatrix} x_r(k) + v_{x_r,\text{lidar}}(k) \\ y_r(k) + v_{y_r,\text{lidar}}(k) \\ v_{r,x}(k) + v_{v_{r,x},\text{lidar},x}(k) \\ v_{r,y}(k) + v_{v_{r,y},\text{lidar},y}(k) \\ x_r(k) + v_{x_r,\text{comm}}(k) \\ y_r(k) + v_{y_r,\text{comm}}(k) \\ v_{r,x}(k) + v_{v_{r,\text{comm},x}}(k) \\ v_{r,y}(k) + v_{v_{r,\text{comm},y}}(k) \end{bmatrix}. \quad (3.57)$$

When data from only one source is available, four lines in the above equation are left out. The measurement covariance matrix \mathbf{V} in (3.57) is chosen in accordance with the noise variance of the lidar (see Table 3.2) and the error covariance \mathbf{P} corresponding to the state estimate $\hat{\mathbf{x}}$ (see Table 3.4). By tuning the process covariance \mathbf{W} in (3.56), a relative state estimation with sufficient performance is obtained. In addition, the filters generate residuals, which are useful for environment sensor fault detection. As illustrated in Figure 3.13, the fault management system assesses the residuals and switches to the appropriate state vector: $\hat{\mathbf{x}}_{r,\text{comm}}$, $\hat{\mathbf{x}}_{r,\text{lidar}}$, OR $\hat{\mathbf{x}}_{r,\text{fusion}}$.

3.6.4 Validation results

Figure 3.15 shows the distance and relative velocity during the same test run as Figure 3.14. The plots show the values obtained from lidar measurements, the relative motion computed using the model (3.52)-(3.54) and the resulting Kalman filtered states. These results show that the resulting estimated signal is more accurate and reliable than each of the two sensor signals separately. At $t = 1644.2$ s information from object $i-3$ (where the host vehicle is vehicle i) becomes available through VVC, with no matching lidar target, as can be seen from the first subplot. The Kalman filter then simply filters the state obtained through VVC. Vice versa, at $t = 1653.6$ s the lidar information from target $i-2$ disappears, because this target becomes occluded by target $i-1$. Consequently, the filter uses the information from VVC only. The second subplot shows the relative velocity information for the most important object $i-1$. This plot shows that the estimated $v_{r,x}$ responds faster to deceleration maneuvers of preceding vehicles, since the information from the VVC is available with a higher update rate than that of the lidar. This is an advantage in emergency braking maneuvers, since the host vehicle will be able to react earlier.

After sensor fusion, the targets are evaluated for their criticality. If their position is estimated as present in the host vehicle's path, they are identified as critical targets, and considered in the longitudinal control system. If a target is outside the host vehicle's predicted path, the target is discarded. Figure 3.14 includes the path prediction for the host vehicle.

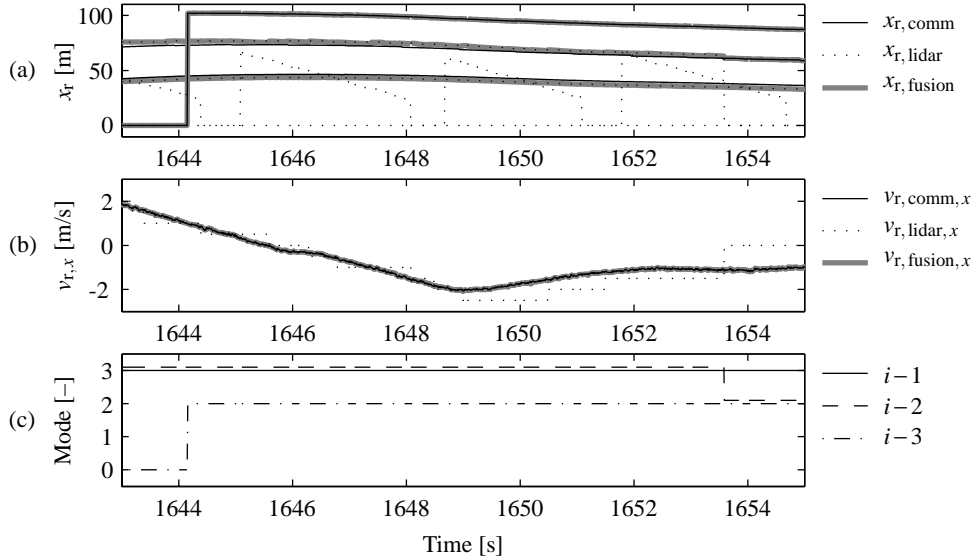


Figure 3.15: Sensor fusion of objects received through communication and targets acquired by lidar: (a) relative position x_r for all targets; (b) relative velocity $v_{r,x}$ for the most important object $i-1$; (c) Kalman filter mode for all objects (0 = no target, 1 = lidar target only, 2 = VVC object only, 3 = fused targets).

3.7 Summary

This chapter has presented an approach to fault-tolerant state estimation of both the host vehicle state and the inter-vehicle motion. A generalized observer scheme is designed for the vehicle state sensor FDI system. A separate fault management system handles the fusion of redundant target information that is obtained from an environment sensor (a lidar) and received through VVC. This fault-tolerant system for state estimation will be further used in the case study of Chapter 7.

Fortunately, faults are rare events, but this also makes it hard to validate the fault management system, which is a prerequisite for demonstrating the dependability of the ADAS. Furthermore, it is very time-consuming to identify all potential failure modes and it is difficult to reproduce the test conditions and failure modes under which the control system operates. The next chapter will therefore present and extend tools for ADAS control system validation, which are especially suitable for testing fault management systems in a reliable way.

Chapter 4

Development of ADASs with vehicle hardware-in-the-loop simulations

In order to achieve system dependability, three main concerns are important during the development of advanced driver assistance systems (ADASs). The most important starting point is the definition of objective dependability requirements, as presented in Chapter 2. Furthermore, integrated design, implementation, and verification of a fault management system is a prerequisite for a dependable ADAS. The previous chapter has therefore presented a design for a fault-tolerant state estimator that will be further demonstrated in Chapter 7, where it is implemented in a fault-tolerant longitudinal control system. Finally, validation of such an ADAS is an important issue, in order to gain confidence that the dependability requirements have been achieved.

The design and validation phases should therefore be supported by appropriate testing tools and methods, as defined in Objective 3 on page 8. The next chapter will present a new methodological framework for ADAS development, whereas this chapter presents a new development tool to support this framework. We start in Section 4.1 with a general overview of the ADAS development process, highlighting the challenges that are involved. The conventional tools that are used in this process are discussed in Section 4.2, where it will be shown that these tools have certain practical limitations. Especially the issue of fault management testing has limitations. Section 4.3 therefore introduces *vehicle hardware-in-the-loop* (VeHIL) simulations as a new tool that makes the development process of ADAS-equipped vehicles safer, cheaper, and more manageable. Based on the ‘V’ diagram, the position of VeHIL in the development process of ADASs is illustrated. The working principle and the added value of VeHIL are demonstrated in Section 4.4 with test results of a sensor fusion system, an adaptive cruise control (ACC) system, and a forward collision warning (FCW) system. Finally Section 4.5 summarizes the chapter, and identifies some remaining problems that will be addressed in the following chapters.

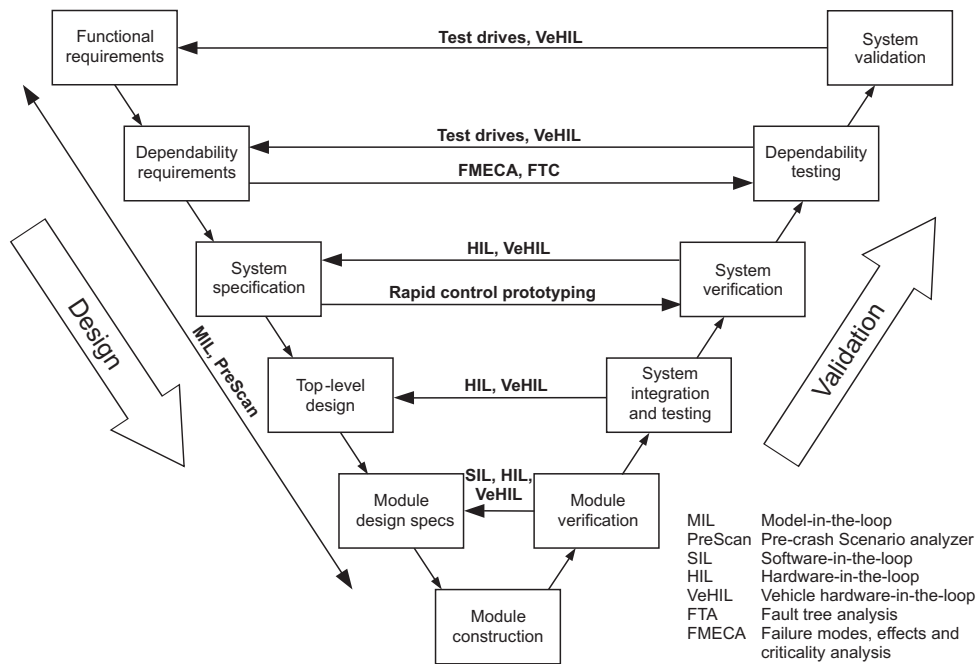


Figure 4.1: The ‘V’ diagram represents the sequential design and validation phases in the development of automotive safety-critical systems. The horizontal arrows indicate the mapping of design phases onto the corresponding validation phases (or vice versa), using the appropriate test tools.

4.1 Challenges in the ADAS development process

In the automotive industry the different phases in the development process of safety-critical mechatronic systems are often connected using the ‘V’ diagram [189]. As depicted in Figure 4.1, this diagram uses a ‘top-down’ approach for *design* and a ‘bottom-up’ approach for *validation*, although in practice the development process does not strictly follow all phases in this sequence and goes through several iteration loops. For relatively simple mechatronic systems, the design process is quite surveyable, as formalized in various generic methodologies for the design of mechatronic systems (*e.g.*, [112]). However, the various development phases for complex mechatronic systems, such as an ADAS-equipped vehicle, face some specific challenges that are addressed in this chapter.

4.1.1 Requirements and specification phase

There are guidelines and procedures available for ADAS development, such as the ADAS Code of Practice [206], ISO standards [106, 107], and the CONVERGE guidelines [120]. Unfortunately, these can only be applied at a high abstraction level in the development process, and do not provide objective requirements and evaluation criteria for ADAS validation and benchmarking, nor do they prescribe the use of specific tools and methods in the validation process.

In Chapter 2 we have therefore defined quantitative *functional requirements* in terms of the desired performance, driver comfort, and compatibility with the operating conditions. In addition, ADASs are safety-critical systems that require a high level of dependability. The system designer should therefore perform hazard and risk analyses to identify the *dependability requirements* in terms of the required level of reliability, safety, and fault tolerance. Important tools in this phase are FMECA and FTA to identify the risk areas that exist and use fault-tolerant control techniques to address them (see Figure 4.1). Blanke *et al.* [19] have used such an analysis of fault propagation in a systematic design strategy for fault-tolerant control systems.

From the functional and safety requirements a *system specification* is produced to define the precise operation of the system. However, in practice dependability requirements are often difficult to define and subject to ambiguity, which may lead to an incomplete or incorrect specification.

Subsequently, the system specification is used as the basis for the top-level design of the system architecture, followed by detailed module design (environment sensor, controller, actuator, human-machine interface). After implementation of the individual hardware and software modules, system integration takes place by assembling the complete system from its component modules.

4.1.2 Verification and validation

In every integration phase *verification* takes place to determine whether the output of a phase meets its specification, as illustrated by the horizontal arrows in Figure 4.1. On the component level this could mean, for example, testing the range, accuracy, and tracking capabilities of the environment sensor [1, 201]. On a higher level, verification must assure that integration with other subsystems does not have any negative side-effect.

Since verification only confirms compliance with the specification, errors in the specification may result in a faulty product. Furthermore, faults must be identified that have not yet been found during the design process. It is therefore important to perform *validation* of the integrated system against its requirements, especially for type approval and certification purposes. An example in a related domain is the European New Car Assessment Programme (Euro NCAP) to validate the crash safety against consumer requirements.

Unfortunately, it is difficult to validate fault-tolerant control systems against their dependability requirements. Firstly, the identification of *all* potential failure modes and their interactions is a very time-consuming process. Secondly, it is difficult to reproduce the test conditions and failure modes under which the control system operates. Furthermore, it is difficult to obtain a reliable estimate of a very small probability. Often only rough estimates of dependability can therefore be provided in an early stage without a fully implemented design or prototype.

Usually, the development process involves several iterations, where the results of verification and validation are used to modify the system specification and design, after which another test cycle takes place. Consequently, manufacturers are facing longer development times, whereas they have an increasing desire for a shorter time-to-market of their products. Likewise, the costs for the validation process increases. It is estimated that verification and validation of an automotive control system may take up to 50 % of the total development costs [100]. Obviously, there is a need to reduce the number of iterations and speed up this process. Because of the need for fast, flexible, and repeatable test results, various ‘in-

the-loop' simulation tools are increasingly being used for design and validation of ADAS controllers, as indicated in Figure 4.1. After a review of these tools, the position of the new VeHIL simulation tool in this development process will be clarified in Section 4.3.

4.2 State-of-the-art tool chain

4.2.1 Model-in-the-loop simulation

The initial design and specification of the ADAS controller is supported by off-line *model-in-the-loop* (MIL) simulations, where the controller logic is simulated in closed-loop with models of vehicle dynamics, sensors, actuators, and the traffic environment. The control system configuration and control law parameters are tuned to achieve the desired stability and performance requirements. Unfortunately, current simulation tools lack the possibility for testing a complete ADAS in a reliable way with full integration of operating conditions, sensor characteristics, vehicle dynamics, and traffic scenarios. The new simulation concept PreScan (Pre-crash Scenario analyzer) was therefore developed by TNO to allow reliable MIL simulation of ADASs in a microscopic traffic simulation, using sensor models for radar, lidar, and camera vision in a virtual environment [250]. The simulation engine of PreScan connects several modules, as illustrated in Figure 4.2:

- A pre-processor in the form of a graphical user interface that is used to define experiments. PreScan experiments typically contain a definition of the world (roads, buildings, etc.), the actors (vehicles and other road users), and the traffic scenario that these actors are involved in.
- A sensor world, in which several types of sensor models are available for lidar, radar, and camera vision, modeled after physical principles.
- A run-time environment that consists of a dedicated MATLAB/SIMULINK session, in which the vehicle dynamics are simulated. In addition, this module contains the ADAS control system, which includes sensor processing, decision logic, and vehicle control algorithms.
- A visualization server with 3D virtual road infrastructure, as illustrated in Figure 4.11 on page 93.

An addition to the pre-processor is the single-lane traffic model that was developed in Chapter 2. Furthermore, a dedicated MATLAB/SIMULINK toolbox CACClib has been developed within the framework of Chapter 3 that is available in the run-time environment. The use of these modules will be illustrated later on in the case studies of Chapters 6, 7, and 8. The compilation and simulation of traffic scenarios in PreScan is based on a multi-agent approach, as will be explained in Section 4.3.

PreScan is useful in several phases of the design process. Initially, it supports the definition of the overall system architecture, and the development of requirements and specifications. Different sensor types and sensor positioning can be assessed to aid the initial system design. Next, sensor post-processing algorithms, data fusion algorithms, controllers, and decision algorithms can be designed, evaluated, and fine-tuned, as illustrated by the work of Van der Mark [160].

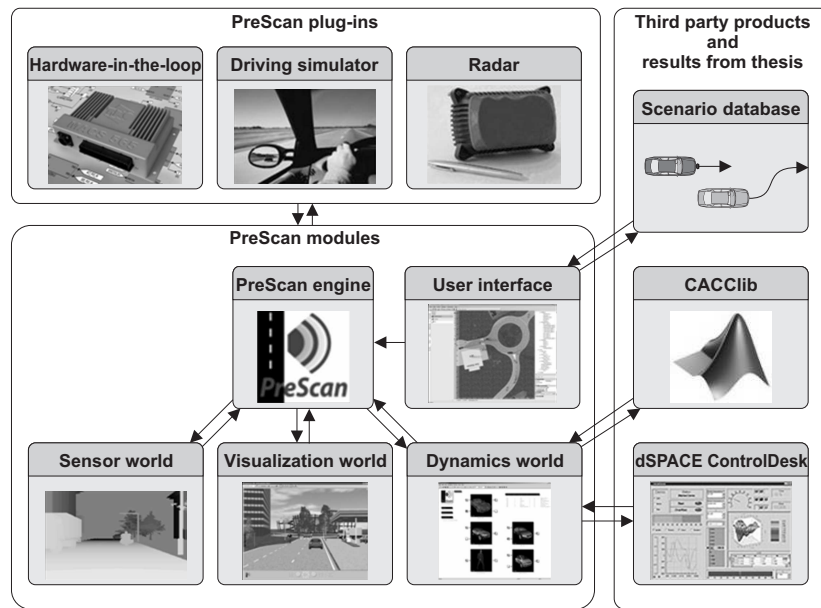


Figure 4.2: Schematic representation of the PreScan modules. This scheme also includes some plug-ins and additional modules that are developed within the framework of this thesis.

Depending on the objective of the simulation, the timing aspect is an important issue. The initial design phase requires faster than real-time simulation in order to speed up the design process, *e.g.*, using Monte Carlo methods. On the other hand, limitations in PC processing power may require complex sensor fusion algorithms to run slower than real-time. Finally, for testing the final product, a real-time simulation is always required, since the simulated components must run synchronized with the hardware or software modules, as discussed next.

4.2.2 Software-in-the-loop simulation

When MIL simulations have provided sufficient results, software code can be compiled from the simulation model of the control system using automatic code generation. The real code can then be verified with software-in-the-loop (SIL) simulations, where the remaining hardware components, vehicle dynamics, and environment are simulated in real-time. For more information on SIL simulation within the framework of the ‘V’-diagram, the reader is referred to [189].

4.2.3 Hardware-in-the-loop simulation

Similar to testing the real software in a SIL simulation, the real hardware can be tested in a real-time hardware-in-the-loop (HIL) simulation. HIL simulations consist of a combination of *simulated* and *real* components, see Figure 4.3. Alternatively, a real component can be *emulated*, *i.e.*, replaced by an *artificial* component that has the same input and output characteristics. Ideally, every component should be unable to distinguish between real,

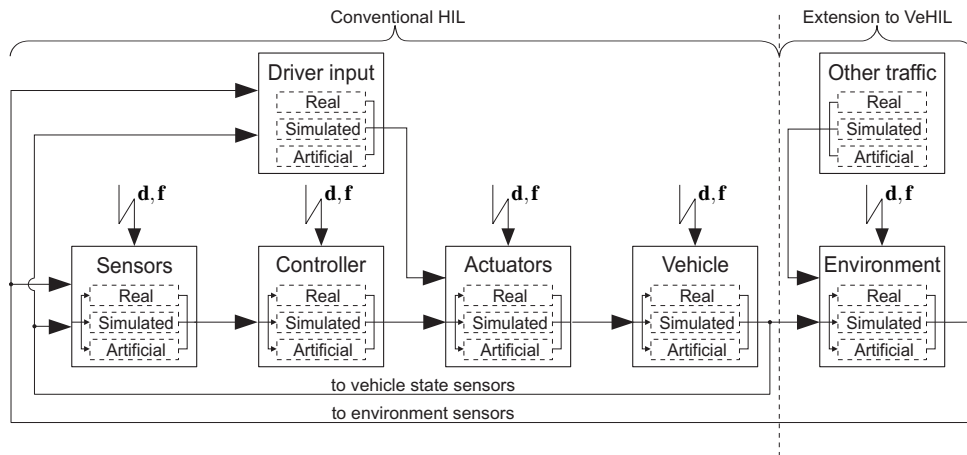


Figure 4.3: Possible configurations for HIL simulations, where parts of the system may be real, simulated or artificial. Feedback of signals from environment sensors and vehicle state sensors provides a closed-loop simulation. Additional disturbances \mathbf{d} and faults \mathbf{f} can be injected by the operator to test the system's dependability.

simulated or emulated components that it is connected to in the closed-loop configuration. Therefore, HIL offers the flexibility of a simulation, while the use of real hardware offers a high level of reliability.

The main advantage of a HIL simulation is that it provides a repeatable laboratory environment for safe, flexible, and reliable controller validation. Controller performance and stability can be systematically tested without disturbances from other unrelated systems, and dependability can be tested by controlled injection of disturbances and faults. HIL also allows validation of the real hardware in an early development phase without the need for a prototype vehicle, since any missing vehicle components can be simulated.

For these reasons, HIL simulations are more efficient and cheaper than test drives, and are extensively used for the development of vehicle control systems, such as ABS [149], engine control systems [114], and semi-active suspension systems [111]. ADASs can also be tested in several HIL configurations, as discussed next.

As indicated in Figure 4.1, in an early stage *rapid control prototyping* is carried out with emulated control functions. This involves implementing a model of the desired controller in a prototype vehicle for the purpose of rapid proof-of-concept, controller testing, and parameter adjustments. The software of the electronic control unit is then derived from the prototyped algorithm and tested with SIL simulations, as discussed above.

Next, the hardware controller can be tested in a HIL simulation for its real-time behavior [12]. This HIL testing can be done without the need for a prototype vehicle in order to assess the performance of the electronic control unit. This limited HIL setup can gradually be extended to include other modules, as the integration of the vehicle progresses. For instance, ADAS controllers can be tested in a HIL simulation with real actuators [12] and real sensors [273], where all other components are simulated. However, a complex interface between the *simulated* environment and the *real* sensor is required to generate a sensor signal. Instead of a real environment, an *artificial* environment can therefore be created to emulate the input to the process. For example, for a semi-active suspension system a hydraulic shaker

is often used to emulate the interaction with the road. Yet another type of HIL simulation is a driving simulator, which creates an artificial environment for an ‘in-the-loop’ human driver [198]. Driving simulators are useful for subjective evaluation of the ADAS and for fine-tuning ADAS controller settings.

Finally, the complete system can be real, including sensor, controller, actuator and vehicle dynamics. This complete vehicle system is in interaction with the road surface (through its actuators), as well as with the traffic environment that is formed by other objects in the world (through its sensors). Since environment sensors should receive a real input, an *artificial* traffic environment must be created to test an ADAS-equipped vehicle in a HIL simulation. Until recently, no such HIL environment has been available for testing complete intelligent vehicles.

4.2.4 Full-scale test drives

Test drives with prototype vehicles are always the final link in the validation chain to evaluate the system’s performance in the real-world environment that it will finally be used in. However, the value of test drives for control system design is limited, because they are hard to repeat and often inaccurate, due to the lack of ground truth knowledge on the exact state (*e.g.*, obstacle position) of the vehicles involved in the test. Much time and effort is therefore required to obtain useful results [201]. In addition, these tests are often expensive, unsafe, time consuming, and heavily dependent on weather conditions [12]. In the next section we therefore propose a solution to combine the advantages of HIL simulations with the representativeness of test drives, by extending the HIL environment from vehicle level to the traffic level, as indicated in Figure 4.3.

4.3 Vehicle hardware-in-the-loop simulations

To address the challenges mentioned in the previous sections, we present a new tool for the design and validation of intelligent vehicle systems: vehicle hardware-in-the-loop (VeHIL) simulation. VeHIL provides a solution for testing a full-scale intelligent vehicle in a HIL environment, where a chassis dynamometer is used to emulate the road interaction and robot vehicles are used to represent surrounding traffic. The VeHIL concept has been developed by TNO in conjunction with the work done in this thesis. It was first described in [253], patented in [137], and some preliminary test results have been presented in [74, 82, 85, 136, 251]. This section presents the VeHIL working principle in more detail and discusses the added value and position in the ADAS development process based on new test results.

4.3.1 Working principle of the VeHIL simulation

VeHIL constitutes a multi-agent simulator for intelligent vehicle systems, in which some of the simulated vehicles are replaced by real vehicles. These vehicles operate in an indoor laboratory that forms an artificial HIL environment for the intelligent vehicle. The environment sensors that are used in ADASs (radar, lidar, vision), collect relative position data in the absolute traffic environment. VeHIL therefore makes a transformation from the *absolute* motion of the objects in a traffic scenario to *relative* motion between those objects, as illustrated in Figures 4.4(a) and (b). Using only the relative motion between a fixed intelligent vehicle and target vehicles allows to have a controlled and space-efficient environment.

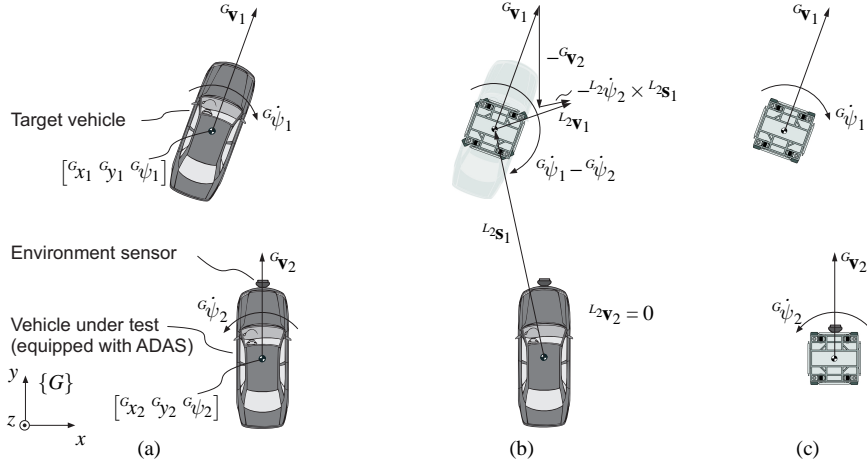


Figure 4.4: Transformation of coordinate frames: (a) absolute motion in the real world; (b) relative motion in VeHIL; and (c) absolute motion with two moving bases in VeHIL.

The software architecture of VeHIL is based on the multi-agent real-time simulator developed by Papp *et al.* [179], as illustrated in the lower-right part of Figure 4.5. This multi-agent framework consists of a collection of autonomous *entities* E (vehicles, other road users, or any other dynamical component), each controlled by its own internal dynamics (*e.g.*, a vehicle model, as discussed in Section 3.3.2). An entity has an absolute state \mathbf{x} in the global coordinate frame $\{G\}$, denoted as ${}^G\mathbf{x} = [\mathbf{s}^T \ \Phi^T \ \mathbf{v}^T \ \dot{\Phi}^T \ \mathbf{a}^T \ \ddot{\Phi}^T]^T$, where ${}^G\mathbf{s} = [x \ y \ z]^T$ represents the position and ${}^G\Phi = [\varphi \ \theta \ \psi]^T$ the orientation in Euler angles (roll, pitch, and yaw) of the entity. The corresponding velocity and acceleration components are represented by ${}^G\mathbf{v} = [\dot{x} \ \dot{y} \ \dot{z}]^T$, ${}^G\dot{\Phi} = [\dot{\varphi} \ \dot{\theta} \ \dot{\psi}]^T$, ${}^G\mathbf{a} = [\ddot{x} \ \ddot{y} \ \ddot{z}]^T$, and ${}^G\ddot{\Phi} = [\ddot{\varphi} \ \ddot{\theta} \ \ddot{\psi}]^T$.

Furthermore, a *virtual world* is defined that serves as a formal representation of the environment relevant to these entities. Entities are typically represented in the virtual world by *objects* O that interact with other objects (vehicles, bicyclists, pedestrians, infrastructure, traffic lights). Objects are *not* simulation models, but are merely the virtual representation of the simulation entities E . A visual representation of this virtual world with objects is shown in Figure 4.11, and is provided by the same visualization module of PreScan. After every integration time step of this multi-agent simulation, the internal dynamics of an entity (*e.g.*, E_2 , representing vehicle 2) result in a state \mathbf{x}_2 in the *global* coordinate frame $\{G\}$, notated as ${}^G\mathbf{x}_2$. Through the link between the simulation entity E and its virtual object O , the entity updates the representation ${}^G\mathbf{x}_2$ of the associated object O_2 in the virtual world. This link between entity and object is indicated by the dashed lines in Figure 4.5.

An important feature of the modeling concept of the multi-agent real-time simulator is that an entity (*e.g.*, a vehicle model) uses abstract sensors S and actuators A to interface with other objects in the virtual world. Through its abstract sensor S_2 the entity E_2 can collect information about the state ${}^G\mathbf{x}_1$ of another object O_1 (*i.e.*, vehicle 1, associated with E_1) in the virtual world. Vice versa, the entity E_2 has an abstract actuator A_2 to act on the state ${}^G\mathbf{x}_1$ of O_1 . Note that these sensors and actuators are handled in an *abstract* way: they have no dynamics and data processing features. Instead they can be interpreted as queries and

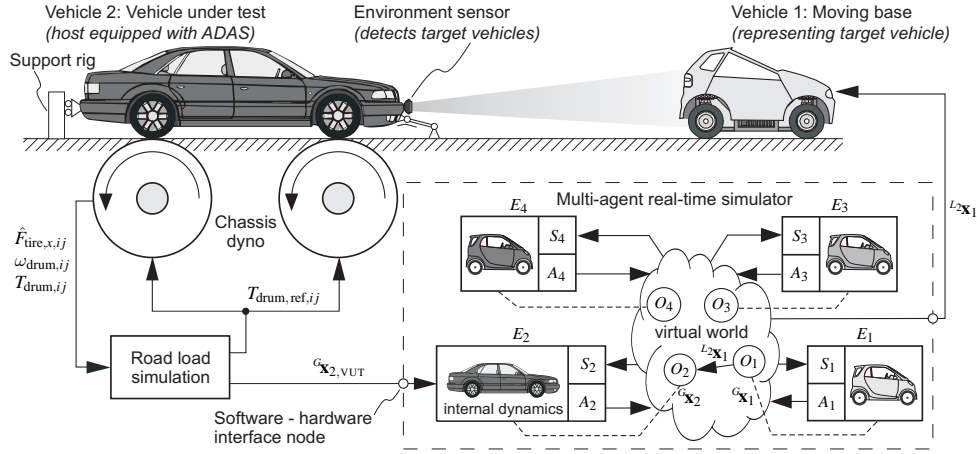


Figure 4.5: Schematic representation of the VeHIL closed-loop working principle. Every integration time step the simulation loop runs counterclockwise via the vehicle under test, the chassis dyno, the multi-agent real-time simulator, and the moving base, whose motion is detected by the sensor of the vehicle under test.

actions on the virtual world. *Real* sensors and actuators are modeled as part of the entity's internal dynamics [180].

Using this simulation principle, the relative motion between vehicles 1 and 2 (entities E_1 and E_2) from the viewpoint of vehicle 2 is obtained by a coordinate transformation, where the state ${}^G\mathbf{x}_1$ of vehicle 1 is represented in the *local* coordinate frame $\{L_2\}$ of vehicle 2, *i.e.*, ${}^{L_2}\mathbf{x}_1$. For the transformation to relative position and orientation, we then get

$${}^{L_2}\mathbf{s}_1 = {}^L_2\mathbf{R}({}^G\mathbf{s}_1 - {}^G\mathbf{s}_2), \quad (4.1)$$

$${}^{L_2}\mathbf{t}_1 = {}^L_2\mathbf{t}^G\mathbf{t}_1, \quad (4.2)$$

where ${}^L_2\mathbf{R}$ is the rotation matrix from frame $\{G\}$ to $\{L_2\}$ and \mathbf{t} represents the orientation in Euler parameters (also known as quaternions) [37]. If we neglect vertical vehicle dynamics (z , φ , θ) and only consider relative motion in the horizontal plane (x , y , ψ), the coordinate transformation in (4.1) and (4.2) simplifies to

$$\begin{bmatrix} {}^{L_2}x_1 \\ {}^{L_2}y_1 \end{bmatrix} = \begin{bmatrix} \cos {}^G\psi_2 & \sin {}^G\psi_2 \\ -\sin {}^G\psi_2 & \cos {}^G\psi_2 \end{bmatrix} \left(\begin{bmatrix} {}^Gx_1 \\ {}^Gy_1 \end{bmatrix} - \begin{bmatrix} {}^Gx_2 \\ {}^Gy_2 \end{bmatrix} \right), \quad (4.3)$$

$${}^{L_2}\psi_1 = {}^G\psi_1 - {}^G\psi_2. \quad (4.4)$$

Please refer to Figure 4.4 for a visual representation of this transformation. In a similar way, the transformations to relative velocity (${}^{L_2}\mathbf{v}_1$, ${}^{L_2}\dot{\psi}_1$) and relative acceleration (${}^{L_2}\mathbf{a}_1$, ${}^{L_2}\ddot{\psi}_1$) are derived [37]. For brevity, these derivations are omitted here.

The simulation is run by execution of entities on computing nodes, which are connected via a local area network. Each node has its own runtime environment, which also contains a representation of the virtual world. Entities communicate with this virtual world via their abstract sensors and actuators. The 'engine' of the entity simulation is an integrator (numerical solver), which invokes the entity's code (*i.e.*, the vehicle model) in timely

manner (synchronized with other entities in real-time). The implementation of the system architecture is Java-based with time-critical parts in C/C++. An interface is established to MATLAB/SIMULINK: C code compiled from SIMULINK models can be embedded into the runtime environment as entities. More details on this modeling concept and the runtime environment are described in [179].

The multi-agent simulator provides the framework, in which *any* type of vehicle model can be simulated. The model complexity depends on the type of ADAS and the objective of the simulation. Based on the scenario definition of Chapter 2, a scenario library is available that contains traffic scenarios, such as car-following, tailgating, cut-ins, and lane-changes. In addition, collisions, and near-miss scenarios are created from an in-depth accident analysis, see Chapter 8. The PreScan simulation tool, described in Section 4.2.1, is used for scenario definition and simulation before the actual VeHIL test takes place, based on the same multi-agent approach. Alternatively, predefined trajectories (*e.g.*, for benchmark and certification tests) or recorded test drives can be accurately reproduced in VeHIL.

4.3.2 Substitution of a vehicle dynamics model by a vehicle under test

With the ADAS-equipped vehicle and other road users modeled, the real-time simulation could run as a MIL simulation only, *i.e.*, a PreScan simulation without hardware. However, even a two-track model (3.27)-(3.47) is usually not sufficient to accurately model the ADAS-equipped vehicle. In order to test a *real* intelligent vehicle in a HIL configuration, the vehicle model of entity E_2 is substituted by the real vehicle under test (VUT), hence the term ‘*vehicle hardware in-the-loop*’. The ADAS-equipped VUT is therefore placed on a chassis dynamometer that provides a realistic load for the vehicle’s actuators (engine, brake system) and sensors (*e.g.*, wheel speed sensors).

The dynamic response of the chassis dyno, depicted in Figure 4.6, to driving actions of the VUT must be representative of real road conditions, especially in terms of delay time and phase lag. The operating frequency of the multi-agent real-time simulator is 100 Hz, which means that the delay time is at the most an acceptable 10 ms. The yaw response to steering input $\dot{\psi}/\delta$ and velocity response to throttle/brake input $\dot{x}/(T_{th} + T_{br})$ of a passenger vehicle typically show a bandwidth in the 1 Hz frequency range. This implies that the chassis dyno must at least have a bandwidth of about 5 Hz in order to minimize positioning phase lag. Furthermore, an emergency stop of a passenger vehicle can cause a maximum deceleration of around 10 m/s^2 . Consequently, the chassis dyno must be able to achieve this as well.

Table 4.1: Specifications of the chassis dyno.

Wheelbase	1.8 m to 4.0 m
Track width	1.2 m to 2.4 m
Drum configuration	4-wheel independent drive
Drum diameter	1592 mm
Total peak power	832 kW
Traction force	24 kN
Response time	< 10 ms
Maximum velocity	250 km/h
Dynamic range passenger cars (500 to 3500 kg)	full dynamics, -10 m/s^2 to $+10 \text{ m/s}^2$
Dynamic range commercial vehicles ($\leq 12\,000 \text{ kg}$)	reduced dynamics

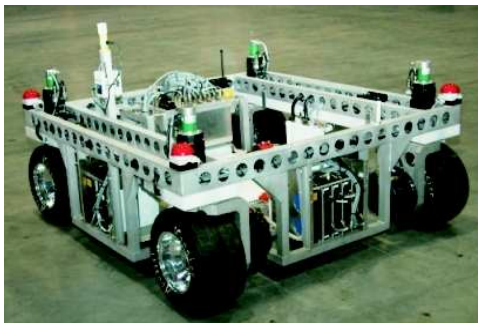


Figure 4.6: Vehicle under test with radar sensor at the front bumper, driving on the chassis dyno in VeHIL and supported by a rig at the front and back bumper.

These real-time requirements are met by a setup with four individual electric motor driven drums. The chassis dyno can fully simulate a vehicle mass between 500 and 3500 kg up to a maximum velocity of 250 km/h. The adjustable wheelbase accommodates a wide range of vehicle types: apart from passenger vehicles, also trucks, busses, and other automated guided vehicles can be tested. Table 4.1 summarizes the main specifications.

Note that the VUT itself replaces the vehicle model of (3.27)-(3.47). The chassis dyno only needs to emulate the tire forces $F_{\text{tire},x,ij}$ that the VUT would encounter on the road. Each force $F_{\text{tire},x,ij}$ is emulated by the drum moment of inertia I_{drum} and the electric motor torque $T_{\text{drum},ij}$ as

$$F_{\text{tire},x,ij} = \frac{C_0 + C_1 \omega_{\text{drum},ij} + C_2 \omega_{\text{drum},ij}^2 + I_{\text{drum}} \dot{\omega}_{\text{drum},ij} - T_{\text{drum},ij}}{R_{\text{drum}}}, \quad (4.5)$$



(a)



(b)

Figure 4.7: Moving base: (a) without body and (b) with body.

where the first three terms in the numerator represent friction losses in the chassis dyno, $\omega_{\text{drum},ij}$ is the measured drum speed, and R_{drum} the drum radius. From (4.5) the reference signals for the required motor torque are then calculated as

$$T_{\text{drum,ref},ij} = I_{\text{drum}}\dot{\omega}_{\text{drum},ij} + C_0 + C_1\omega_{\text{drum},ij} + C_2\omega_{\text{drum},ij}^2 - \hat{F}_{\text{tire},x,ij}R_{\text{drum}}, \quad (4.6)$$

where $\hat{F}_{\text{tire},x,ij}$ are observer estimated tire forces.

This setup also emulates the correct correlation between the individual drum speeds:

$$\omega_{\text{drum},ij} = \frac{v_{\text{wheel},x,ij}}{R_{\text{drum}}}, \quad (4.7)$$

to enable simulation of different wheel speeds when driving through curves, where $v_{\text{wheel},x,ij}$ are calculated from (3.32)-(3.34). In addition, a special restraint system that keeps the vehicle on top of the drums allows realistic heave and pitch motions of the vehicle body, as depicted in Figures 4.6 and 4.12. This rig produces a realistic dynamic vertical load transfer between rear and front axle during acceleration and deceleration in accordance with (3.44)-(3.47).

Finally, a road load simulation model estimates the VUT state vector ${}^G\mathbf{x}_{2,\text{VUT}}$ using the chassis dyno measurements and updates the state ${}^G\mathbf{x}_2$ of the associated object in the virtual world. No further interfacing between the real VUT and the simulation environment is necessary, such that the VUT can be tested as a black box system in a genuine HIL setup.

4.3.3 Substitution of a simulated target by a moving base

Similar to incorporation of the real VUT in a HIL simulation, surrounding road users can be represented by a so-called moving base, depicted in Figure 4.7(a) on the previous page. The moving base is a 4-wheel driven, 4-wheel steered robot vehicle that responds to position commands of the multi-agent real-time simulator and emulates the motion ${}^{L_2}\mathbf{x}_1$ of other road users *relative* to the VUT, such that this motion is detected by the VUT's environment sensor. For this purpose, the soft real-time simulator (Ethernet network) and the hard real-time chassis dyno and moving bases (CAN bus) are linked through dedicated interfaces, indicated in Figure 4.5. In order to carry out the desired relative maneuvers, the moving base must be able to perform motions that are not possible with a standard car (*e.g.*, sideways), as illustrated by the resulting velocity vector ${}^{L_2}\mathbf{v}_1$ in Figure 4.4(b). For this reason the individual wheels can be steered in a range of -350° to $+350^\circ$.

Like the chassis dyno, the moving base should also have a control bandwidth of about 5 Hz in order to minimize positioning phase lag. In addition, the moving base should be capable of accelerating with 10 m/s^2 in order to emulate the relative motion resulting from an emergency stop of the VUT. Finally, the top speed, which in view of the relative VeHIL world corresponds to the maximum relative velocity, should at least be equal to 50 km/h. This covers almost all highway scenarios, as was shown in Chapter 2.

These requirements are met by a vehicle platform equipped with independent all-wheel steering and all-wheel drive, using battery-powered DC servomotors. The trajectory controller of the moving base realizes the desired trajectory $\mathbf{x}_{\text{MB,ref}}(t)$, defined by the relative motion ${}^{L_2}\mathbf{x}_1(t)$ of the target vehicle in the horizontal plane. The only conditions are that the trajectory ${}^{L_2}\mathbf{x}_1(t)$ fits within the dimensions of the VeHIL laboratory (200 m by 40 m, see Figure 4.14) and meets the specifications of Table 4.2. The moving base controller determines the drive torques and steering torques so as to minimize the difference between the actual

Table 4.2: Specifications of a moving base.

Vehicle mass (including body)	650 kg
Wheelbase	1.4 m
Track width	1.4 m
Chassis configuration	4-wheel independent drive/steer from -350° to $+350^\circ$
Installed power	52 kW
Battery pack	288 NiMH D-cells, 375 V, 100 kg
Maximum velocity	50 km/h
Maximum longitudinal acceleration	10 m/s^2
Maximum longitudinal deceleration	-10 m/s^2
Acceleration from 0 to 50 km/h	2.1 s
Maximum centripetal acceleration	12 m/s^2

and desired moving base position, such that a repeatable trajectory is achieved within a position error ε of at the most 0.10 m, depending on the dynamics of the scenario. The moving base navigation system uses a combination of a magnet grid and odometry with a measurement accuracy of 0.04 m, resulting in a total positioning accuracy of (0.10 ± 0.04) m. For more information on the design and control of the moving base, the reader is referred to the work by Ploeg *et al.* [192, 193].

In order for the VUT to obtain realistic sensor data, the moving base is equipped with a vehicle body that represents similar target characteristics as a real vehicle, see Figure 4.7(b). Its radar cross section is similar to that of a standard passenger car, and the body has a similar shape and reflection properties for testing vision and lidar systems. A comparison of sensor systems using sensor calibration techniques will be demonstrated in Section 4.4.

Subsequently, the ADAS controller receives realistic input signals through its vehicle state sensors and environment sensors, and outputs command signals to the vehicle actuators (engine, brake) with a realistic actuator load, just as if the VUT was driving on the road. It must be emphasized that the actual moving base motion in VeHIL is *not* known *a priori*, but is the real-time equivalent of the resulting relative motion between an autonomously simulated target vehicle and an ADAS-controlled VUT. For example, when the VUT makes an emergency stop with deceleration $a_{2,\text{VUT}}$, the moving base accelerates forward with $a_{1,\text{MB}} = -a_{2,\text{VUT}}$. In this way a *closed-loop* HIL simulation is obtained, such that the ADAS is validated in an artificial traffic environment, including real vehicle dynamics and real sensor input.

4.3.4 Fault injection for validation of fault management systems

Apart from testing the nominal system behavior of an ADAS, it is important to validate its fault tolerance, which requires the use of *fault injection* techniques. Fault injection is the deliberate introduction of faults into a system, and the subsequent examination of the system for the errors and failures that result [10]. In the design phase fault injection can be applied for understanding the effects of faults, and feeding back test results to improve the controller and test procedures. In the validation process, the objective is to validate the fault management system and to forecast the faulty behavior of the target system.

Because of their flexibility in simulating operating conditions and manipulating controller inputs, HIL simulators are very suitable for fault injection. Fault injection for ADAS

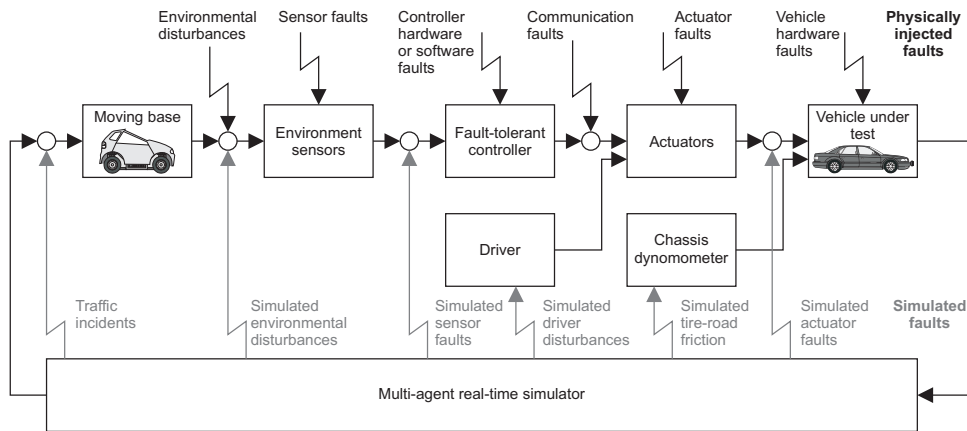


Figure 4.8: Possibilities for simulated and physical fault injection in VeHIL.

validation has usually been applied on the level of the embedded controller [21, 103]. New fault injection techniques are therefore required for faults that enter the target system at system level, such as sensor faults, actuator faults, and perturbations from the environment.

Fault injection in VeHIL can contribute to the dependability assessment of an ADAS in a number of ways. It can be used to assess the effectiveness of fault tolerance mechanisms built in the ADAS control system. Furthermore, fault injection may reveal potential failure modes that were not previously discovered. In VeHIL faults can be introduced in a controlled and repeatable way, which allows to determine the effect of a single fault or a combination of faults under specific conditions. Within the same test faults can be injected from the simulation environment and by physical injection, as illustrated by Figure 4.8. In this respect several performance indicators of fault injection have to be considered:

- *Controllability* of the experiment, which relates to the capability of the fault injection to control the appearance of errors in the system as they are produced by faults.
- *Observability* of the response of the fault-tolerant system in the presence of faults.
- *Fault representativeness* reflects to what extent the induced errors are similar to those provoked by real faults.
- *Fault equivalence* reflects to what extent distinct fault injection techniques lead to similar consequences (errors and failures).

4.3.5 Representativeness of VeHIL

A fundamental aspect of a HIL test environment is that it provides a *repeatable*¹ and *representative* testing environment. As we have demonstrated, the error variance and bandwidth

¹Note that the concept of *repeatability* — the variation arising when all efforts are made to keep conditions constant by using the same instrument and operator, and repeating during a short time period — should not be confused with *reproducibility* — the variation arising using the same measurement process among different instruments and operators, and over longer time periods.

of the moving base and chassis dyno are within the noise levels of environment sensor systems, such that VeHIL can be regarded as a repeatable testing environment.

Furthermore, the input from the artificial VeHIL environment into the VUT must be representative for the actual driving conditions on the road. A restriction of the VeHIL simulation in this respect is that vehicle-based inertial sensors (accelerometers and yaw rate sensors) do not give a representative signal, since the VUT is held at a stationary position. Another restriction is that the chassis dyno does not produce lateral tire forces $F_{\text{tire},y,lj}$ in accordance with (3.36)-(3.37) during steering actions of the VUT, since the slip angles of the front wheels α_{1j} equal the wheel angles δ_{1j} . However, the resulting relative lateral and yaw motion can still be correctly emulated, as shown in Figure 4.4(b). On the road, environment sensors can be perturbed by obstacles outside the relevant area (*e.g.*, infrastructure elements outside the path of motion). Much of the effort in sensor post-processing is associated with filtering out these disturbances. In VeHIL these disturbances can be different from the real world or even absent, although the absence of these disturbances does not affect the basic operation of the ADAS. A comparison between a test track scenario and a replay in VeHIL will be shown in Chapter 7 to confirm this.

To solve these issues, the HIL concept allows to feed the ADAS in real-time with a ‘mixture’ of real and virtual sensor signals. Any missing sensor signal can be generated from the real-time simulation of the vehicle model (3.27)-(3.47) in the multi-agent real-time simulator (the internal dynamics of entity E_2). This signal then replaces the real sensor signal and is subsequently fed into the ADAS controller, as schematically presented in Figure 4.8.

Alternatively, inertial and environment sensors can be installed on a moving base that executes a traffic scenario as if it were a standard road vehicle, while another moving base represents a target vehicle, as shown in Figure 4.4(c). This setup also allows to obtain a relative velocity of up to 100 km/h, when two moving bases drive towards each other.

Due to the absence of a realistic driving environment, VeHIL is not intended to serve as a driving simulator, although it has potential to include driver interaction, as will be illustrated in Section 4.4.3. VeHIL is therefore not meant to replace test drives, but focusses on repeatable and accurate testing of the ADAS performance and dependability before ‘human-in-the-loop’ test drives take place. In addition, VeHIL tests are used for those scenarios that are too difficult or dangerous to perform on the road.

4.3.6 Added value of VeHIL in the development process of ADASs

By providing a world-wide unique HIL environment for intelligent vehicle systems, the VeHIL laboratory offers a number of distinct advantages:

- Tests are performed in a repeatable and flexible way with high accuracy, since the moving bases are operated from a computer-controlled environment. This allows precise variation of test parameters to assess the influence of specific parameters and failure modes on the ADAS performance.
- Tests are safer, due to the absence of high absolute velocities. Furthermore, traffic scenarios are monitored by a supervisory safety system, which prevents any real collisions. This allows to test ADASs in safety-critical scenarios.
- The costs of the validation process are reduced, because many tests are performed in a short time frame. The VUT can drive for hours and be continuously tested, which

is not possible during test drives. Depending on the complexity of the scenarios, up to 20 tests per hour can be performed, including scenario compilation, trial runs, test execution, and data acquisition. A test cycle is therefore significantly faster than is possible with test drives [133]. This will also be demonstrated in the case studies.

- Due to the high accuracy of tests, a high success rate for testing is obtained.

Because of these advantages, VeHIL complements the existing development process of ADASs in many phases and on many of the levels of the ‘V’-diagram of Figure 4.1:

- Rapid control prototyping in VeHIL can help to define the system specifications in an early development stage. In addition, based on safety-critical maneuvers and fault injection, potential hazards can be analyzed.
- The flexible transition from MIL simulation in PreScan to HIL simulation in VeHIL allows a model-based development of the controller. Critical scenarios that are identified with MIL simulations can be quickly uploaded in VeHIL for experimental testing. Test results can then be compared with the simulation results for validation of sensor and vehicle models.
- On the module level the ability to combine high position accuracy with high and accurate relative speeds makes VeHIL an efficient tool in verification and benchmarking of the exact performance of environment sensors (*e.g.*, sensor calibration).
- On the system level VeHIL especially facilitates the *functional* validation of the performance and dependability of complex black-box controllers against objective measures. Algorithm evaluation, fine-tuning, and benchmarking can be done efficiently.
- For production sign-off and certification purposes the high repeatability and ability to deal with safety-critical applications make VeHIL a strong tool.
- Finally, VeHIL facilitates the transition from simulations to outdoor test drives that are used to evaluate the real performance and dependability on the road. These test drives can be performed with a much higher confidence and less risk, when the ADAS has already been thoroughly tested in VeHIL.

We will demonstrate the suitability and added value of VeHIL in the next section with several examples.

4.4 VeHIL test results for ADAS applications

In cooperation with industrial parties, tests have been conducted for several vehicle types (trucks, cars), ADAS applications (ACC, stop-and-go, FCW, pre-crash systems, blind spot systems), and sensors (radar (pulse-Doppler, FSK, FMCW), vision (both mono and stereo), and lidar). Here we will discuss the test results for sensor calibration, ACC, and FCW. For clarity, the presented controllers are simpler than the actual implementation, since the focus is not on their actual performance, but on the way they are tested.

4.4.1 Sensor calibration

Calibration of environment sensors is necessary for verification of the sensor specifications, and for modeling and benchmarking purposes. VeHIL provides an accurate ground truth on the obstacle position, since the moving base state vector $\mathbf{x}_{\text{MB}}(t)$ is accurately known in real-time. The ability to combine high position accuracy with high and accurate relative speeds is an efficient tool in identifying and benchmarking the exact performance of a sensor over its entire operating range. Since sensor experiments are open-loop (there is no vehicle system active), a high number of tests can be performed.

To illustrate the added value, sensor calibration tests were done on the TNO RoboJeep, depicted in Figure 4.9. The RoboJeep is an autonomous vehicle, equipped with a 2.4 GHz FMCW (single beam) radar and a stereovision system, consisting of two CMOS-cameras, with 1024×1024 pixels and a field-of-view of 50° . A dense disparity algorithm was used to detect obstacles and compute the distance to the RoboJeep [160].

Figure 4.10 shows the test result of an moving base moving in longitudinal direction in front of a radar and vision sensor. In-depth analysis of the corresponding data shows that in low-speed maneuvers the moving base has a real-time position accuracy of (0.03 ± 0.04) m, which is within the resolution of state-of-the-art environment sensors. Typical sensor characteristics, such as the detection range, field of view, accuracy, and resolution can therefore be accurately verified. In addition, information on the detection delays can be retrieved. Furthermore, the performance of the sensor fusion system, which tries to correct disturbances and faults by fusing the data with input from other sensors, can be validated for the detection rate and classification rate.

4.4.2 Adaptive cruise control system

An ACC controller must be tested in a closed-loop experiment, since the ACC control actions affect the relative motion, which in turn is detected by the environment sensor. Apart from the vehicle itself, optionally a human driver can be included ‘in-the-loop’ to operate the ACC control lever and to introduce disturbances. The prototype vehicle, depicted in Figure 4.6 has been implemented with the ACC control law (2.12).

This control law is tested for the traffic scenario of Figure 4.11, which is transferred to VeHIL, as shown in Figure 4.12: the ACC-equipped vehicle 2 drives on the middle lane when suddenly target vehicle 1 cuts in from the right lane at close distance ($e_x > 0$) and driving at a lower speed ($v_r < 0$). This happens at $t = 22.9$ s, which can be seen from the range x_r and angle ϕ to the target in Figure 4.13. At $t = 25.3$ s the radar sensor on vehicle 2 detects the target in the host vehicle’s lane, *i.e.*, the angle ϕ to the target is approximately 0. Control law (2.12) then results in a reference acceleration $a_{\text{ref}} < 0$ and the ACC activates the brake system. Vehicle 3 stays on the right lane and is used to test the ability of the ACC system to distinguish between important and irrelevant targets in the traffic environment (*i.e.*, vehicle 3 should not be considered a target). On a test track it would be very difficult to carry out such a test with human drivers in a safe and repeatable manner, but in VeHIL the scenario can be accurately reproduced. Especially note the transformation from absolute to relative motion, *i.e.*, $\mathbf{v}_{\text{MB}} = \mathbf{v}_1 - \mathbf{v}_2$. The results also show that the moving base has a maximum position error ε of 0.10 m between desired and measured position, and a repeatability within 0.01 m between consecutive test runs. The velocity error is usually smaller than 0.1 m/s.

For this type of tests VeHIL has an added value in identifying the requirements and capabilities of an ACC system for safety-critical traffic scenarios in an early development



Figure 4.9: Experimental setup in VeHIL for sensor calibration with the moving base on the left, and the RoboJeep, equipped with lidar, radar and stereovision, on the right.

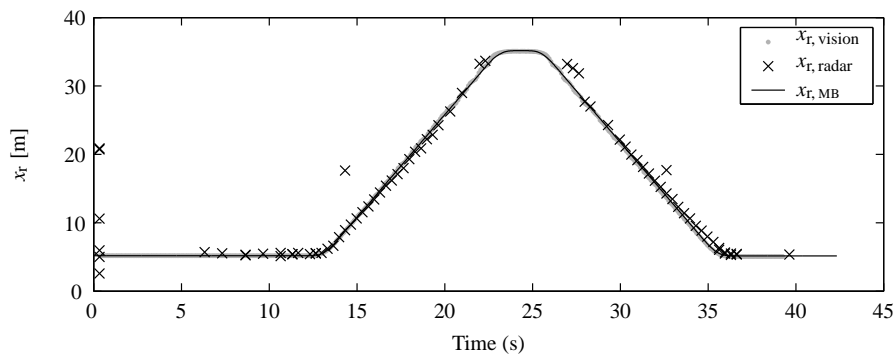


Figure 4.10: Comparison of the stereovision and radar sensor data with the ground truth from the moving base.

stage. Using rapid control prototyping techniques, various control settings are efficiently tested for a variety of scenarios. When the effect of various traffic disturbances on control performance is known, controller parameters can be optimally tuned. In a later stage functional validation of the completed system to these requirements can be done unambiguously and efficiently.

4.4.3 Forward collision warning system

Testing an FCW system is more safety-critical than ACC, since a collision warning system is activated shortly before a collision is expected. A warning is issued when a threshold of maximum braking capability a_{\min} is crossed by the deceleration a_{ref} required to prevent a collision.

The truck, shown in Figure 4.16 on page 96, is equipped with a control law similar to (2.4). In the simulated scenario, an inattentive truck driver, driving at 25 m/s, slowly approaches another vehicle, driving at 23 m/s (represented by the moving base). After the



Figure 4.11: Visual representation of a cut-in scenario in the virtual world: an ACC-equipped vehicle drives on the middle lane, when suddenly one of the two preceding vehicles cuts in from the adjacent lane with a lower velocity.

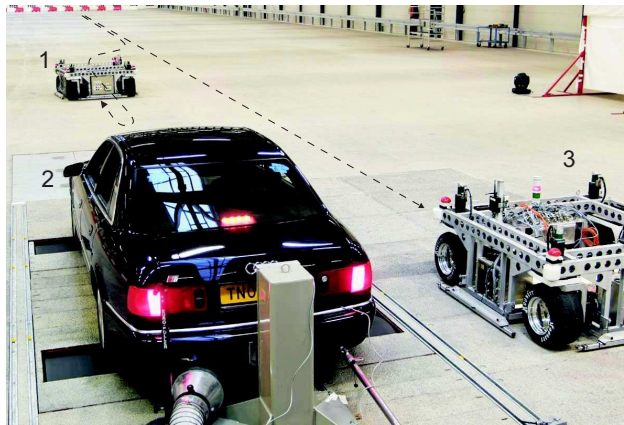


Figure 4.12: The cut-in scenario of Figure 4.11 is reproduced in VeHIL to test an ACC system: (1) moving base no. 1 (MB1), performing the cut-in maneuver of vehicle 1 from the viewpoint of vehicle 2; (2) vehicle under test; (3) MB2, representing the relative motion of the other target vehicle 3.

preceding vehicle suddenly brakes at $t = 46.7$ s, a_{ref} in (2.4) drops below $a_{2,\text{min}}$ at $t = 49.2$ s, and subsequently the FCW system sends a collision warning to the driver. The corresponding test results in Figure 4.15 show that, after a slight delay due to driver reaction time, the driver brakes at $t = 49.9$ s and avoids the collision.

In this way, optimum warning thresholds are defined by executing repeatable and safe experiments. Apart from objective parameter tuning, VeHIL also has potential for subjective evaluation in addition to on-road tests. It can be verified whether the warnings, when given in defined critical situations, are timely and appropriate. Although the final subjective evaluation should be done on the road, VeHIL can be used for an initial evaluation using the measures of Section 2.3.4. Ongoing research focusses on linking VeHIL to a driving simulator [13], which would enable to run high-fidelity dynamical behavioral studies.

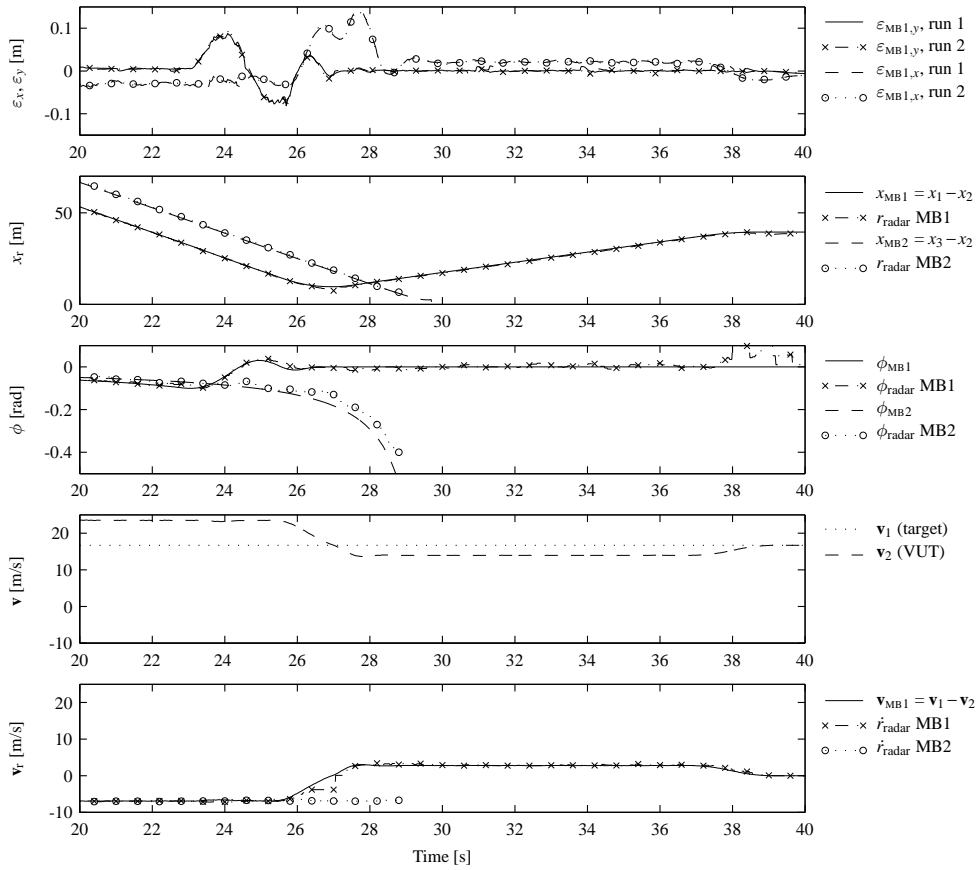


Figure 4.13: VeHIL test results for the cut-in scenario with the ACC system.

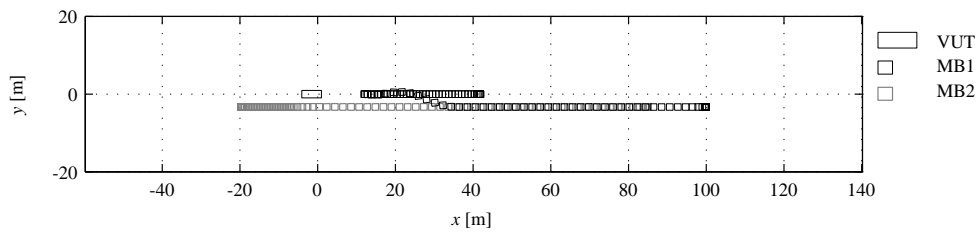


Figure 4.14: Trajectories of the moving bases in the VeHIL laboratory during the ACC test. Moving base 1 starts at $(x, y) = (100, -4)$ and ends at $(x, y) = (42, 0)$, whereas moving base 2 starts at $(x, y) = (85, -4)$ and ends at $(x, y) = (-20, -4)$.

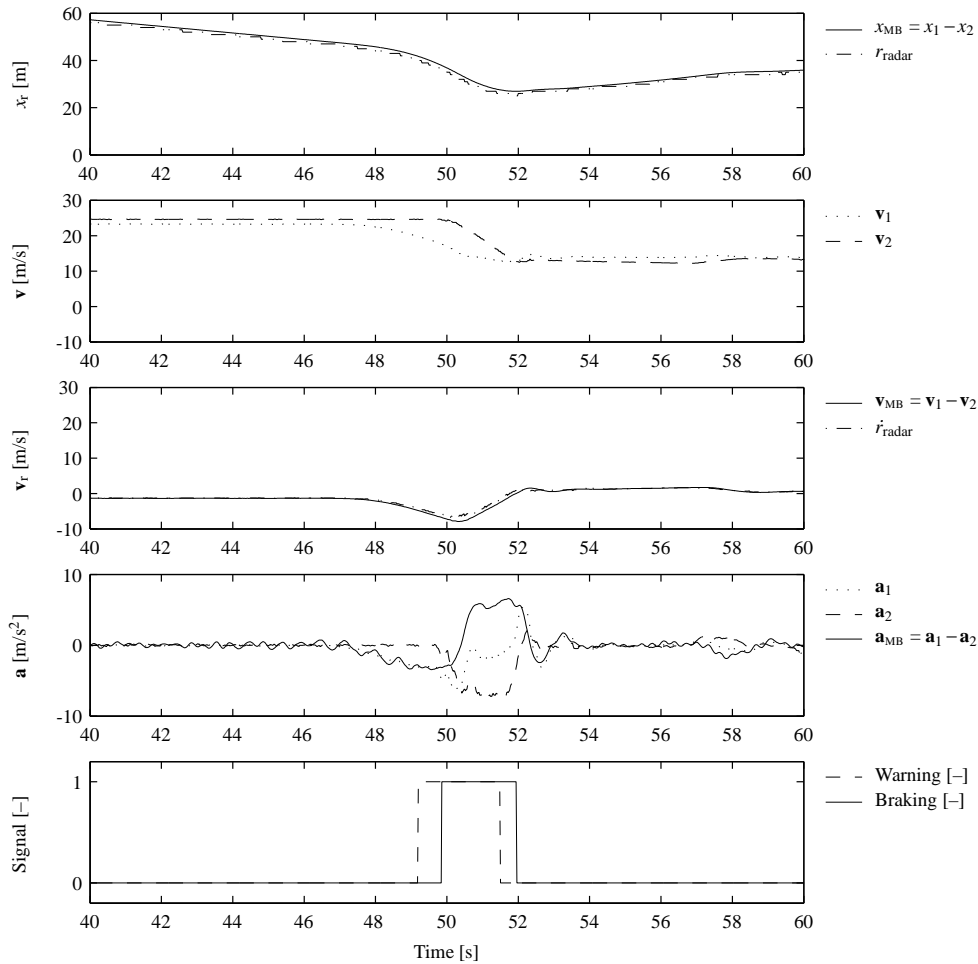


Figure 4.15: VeHIL test results for the FCW system in an approach scenario with emergency brake maneuver.

4.5 Summary

This chapter has introduced the concept of HIL simulation of hardware components, which allows quick and accurate evaluation of the system before test drives take place. In that way, HIL can resolve most of the difficulties associated with full-scale prototype tests, while still having a relatively high level of reliability due to the connection with real hardware. We have then presented the new VeHIL concept for testing ADASs, where a real intelligent vehicle is operated in a HIL environment. VeHIL is suitable for various types of ADASs: ACC, stop-and-go, FCW, pre-crash systems, blind spot systems, and fully autonomous vehicles. VeHIL experiments can be performed in an accurate, repeatable, and controllable way to create a representative test environment. With test results it has been demonstrated that VeHIL has an added value in several phases of the development process of an ADAS: sen-



Figure 4.16: Experimental setup of a VeHIL test with a truck, equipped with an FCW system.

sor verification, rapid control prototyping, functional performance validation, fine-tuning of control algorithms, and production sign-off tests, as will be further demonstrated in the case studies of Chapters 6, 7, and 8.

Furthermore, tests can be performed more efficiently than with outdoor test drives, and test scenarios can be varied very easily, due to the connection to the underlying simulation environment PreScan. Subsequent test drives can then be performed with a much higher confidence in the system, since the ADAS has already been thoroughly tested in VeHIL. VeHIL is therefore *not* meant to replace MIL simulations and test drives, but to form an efficient link between them. Consequently, the number of iteration loops in the development process can be reduced, saving time and costs. Unfortunately, in practice there is a huge number of combinations of scenario parameters possible, so the ADAS cannot be tested exhaustively. Generation of a representative set of test vectors is crucial to an efficient and effective validation program using PreScan and VeHIL. In the next chapter we will therefore present a probabilistic approach for testing the ADAS for a representative set of scenarios, by identifying the number and type of tests to perform.

Chapter 5

A methodological framework for probabilistic validation of ADASs

The previous chapter has presented the challenges regarding development of control systems for advanced driver assistance systems (ADASs), and introduced tools to speed up the validation process. The challenges involved in validation, as stated in Objective 4 on page 9, will be further illustrated in Section 5.1, where several state-of-the-art methods for control system validation are discussed. To address these challenges, Section 5.2 introduces the concept of randomized algorithms for probabilistic controller validation. The theory is illustrated with a simple case study that highlights the large number of samples that are required for validation. Section 5.3 then discusses various possibilities for improving the efficiency of randomized algorithms, one of which is the use of importance sampling. This method is further extended to a new algorithm for adaptive importance sampling in Section 5.4, and the case study is extended for the multi-dimensional case. Section 5.5 presents a methodological framework to integrate this randomized algorithm with the use of the simulation tool PreScan, the VeHIL laboratory, and test drives. Finally, Section 5.6 summarizes the results of this chapter.

5.1 Objectives and methods for control system validation

As was discussed in the previous chapter, the automotive industry currently lacks a standardized industry-wide validation method for ADAS control systems. Compared to the aerospace industry, where *clearance* of flight control laws has been subject to standardized methods for decades [123], the automotive industry has still to attain the same level of validation. Long before an aircraft can be tested in flight, the dependability and performance of a flight control system must be proven to the authorities under all possible combinations of operating conditions, parameter variations, and failure modes. Although stability and handling requirements are even more stringent for an aircraft than for an automobile, the aircraft's environment is quite predictable. Furthermore, the trained pilot and the flight control system usually have sufficient time available to recover from a malfunction or to anticipate on disturbances. However, the type and size of uncertainty for an automobile in a complex traffic environment is considerably larger. The huge amount of validation

work therefore requires fast, efficient, and numerically reliable methods. New techniques are therefore needed for the faster detection of combinations of parameter values and traffic maneuvers, for which the requirements are not met. Such worst cases may be caused by rare combinations of events, which makes it particularly difficult to detect them. In the aerospace industry several methods have therefore been developed for the analysis of complex uncertain systems, as described by Fielding *et al.* [61]. As we will see in this chapter, each of these techniques has its known strengths and weaknesses. To illustrate this, we will first describe a simple linear control problem for adaptive cruise control (ACC).

5.1.1 A case study: The ACC control problem

The ACC longitudinal control problem consists of two vehicles, as was illustrated in Figure 2.3. Since the objective is to control the motion of the host vehicle 2 *relative* to the preceding target vehicle 1, the vehicle state is chosen as $\mathbf{x}_2 = [x_{r,2} \quad v_{r,2} \quad v_2]^T$. Every time that the sensor detects a new target vehicle that the ACC system should follow, the time counter t is reset to zero, and the initial condition of \mathbf{x}_2 is defined as $\mathbf{x}_2(0) = [x_{r,2}(0) \quad v_{r,2}(0) \quad v_2(0)]^T$. The state space representation (2.15) can then be written as

$$\dot{\mathbf{x}}_2 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{x}_2 + \begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix} a_2 + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} a_1, \quad (5.1)$$

where the acceleration of the host vehicle a_2 is the control input, and the acceleration of the target vehicle a_1 forms the disturbance to the system. Since this is a linear feedback control system, perturbed by bounded disturbances and uncertain parameters, the system can be rearranged into the general control configuration, known as the $\mathbf{M}\mathbf{\Delta}$ -structure. In this configuration \mathbf{M} represents the known part of the system and $\mathbf{\Delta}$ represents the uncertainty present in the system. The system can then be investigated for robust stability and robust performance using standard μ -analysis techniques, as discussed in several textbooks [220].

5.1.2 Characterization of performance measures by cost functions

Before any validation can take place, we should define stability and performance more clearly. As discussed in Chapter 2, ACC performance can be quantified using several measures ρ_k . These can be Boolean, such as the safety measure $\rho_{\text{coll}} \in \{0, 1\}$, where $\rho_{\text{coll}} = 0$ means that the ACC manages to safely follow the target vehicle, and $\rho_{\text{coll}} = 1$ means that the scenario would require a brake intervention by the driver to prevent a collision. However, such a Boolean measure cannot distinguish in severity between different situations for which $\rho_{\text{coll}} = 1$. Therefore, a continuous safety measure can also be used, such as the time-to-collision (TTC), tracking error, control effort, ride comfort, and string stability.

The value of these performance measures for a particular scenario depends on the perturbations imposed by that scenario. As we have seen in Chapter 2, this relates to the motion of other vehicles, as well as the initial conditions. As indicated in Figure 2.5, these scenario parameters, together with driver input, disturbances, and failure modes form the *parameter set* \mathcal{Q} : an n -dimensional set of all possible parameter combinations. Control system validation then requires to evaluate the function $\rho_k : \mathcal{Q} \rightarrow \mathbb{R} : \mathbf{q} \mapsto \rho_k(\mathbf{q})$, where \mathbf{q} is an n -dimensional vector, with q_i the i -th element of \mathbf{q} . Some of these individual cost functions

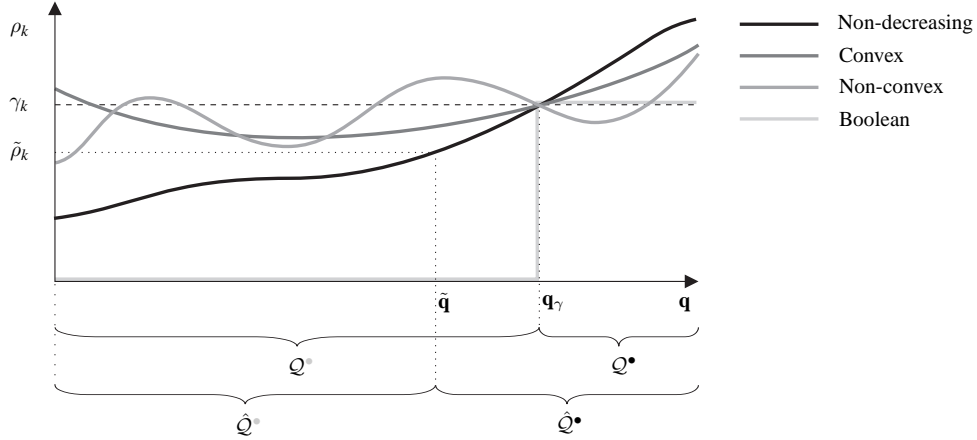


Figure 5.1: Examples of cost functions: non-decreasing, convex, non-convex, and Boolean. A verification test at $\mathbf{q}_j = \tilde{\mathbf{q}}$ may give estimates for the cleared parameter set $\hat{\mathcal{Q}}^*$, as well as the non-cleared set $\hat{\mathcal{Q}}^\bullet$. Note that the actual boundary between \mathcal{Q}^* and \mathcal{Q}^\bullet , which lies at \mathbf{q}_γ , is usually difficult to estimate.

can be unimodal, *i.e.*, monotonously increasing or decreasing, but others will have more than one local minimum, as illustrated in Figure 5.1 for the one-dimensional case.

5.1.3 Verification of the system specifications

Experiments, *i.e.*, individual simulation runs of a traffic scenario with a particular realization of $\mathbf{q}_j \in \mathcal{Q}$, can be performed to evaluate $\rho_k(\mathbf{q}_j)$ for different objectives. One of these objectives is to verify that the system specifications are met, such as a sign-off test before the start of production. The (sub)system is then required to pass a pre-defined test that is expected to represent the most stringent condition the system will encounter in practice. This can be formalized in the following problem statement.

Problem 5.1 Verification of system specifications

Given a specific condition $\mathbf{q}_j \in \mathcal{Q}$, verify whether the performance measure ρ_k meets the specification γ_k . \square

This problem can be easily solved for a particular ρ_k under the influence of a particular \mathbf{q}_j , by just checking the parameter combination of interest $\tilde{\mathbf{q}}_j$ and evaluate whether $\rho_k(\tilde{\mathbf{q}}_j) < \gamma_k$. In case the cost function ρ_k is non-decreasing, as illustrated in Figure 5.1, every \mathbf{q}_j within the set $\{\mathbf{q} \in \mathcal{Q} | q_i \leq \tilde{q}_i\}$ is then also cleared. In case the verification problem involves checking $\rho_k(\tilde{\mathbf{q}}_j) > \gamma_k$, this statement can be easily reversed to apply to the cleared set $\{\mathbf{q} \in \mathcal{Q} | q_i \geq \tilde{q}_i\}$.

Example 5.1 Verification of the system specifications. Consider a car-following scenario for the system (5.1), where both the host and target vehicle are initially driving with constant and equal velocity, and the target vehicle suddenly brakes. It is obvious that the likelihood of a collision event, denoted with ρ_{coll} , increases with a stronger deceleration of the target vehicle. In this way the cost function $\rho_{\text{coll}}(a_1)$ is non-increasing.

If we verify that for a particular value of the target vehicle acceleration \tilde{a}_1 the ACC system is safe, it can be concluded that it is also safe for all other $a_1 > \tilde{a}_1$.

The benefit of verification testing is that a single test clears the entire parameter set \mathcal{Q} . However, verification gives no information on the volume of the cleared parameter set \mathcal{Q}^* , since it may extend to other cleared parameters \mathbf{q}_j in the set $\{\mathbf{q}_j | (\mathbf{q}_j)_i > (\tilde{\mathbf{q}}_j)_i\}$ for which $\rho_k(\mathbf{q}_j) < \gamma_k$, as illustrated in Figure 5.1. It is also unclear where the boundary $\rho_k = \gamma_k$ between \mathcal{Q}^* and \mathcal{Q}^\bullet lies. The complexity of these problems obviously grows with increasing dimension of \mathcal{Q} . Furthermore, no information on the probability of the set \mathcal{Q}^* is given, *i.e.*, how often the specification is satisfied, and how the system performs on average. Relying on verification testing also increases the likelihood that manufacturers design their systems to meet a specific verification test, as is sometimes done with the Euro emission standards (so-called ‘cycle-beating’ [127]). System validation using a predetermined and limited set of operating conditions should therefore not be considered suitable for future ADAS validation procedures.

5.1.4 Evaluation of the cleared parameter set

Both for controller synthesis and validation it is necessary to gain more insight in the performance of the system over the entire parameter set \mathcal{Q} . We therefore define the following validation problem.

Problem 5.2 Evaluation of the cleared parameter set

Validate that the criteria γ_k are satisfied for the entire parameter set \mathcal{Q} , that is $\rho_k(\mathbf{q}) < \gamma_k$ for all $\mathbf{q} \in \mathcal{Q}$, where \mathcal{Q} is an n -dimensional set in \mathbb{R}^n . If there are \mathbf{q} for which γ_k is not satisfied, *i.e.*, $\rho_k(\mathbf{q}) \geq \gamma_k$, we would like to know how large the cleared subset $\mathcal{Q}^* = \{\mathbf{q} \in \mathcal{Q} | \rho_k(\mathbf{q}) < \gamma_k\}$ and the non-cleared subset $\mathcal{Q}^\bullet = \{\mathbf{q} \in \mathcal{Q} | \rho_k(\mathbf{q}) \geq \gamma_k\}$ are. In addition, it is of interest to know where the boundary \mathbf{q}_γ lies between the cleared and non-cleared subsets of the parameter set. \square

For a linear low-order system as in (5.1), μ -analysis techniques can be used to guarantee robust stability and performance for all possible combinations of the values of uncertain parameters. These techniques are however restricted to multivariable linear models and frequency domain clearance criteria, and they are not well-suited for complex models with nonlinear and time-domain clearance criteria. With the complexity of the system, the control validation problem scales up, due to the exponential increase in volume associated with adding extra dimensions to the parameter set.

This so-called *curse of dimensionality* has given rise to the notion of *tractability*, as discussed by Vidyasagar [255]. Let $f(m)$ denote the maximum number of operations required by an algorithm on any problem instance of size m . If the function $f(m)$ increases no faster than some polynomial in m , then the algorithm is said to be *polynomial-time*. A particular problem is considered to be *tractable*, if a polynomial-time algorithm can be found for that problem.

Unfortunately, for a realistic nonlinear vehicle model with an increasing dimension of \mathcal{Q} , clearance of the entire parameter set will become more difficult to solve, and eventually become *intractable*. That is, it is unknown whether there exists a polynomial-time solution to the problem.

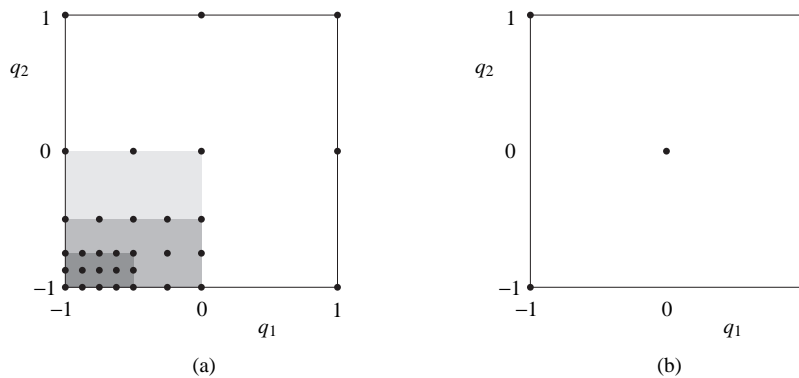


Figure 5.2: Validation of \mathcal{Q} with: (a) grid search; (b) fractional factorial design.

5.1.5 Grid-based techniques to cover the parameter set

In order to provide a polynomial-time solution, a controller is often validated with a *grid search* over the operating range of all parameters, which is common practice in the aerospace industry [61]. However, the procedure to put a grid on the parameter set and do all evaluations at the grid points, as illustrated in Figure 5.2(a), does not give an answer about the behavior between these points. For critical subsets in \mathcal{Q} the number of grid points could be increased (as illustrated). The hope is that with a very narrow grid, all critical points will be found, but a guarantee can never be given. Furthermore, a thorough grid search requires a very large number of experiments, often too large to be feasible. Although a grid search is a tractable solution, it therefore has low *efficiency*, in terms of the number of function evaluations required to reach a desired accuracy in the validation.

A method to improve the efficiency of grid-based validation and gain insight in the system behavior is *design of experiments* [165]. Design of experiments provides a statistical means for analyzing how parameters interact in an experiment, and how they influence the performance measure. It can be effectively used to determine cause-and-effect relationships in experiments. Yet, the most important characteristic is that it reduces the number of experiments that are required to analyze these relationships.

This reduction is accomplished through a *fractional factorial design*, which consists of a carefully chosen subset (fraction) of the experimental runs of a full grid search, as illustrated in Figure 5.2(b). The subset is chosen so as to access information about the most important features of the problem studied, such as the main effects and interactions of the elements q_i of \mathbf{q} on the performance measure ρ_k . The resulting design requires considerably fewer experiments than a full factorial design (*i.e.*, a full grid search). In case *a priori* information on the cost function $\rho_k(\mathbf{q})$ is available, an optimal subset can be configured using D-optimal design methods [165].

Using the information obtained from design of experiments, a response surface can be constructed, such that the shape of multivariate cost functions can be investigated. The *response surface methodology* explores the relationships between several parameters and one or more performance criteria [165], as will be illustrated later on in Example 5.8.

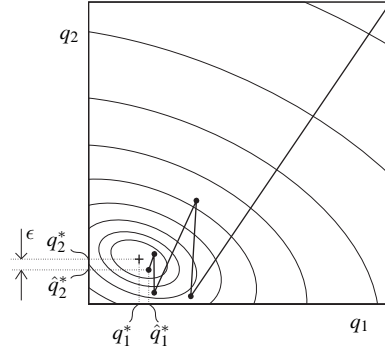


Figure 5.3: Example of a worst-case search process for a two-dimensional parameter set.

5.1.6 Evaluation of worst-case performance

Instead of investigating the *entire* parameter set \mathcal{Q} , as discussed above, an alternative approach is to search for the *worst-case* performance ρ_k^* , formalized as follows.

Problem 5.3 Evaluation of worst-case performance

Estimate the global worst-case performance, defined by

$$\rho_k^* = \min_{\mathbf{q} \in \mathcal{Q}} \rho_k(\mathbf{q}). \quad (5.2)$$

In case the worst-case performance is defined by a maximum, the maximization problem can easily be rewritten as a minimization problem, since $\max_{\mathbf{q} \in \mathcal{Q}} \rho_k(\mathbf{q}) = -\min_{\mathbf{q} \in \mathcal{Q}} (-\rho_k(\mathbf{q}))$. \square

The corresponding parameter vector \mathbf{q}^* is defined as the location of ρ_k^* . If this worst-case point can be cleared, it implies that the controller can be cleared for the entire \mathcal{Q} . Since there is usually no analytical solution for (5.2), a search algorithm of some sort must be used. We can therefore define an optimization problem, and find the worst-case performance ρ_k^* at the local worst-case point \mathbf{q}_j^* within an error tolerance ϵ , as illustrated in Figure 5.3.

Compared to a conventional grid search, the optimization-based validation offers a considerable improvement in efficiency, and it is very suitable for simulation purposes. This occurs because the number of function evaluations required to compute the worst-case parameter combination is usually much lower than that corresponding to evaluating the function in all vertex points of a grid. A second advantage is achieved by allowing a continuous variation of parameters within the given parameter set \mathcal{Q} , which reduces the risk of missing critical subsets of \mathcal{Q} .

On the other hand, a cost function may exhibit multiple local minima, such as the non-convex example in Figure 5.1. In that case a computationally expensive global search algorithm must be used, and it cannot be guaranteed that the optimization converges to the global minimum. An optimization algorithm only results in *one* parameter combination \mathbf{q}_j^* , for which the controller may or may not be cleared. In case $\rho_k^* > \gamma$, little information is available on the size of the non-cleared subset of \mathcal{Q} . Optimization-based validation is therefore only useful in applications, where *every* point in \mathcal{Q} must be cleared, such as in the aerospace industry. However, tuning an ADAS controller to the results of a worst-case analysis may result in a too conservative controller, if this worst-case scenario is an extremely

rare event. It is therefore unreasonable to expect the controller to exhibit satisfactory performance for every point $\mathbf{q} \in \mathcal{Q}$. Instead, a small failure probability is considered to be acceptable.

5.1.7 The need for a new methodology

Because of the impossibility of exhaustive testing, an iterative ad-hoc process of simulations and test drives is often used for validation. Although the results of test drives are realistic, the test schedule can never cover the entire set of operating conditions, due to time and cost constraints. Especially rare events are difficult to reproduce and analyze by using test drives.

Conversely, simulations have their limitations with regard to the ability to model the traffic environment realistically, and with regard to the validation of those models. This increases the model uncertainty Δ and consequently the error in the estimate of the performance measure. Unfortunately, little information is available on the accuracy of the estimated performance and the associated confidence. We would therefore like to obtain a guarantee on the performance and dependability of a system with a desired level of accuracy and confidence.

In order to improve the transition from simulations to test drives, the VeHIL laboratory was introduced in the previous chapter. VeHIL allows to validate an ADAS in an early stage of development, and provides the possibility for model validation. However, an efficient test schedule is essential for testing an ADAS with simulations, VeHIL, and test drives.

The objective of this chapter is therefore to develop a methodological framework for ADAS control system validation. A key element of this methodology is the generation of an efficient test schedule consisting of the *minimum* number of experiments that is sufficient to validate the ADAS with a desired level of accuracy and confidence¹. As discussed earlier, this problem cannot be solved *exactly* and is a computationally 'hard' problem.

5.2 Randomized algorithms for control system validation

5.2.1 Motivation for a probabilistic approach

An alternative approach for solving a complex problem exactly, is to solve it *approximately* by using a *randomized algorithm*. A randomized algorithm is an algorithm that makes random choices during its execution, and covers sequential, Las Vegas, Monte Carlo, and other probabilistic algorithms, as investigated in detail by Motwani [169] and Tempo [237].

The use of a randomized algorithm can turn an intractable problem into a *tractable* one, but at the cost that the algorithm may fail to give a correct solution. The probability δ that the randomized algorithm fails can be made arbitrarily close to zero, but never exactly equal to zero. This probability δ mainly depends on the *sample complexity*, *i.e.*, the number of function evaluations, but also on the type of problem to be solved.

A popular example of a randomized algorithm is the Monte Carlo simulation strategy, where the system is simulated for a representative, though very large, set of operating conditions, based on the probability that these conditions occur. Monte Carlo simulation methods are especially useful in studying systems with a large number of coupled degrees of freedom

¹Note the difference between *necessary* and *sufficient* conditions. It is often impossible to determine the minimum number of experiments that is necessary for validation, whereas it is often much simpler to determine a sufficient number, which has a larger but still acceptable size.

and find application in a variety of fields, such as physics, computer science, reliability engineering, and financial mathematics. Monte Carlo simulation is also widely used for design and analysis of control system performance, such as in the aerospace industry [61, 225], process industry [275], and automotive industry. Several authors discuss the application of Monte Carlo simulations to the evaluation of ADAS controllers. Michael *et al.* [163] present the validation of a lateral vehicle control system, Touran *et al.* [242] show the safety effects of ACC by a Monte Carlo analysis, and Godbole *et al.* [88] evaluate a collision avoidance system, based on probabilistic assumptions for scenario and driver parameters. More recent applications of randomized algorithms include controller synthesis and robustness analysis for systems with bounded disturbances, as discussed in the work by Tempo *et al.* [236] and Vidyasagar [254].

The main advantage of Monte Carlo is that it can take into account a representative part of a high-dimensional parameter set, including disturbances and failure modes. Furthermore, it allows the simulation of black-box models, where no knowledge of the underlying process is necessary. On the other hand, this means that no use is made of any *a priori* information on the system to make the validation process more efficient. Monte Carlo also allows for dynamic testing, such that the time-dependency between certain inputs is taken into account, and time-domain performance measures can be used.

The downside of Monte Carlo simulation is the size of the sufficient sample complexity that guarantees a certain accuracy and level of confidence for the simulation outcome. In this section we demonstrate that this sample complexity is bounded, but also that these bounds are rather conservative.

5.2.2 Problem definition for Monte Carlo sampling

The use of a randomized algorithm for controller validation is illustrated by considering the following problem.

Problem 5.4 Probability of clearance

Consider an arbitrary process with only two possible outcomes, ‘cleared’ ($\rho < \gamma$) and ‘non-cleared’ ($\rho \geq \gamma$). The problem is then to determine the probability p of a non-cleared outcome of the process for all $\mathbf{q}_j \in \mathcal{Q}$, such that the cleared parameter set \mathcal{Q}^* and the non-cleared parameter set \mathcal{Q}^\bullet can be evaluated in a *probabilistic* sense². \square

Problem 5.4 and the sample complexity are investigated as follows. In order to estimate p , we generate N independent identically distributed (iid) samples $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_N$ in the parameter set \mathcal{Q} according to its probability density function (PDF) $f_{\mathcal{Q}}$. The outcome of every j -th experiment is represented by an indicator function J :

$$J(\mathbf{q}_j) = \begin{cases} 0, & \text{if } \rho(\mathbf{q}_j) < \gamma \\ 1, & \text{if } \rho(\mathbf{q}_j) \geq \gamma \end{cases}, \quad (5.3)$$

where $J = 0$ is a successful outcome (the performance ρ satisfies the requirement γ and the parameter \mathbf{q} is cleared), and $J = 1$ is a faulty outcome (ρ does not satisfy γ and \mathbf{q} is

²Note the difference in notation between ρ and p . The performance level ρ applies to *one* particular experiment, whereas the probability p applies to *all* experiments. For instance, the safety of a single experiment can be expressed as $\rho = 0$ (no collision) or $\rho = 1$ (collision), whereas the corresponding collision probability is expressed in the form of $p = 0.001$, in case one in a thousand scenarios results in a collision.

non-cleared). The probability p of an unsuccessful outcome can then be estimated by the *empirical probability* $\hat{p}_N(\gamma)$ as³

$$\hat{p}_N(\gamma) = \frac{1}{N} \sum_{j=1}^N J(\mathbf{q}_j). \quad (5.4)$$

Equation (5.4) is also known as the *simple sampling* estimator and is an unbiased estimator, since the expected value of \hat{p}_N is p :

$$\mathbb{E}\{\hat{p}_N\} = \mathbb{E}\left\{\frac{1}{N} \sum_{j=1}^N J(\mathbf{q}_j)\right\} = \frac{1}{N} \mathbb{E}\left\{\sum_{j=1}^N J(\mathbf{q}_j)\right\} = \frac{1}{N} \sum_{j=1}^N \mathbb{E}\{J(\mathbf{q}_j)\} = \frac{1}{N} Np = p. \quad (5.5)$$

The simple sampling variance σ_{ss}^2 of \hat{p}_N is given by

$$\begin{aligned} \text{var}\left\{\frac{1}{N} \sum_{j=1}^N J(\mathbf{q}_j)\right\} &= \frac{1}{N^2} \text{var}\left\{\sum_{j=1}^N J(\mathbf{q}_j)\right\} = \frac{1}{N^2} N \text{var}\{J(\mathbf{q}_j)\} = \\ &= \frac{1}{N} \left(\mathbb{E}\{(J(\mathbf{q}_j))^2\} - (\mathbb{E}\{J(\mathbf{q}_j)\})^2\right) = \frac{1}{N} (p - p^2) = \frac{p(1-p)}{N}. \end{aligned} \quad (5.6)$$

The accuracy of the estimate \hat{p}_N can be expressed in the relative root mean square (RMS) error

$$\sqrt{\frac{\text{var}\{\hat{p}_N\}}{p^2}} = \sqrt{\frac{1-p}{pN}}. \quad (5.7)$$

From (5.7) the necessary sample complexity can then be calculated, given a desired relative RMS error and an expected value for p . However, this measure requires *a priori* knowledge of p , which is exactly the parameter we wish to estimate! Furthermore, we do not know what confidence to attach to any particular estimate \hat{p}_N . In that respect, \hat{p}_N is unlikely to be exactly equal to the real probability p , although the ‘law of large numbers’ [254] states that \hat{p}_N converges to p with probability 1, as $N \rightarrow \infty$, and as long as the samples are chosen to be representative of the set \mathcal{Q} .

The question thus arises in what sense \hat{p}_N converges to p , and how many samples N are required to estimate \hat{p}_N , such that it differs from the real (unknown) value p by no more than $\epsilon > 0$, *i.e.*,

$$|p - \hat{p}_N| \leq \epsilon, \quad (5.8)$$

where ϵ is referred to as the *accuracy* of the estimate.

Since \hat{p}_N is a random variable, depending on the particular realization of N samples, the outcome of the inequality (5.8) is a random variable as well with a certain probability of realization. Therefore, we cannot always guarantee that $|p - \hat{p}_N| \leq \epsilon$, even for very large N . This means that if the experiments are performed another N times, the estimate \hat{p}_N will probably have another value. By introducing a *confidence level* $1 - \delta$ with $\delta > 0$, the probability that $|p - \hat{p}_{N_\ell}| \leq \epsilon$ for any ℓ -th set of N experiments (denoted by N_ℓ) is then defined as

$$\Pr\{|p - \hat{p}_N| \leq \epsilon\} \geq 1 - \delta. \quad (5.9)$$

It is then of interest to know the required N for (5.9) to hold. In other words: how many samples N are necessary to achieve a desired level of accuracy ϵ and confidence $1 - \delta$?

³In the following we will use \hat{p}_N instead of $\hat{p}_N(\gamma)$ for reasons of brevity.

5.2.3 Upper bounds on the sample complexity

In order to derive the required N for a Monte Carlo simulation *a priori*, we would like to determine an upper bound on the sample complexity.

Binomial bound

Since Problem 5.4 is governed by a binomial process with only two possible outcomes, (5.9) can be rewritten as

$$\Pr\{|p - \hat{p}_N| \leq \epsilon\} = \sum_{j=0}^{N^-} \binom{N}{j} p^j (1-p)^{(N-j)} + 1 - \sum_{j=N^+}^N \binom{N}{j} p^j (1-p)^{(N-j)} \geq 1 - \delta \quad (5.10)$$

where $N^- = (p - \epsilon)N$ is the minimum and $N^+ = (p + \epsilon)N$ the maximum number of samples for which it can be expected that $J(\mathbf{q}_j) = 1$. Unfortunately, N cannot be solved recursively from (5.10), such that the necessary sample complexity N has to be approximated.

For moderate values of p , the binomial distribution approaches the normal distribution if the number of samples N is large (as a rule of thumb pN and $(1-p)N$ must both be larger than 5). This is a result of the central limit theorem, which states that any sum of many iid random variables will tend to be distributed according to a normal distribution. An important property of the normal distribution is that it is possible to relate all normal random variables to the standard normal with mean $\mu = 0$ and variance $\sigma^2 = 1$. If the variate $X \sim \mathcal{N}(\mu, \sigma^2)$, then

$$Z = \frac{X - \mu}{\sigma} \quad (5.11)$$

is a standard normal random variable: $Z \sim \mathcal{N}(0, 1)$ with cumulative distribution function

$$F(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z \exp\left(-\frac{u^2}{2}\right) du, \quad z \in \mathbb{R}. \quad (5.12)$$

The mean of a binomial distributed variable (*i.e.*, the expected number of positive samples) equals pN , whereas the standard deviation σ can be approximated by $\sqrt{Np(1-p)}$. With a desired accuracy ϵ the likely number of positive samples is in the interval $[(p - \epsilon)N, (p + \epsilon)N]$. The required N can then be found, when the desired probability δ is equal to the sum of the values of the cumulative distribution function of the standard normal distribution:

$$\delta = \frac{1}{\sqrt{2\pi}} \int_0^{z^-} \exp\left(-\frac{u^2}{2}\right) du + \frac{1}{\sqrt{2\pi}} \int_{z^+}^N \exp\left(-\frac{u^2}{2}\right) du, \quad z^-, z^+ \in \mathbb{N}, \quad (5.13)$$

evaluated at $z^- = \frac{(p - \epsilon)N - pN}{\sqrt{Np(1-p)}}$ and $z^+ = \frac{(p + \epsilon)N - pN}{\sqrt{Np(1-p)}}$.

An algorithm can then easily be constructed that gives an efficient approximation for the least upper bound N , denoted by N_{bin} . Unfortunately, this measure again requires *a priori* knowledge of p , which is exactly the parameter we wish to estimate.

Additive Chernoff bound

To avoid this paradox we can use the Chernoff bound [34], which states that the probability $\delta > 0$ in (5.9) is no larger than $2e^{-2N\epsilon^2}$. Chernoff's method centers around bounding the

random variable p by investigating the random variable e^{tP} rather than the variable p itself:

$$\Pr \left\{ e^{t\hat{p}_N} > e^{t(\hat{p}_N + \epsilon)} \right\} < \frac{E\{e^{t\hat{p}_N}\}}{e^{t(\hat{p}_N + \epsilon)}}, \quad \text{for all } t > 0, \quad (5.14)$$

and applying Markov's inequality [236] to the right-hand side of this formula. See [169] for the lengthy mathematical derivation of this bound. This means that, after performing the experiment N times, we can state with a confidence of at least $1 - 2e^{-2N\epsilon^2}$ that the empirical probability \hat{p}_N is no more than ϵ different from the true but unknown probability p . Therefore, to estimate the unknown quantity p to an accuracy ϵ and with a confidence $1 - \delta$, N should be chosen such that $2e^{-2N\epsilon^2} \leq \delta$. This can be rewritten as

$$N \geq \frac{1}{2\epsilon^2} \ln \frac{2}{\delta}. \quad (5.15)$$

The right-hand side of this inequality is known as the *two-sided additive Chernoff bound*, denoted as N_{Ch} .

One-sided bounds

The validation objective that is investigated in this chapter, is related to estimating the probability of failure p , since this usually is a very small value close to zero. This small probability is the reason that so many samples are necessary to estimate it. Obviously, it is important to know whether the real probability of a failure p is larger than the maximum expected value $\hat{p}_N + \epsilon$, since the estimated \hat{p}_N will then give a false sense of security. Vice versa, it is less critical to know whether the real p is ϵ less than the estimated \hat{p}_N , since this is actually a desired outcome. We therefore propose to only consider the probability that $p - \hat{p}_N > \epsilon$. Instead of the two-sided bound we can therefore use the *one-sided Chernoff bound*, which is defined as

$$N_{\text{Ch}}^- = \frac{1}{2\epsilon^2} \ln \frac{1}{\delta}, \quad (5.16)$$

and we should have $N \geq N_{\text{Ch}}^-$. Similarly, we can define the one-sided binomial bound, as opposed to the two-sided bound in (5.10):

$$\Pr \{p - \hat{p}_N \leq \epsilon\} = 1 - \sum_{j=0}^{N^-} \binom{N^-}{j} p^j (1-p)^{(N^- - j)} > 1 - \delta, \quad (5.17)$$

and modify (5.13) into

$$\delta = \frac{1}{\sqrt{2\pi}} \int_0^{z^-} \exp\left(-\frac{u^2}{2}\right) du \quad z^- \in \mathbb{N}. \quad (5.18)$$

Comparison of sample complexity

The sample complexities corresponding to the above-mentioned bounds are shown in Figure 5.4. For ease of reference, Table 5.1 gives the bounds for some characteristic values of ϵ and δ . Since these bounds are associated with a confidence interval $1 - \delta$, they are referred to as *soft* bounds, as opposed to *hard* bounds that are always guaranteed [277]. Nevertheless, the table illustrates that the Chernoff bound is very conservative (that is, the ratio of the

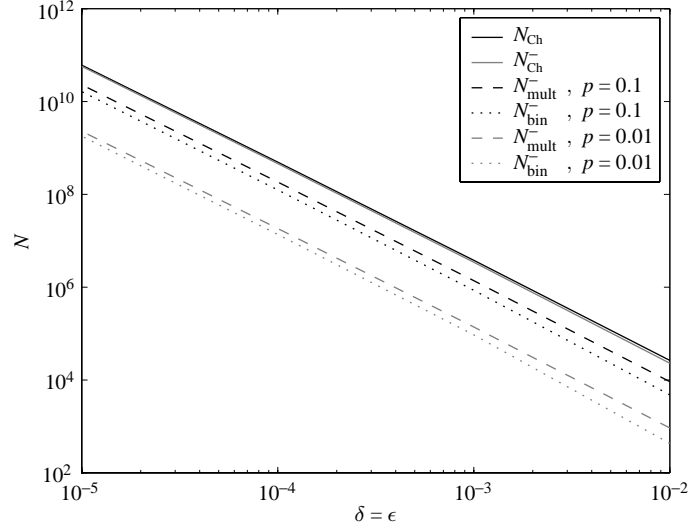


Figure 5.4: Binomial and Chernoff bounds for several values of p , ϵ and δ .

Table 5.1: Chernoff and binomial bounds for several values of ϵ , δ , and the probability p .

p	δ	ϵ	ϵ_r	N_{Ch}	N_{Ch}^-	N_{mult}^-	N_{bin}^-
0.1	0.01	0.01	0.1	$2.65 \cdot 10^4$	$2.30 \cdot 10^4$	$9.21 \cdot 10^3$	$4.77 \cdot 10^3$
0.1	0.01	0.001	0.01	$2.65 \cdot 10^6$	$2.30 \cdot 10^6$	$9.21 \cdot 10^5$	$4.86 \cdot 10^5$
0.1	0.001	0.01	0.1	$3.80 \cdot 10^4$	$3.45 \cdot 10^4$	$1.38 \cdot 10^4$	$8.50 \cdot 10^3$
0.1	0.001	0.001	0.01	$3.80 \cdot 10^6$	$3.45 \cdot 10^6$	$1.38 \cdot 10^6$	$8.58 \cdot 10^5$
0.01	0.001	0.001	0.1	$3.80 \cdot 10^6$	$3.45 \cdot 10^6$	$1.38 \cdot 10^5$	$9.35 \cdot 10^4$
0.01	0.001	0.0001	0.01	$3.80 \cdot 10^8$	$3.45 \cdot 10^8$	$1.38 \cdot 10^7$	$9.44 \cdot 10^6$
0.01	0.0001	0.001	0.1	$4.95 \cdot 10^6$	$4.61 \cdot 10^6$	$1.84 \cdot 10^5$	$1.36 \cdot 10^5$
0.01	0.0001	0.0001	0.01	$4.95 \cdot 10^8$	$4.61 \cdot 10^8$	$1.84 \cdot 10^7$	$1.36 \cdot 10^7$

sufficient N to the necessary N is very large), and that the conservatism increases for lower values of p , ϵ , and δ . In that sense, the Chernoff bound (5.16) only provides an *upper* bound, unlike the *least* upper bound of the binomial bound (5.17). Reduction of N is therefore an important challenge, especially for values of p , ϵ , and δ close to zero.

5.2.4 Formulation of a randomized algorithm

For now we will use the one-sided Chernoff bound (5.16) for estimation of the probability p . The corresponding procedure for probabilistic control system validation using Monte Carlo methods is formalized in Algorithm 5.1. The conservatism of this algorithm is illustrated with an example.

Algorithm 5.1 Probabilistic validation of control performance [236]

Given a desired $\epsilon, \delta \in (0, 1)$, a performance measure ρ , a threshold $\gamma \geq 0$, and the true PDF $f_{\mathcal{Q}}$ for \mathcal{Q} , this randomized algorithm returns with a probability of at least $1 - \delta$ an estimate \hat{p}_N for p , such that $p - \hat{p}_N \leq \epsilon$.

1. Determine the sufficient N_{Ch}^- with the one-sided Chernoff bound

$$\frac{1}{2\epsilon^2} \ln \frac{1}{\delta}.$$

2. Draw $N = N_{\text{Ch}}^-$ iid samples $\mathbf{q}_1, \dots, \mathbf{q}_N$ in the set \mathcal{Q} according to its PDF $f_{\mathcal{Q}}$.
3. Return the empirical probability

$$\hat{p}_N = \frac{1}{N} \sum_{j=1}^N J(\mathbf{q}_j),$$

where $J(\mathbf{q}_j)$ is the indicator function

$$J(\mathbf{q}_j) = \begin{cases} 0, & \text{if } \rho(\mathbf{q}_j) < \gamma \\ 1, & \text{if } \rho(\mathbf{q}_j) \geq \gamma \end{cases}.$$

Example 5.2 Gaussian disturbance with continuous performance measure.

Problem definition Consider the ACC control problem of Section 5.1.1, where we use the constant time headway control law (2.12) with t_h equal to 2 s and the safety margin $s_0 = 6$ m. The fixed feedback gains are $K_1 = 0.17 \text{ s}^{-2}$, and $K_2 = 0.7 \text{ s}^{-1}$. Sensor processing delay and vehicle dynamics are neglected by assuming that the desired acceleration is realized at the input of the controlled system without any time lag, such that $a_2 = a_{\text{ref}}$. However, we do introduce an actuator saturation, since ACC systems usually restrict the minimum and maximum control input for safety reasons. In this case study we use the restriction that a_2 is bounded between -3 and 3 m/s^2 .

Consider a steady-state car-following scenario where the target vehicle suddenly brakes to a full stop. The initial conditions are $x_r(0) = t_h v_2(0) + s_0 = 66$ m and $v_1(0) = v_2(0) = 30$ m/s. We assume that the deceleration of the preceding vehicle is the only disturbance, defined as a Gaussian distributed signal with mean $\mu = 0$ and standard deviation $\sigma = 1.5$, denoted as $\mathcal{N}(0, 1.5)$, truncated on the interval $[-10, 10] \text{ m/s}^2$.

In situations when the target vehicle brakes hard, the ACC vehicle may not obtain the required deceleration a_{ref} , since the actuator saturates at -3 m/s^2 . Now, for fine-tuning the controller parameters, we would like to know the percentage of brake situations for which the TTC falls below a certain threshold. We therefore define a *continuous* safety measure ρ_{TTC} , defined by the TTC $t_{\text{TTC}} = \frac{x_r}{v_r}$. The TTC has a lower value for unsafe situations, with a threshold value $\gamma_{\text{TTC}} = 6$ s, since a traffic situation is subjectively regarded by a driver as ‘dangerous’ when the TTC is less than 6 s [58]. The safety obviously decreases with a stronger deceleration of the target vehicle a_1 , such that the function $\rho_{\text{TTC}}(a_1)$ is non-increasing and crosses the threshold of $\gamma_{\text{TTC}} = 6$ s at $a_1 = -2.69 \text{ m/s}^2$.

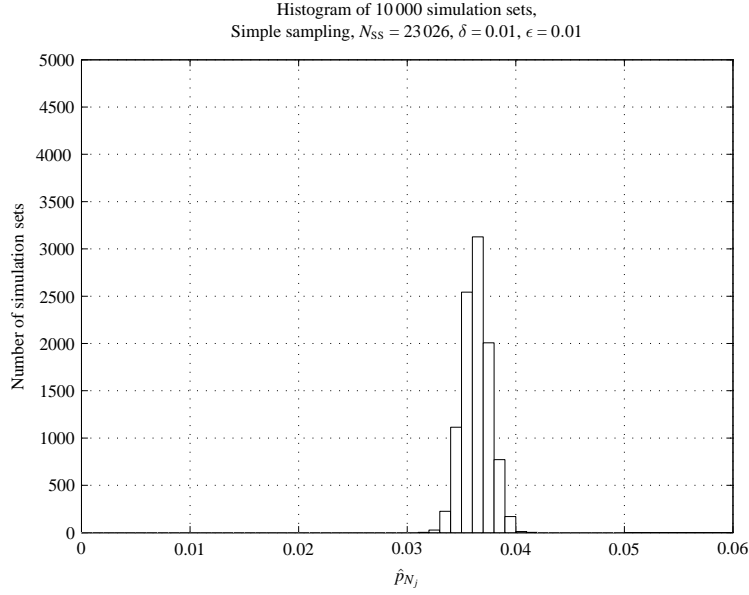


Figure 5.5: Histogram of 10 000 estimates \hat{p}_{N_ℓ} , with $N_\ell = 23\,026$ each, where the acceleration profile is sampled from a Gaussian PDF $\mathcal{N}(0, 1.5)$.

For this example the indicator function is defined as

$$J(a_1) = \begin{cases} 0, & \text{if } \rho_{\text{TTC}}(a_1) > \gamma_{\text{TTC}} \\ 1, & \text{if } \rho_{\text{TTC}}(a_1) \leq \gamma_{\text{TTC}} \end{cases}. \quad (5.19)$$

For verification of this simple example the ‘true’ outcome can be calculated analytically and is known exactly: $p = 0.03630$.

Randomization of the problem Although for this simple problem it is feasible to formulate a deterministic algorithm, in practice it can be difficult or even impossible to determine p in a deterministic way, when the dimension of \mathcal{Q} increases and the function ρ_{TTC} is not monotonous. So instead of calculating p explicitly in a deterministic sense, the function is randomized in such a way that it takes a random input \mathbf{q}_j from its distribution function $f(a_1)$, according to Algorithm 5.1.

In order to verify the performance of Algorithm 5.1, we execute it $M = 10\,000$ times. Suppose it is desired that $\epsilon = 0.01$ and $\delta = 0.01$, corresponding to values commonly used in the automotive industry. The Chernoff bound (5.16) then gives $N = 23\,026$. So each ℓ -th simulation set (with $\ell = 1, \dots, M$) consists of 23 026 simulation runs and gives an estimate \hat{p}_{N_ℓ} . The distribution of this estimate is shown in Figure 5.5. With this example, the accuracy and confidence for a single simulation set can be estimated from this histogram as $\hat{\epsilon}$ and $1 - \hat{\delta}$ respectively.

Analysis of the simulation results The empirical mean of the histogram \hat{p}_M is 0.03633, based on all $M \cdot N = 2.3 \cdot 10^8$ simulations, which is very close to $p = 0.03630$. The variance of each individual estimate \hat{p}_{N_ℓ} can be found by the unbiased estimator for the

variance

$$\hat{\sigma}_{N_\ell-1}^2 = \frac{1}{N_\ell-1} \sum_{\ell=1}^{N_\ell} (\hat{p}_{N_\ell} - \hat{p}_M)^2. \quad (5.20)$$

The simple sampling variance $\hat{\sigma}_{SS}^2$ of the estimator \hat{p}_{N_ℓ} is $1.5373 \cdot 10^{-6}$. The empirical accuracy for a single simulation set (each consisting of 23 026 simulations), can be estimated from the histogram as $\hat{\epsilon} = 0.0028$ when $\hat{\delta} = 0.01$. That is, only 1% of the simulation sets result in an estimate \hat{p}_{N_ℓ} that differs from the grand mean \hat{p}_M by more than 0.0028.

Example 5.2 demonstrates that by choosing $N = N_{Ch}^-$, a higher level of accuracy and confidence is obtained than desired ($\hat{\epsilon} = 0.0028$ instead of $\epsilon = 0.01$). Conversely, this suggests that the desired values for δ and ϵ can be achieved with a much lower N than given by (5.16). Although the degree of conservatism for this example is rather limited, it can be shown that the Chernoff bound conservatism increases with smaller values for δ and ϵ [255].

5.2.5 Characteristic properties of randomized algorithms

Example 5.2 illustrates some characteristic properties of randomized algorithms that were also identified by Tempo *et al.* [236] and Vidyasagar [255].

- A randomized algorithm is very simple, since the Chernoff bound is completely independent of the nature of the underlying process and the parameter set \mathcal{Q} . However, this also means that no advantage is taken of any *a priori* knowledge of the structure of \mathcal{Q} . It is therefore desired to modify the simulation approach such that knowledge of the system is applied. Section 5.3 will take this into account.
- A randomized algorithm, as compared to deterministic algorithms, requires a limited number of test runs, such that it can be run in polynomial time. However, the sample complexity N to achieve a reasonable ϵ and δ , as given by the Chernoff bound, is quite conservative. We will therefore consider this issue in Section 5.4 where a new randomized algorithm with a reduced sample complexity is developed.
- The confidence of the simulation outcome strongly depends on the reliability of the pre-defined PDF $f_{\mathcal{Q}}$. The outcome of the simulation approach also greatly depends on the modeling effort of the system that is investigated. Section 5.5 will therefore present a method for validation of the simulation results with VeHIL and test drives.

5.3 Methods for reduction of the sample complexity

This section investigates several possibilities to reduce the sample complexity N :

- Reduction of the parameter set that is used for the validation problem.
- Reformulation of the control validation problem, such that another bound on the sample complexity can be derived.
- Use of another sampling distribution.
- Increasing the spread and randomness of random sample generation, thereby reducing the variance of the estimate.

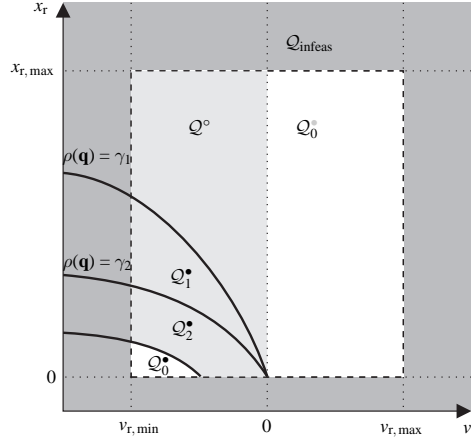


Figure 5.6: A two-dimensional example of a parameter set \mathcal{Q} in the form of the range – range rate diagram. The set \mathcal{Q} is composed of a feasible subset $\mathcal{Q}_{\text{feas}}$ and an infeasible subset $\mathcal{Q}_{\text{infeas}}$ (dark grey). In turn the feasible subset is composed of the a priori known sets \mathcal{Q}_0^* and \mathcal{Q}° (both white), and the unknown set \mathcal{Q}° (light grey).

5.3.1 Reduction of the parameter set

With *a priori* knowledge on the cost functions ρ_k the problem complexity of \mathcal{Q} can be reduced. From experimental data or heuristic knowledge the shape of the cost functions can be identified, which can help to simplify the validation problem. In case there are parameter combinations that are infeasible, only a subset of $\mathcal{Q}_{\text{feas}} \subset \mathcal{Q}$ is considered feasible. This is due to inequality constraints $g(\mathbf{q}) \leq 0$, such that $\mathcal{Q}_{\text{feas}}$ is defined as $\{\mathbf{q} | \mathbf{q} \in \mathcal{Q}, g(\mathbf{q}) \leq 0\}$. In Example 5.2 we assumed that the target vehicle acceleration a_1 is within the interval $[-10, 10] \text{ m/s}^2$, although a Gaussian PDF has tails that extend beyond the extreme values of the interval. In addition, any positive value for a_1 can be considered safe, since the resulting relative velocity v_r will always be positive in that case. In this way, a cleared subset \mathcal{Q}_0^* can be identified beforehand that does not need to be included in the validation process. For this one-dimensional example, reduction of the parameter set to the interval $[-10, 0] \text{ m/s}^2$ would mean that N can be halved. It is important to note that the underlying PDF must be scaled, such that $\int_{\mathcal{Q}_{\text{feas}}} f_{\mathcal{Q}_{\text{feas}}}(\mathbf{q}) = \int_{\mathcal{Q}} f_{\mathcal{Q}}(\mathbf{q}) = 1$.

Figure 5.6 gives an example of a constrained two-dimensional parameter set \mathcal{Q} in the form of the range – range rate diagram, which is often used for analysis of ACC controllers. The range to a target vehicle is bounded between zero and the maximum range of the ACC sensor $x_{r,\max}$. Similarly, the range rate v_r is bounded between minimum and maximum values that are considered possible. The parameter set \mathcal{Q} is thus composed of the feasible set $\mathcal{Q}^* \cup \mathcal{Q}^\circ$ and the infeasible set $\mathcal{Q}_{\text{infeas}}$. The subsets $\mathcal{Q}_0^* \subset \mathcal{Q}^*$, $\mathcal{Q}_0^\circ \subset \mathcal{Q}^\circ$ and $\mathcal{Q}_{\text{infeas}}$ can be identified *a priori* in order to reduce the volume of the parameter set, which leaves an unknown subset \mathcal{Q}° to be validated. *A posteriori*, the level line $\rho(\mathbf{q}) = \gamma_1$ can be identified that indicates the boundary between the cleared subset \mathcal{Q}^* and the non-cleared subset \mathcal{Q}_1^* , where the TTC changes from acceptable to unacceptable. The level line $\rho(\mathbf{q}) = \gamma_2$ indicates the boundary between the non-cleared subsets $\mathcal{Q}_1^* \subset \mathcal{Q}^\circ$ and $\mathcal{Q}_2^* \subset \mathcal{Q}^\circ$, where a collision becomes unavoidable.

In addition, analysis of failure modes using FMECA, as discussed in Chapter 4, can be used to identify the critical failure modes \mathbf{f} in the fault set $\mathcal{F} \subset \mathcal{Q}$. Similarly, accident scenario analysis can be carried out to identify the safety-critical scenarios. In this way insight is gained in critical parameter combinations, and subsets that are considered irrelevant can be neglected for the validation. Design of experiments theory provides a set of useful tools for these trend studies, such as a D-optimal design for the parameter set [165]. A D-optimal design has the advantage that it can take into account constraints $g(\mathbf{q}) \leq 0$ in the parameter set and *a priori* insight in the system process (approximative knowledge on the contours $\rho(\mathbf{q})$). In addition, D-optimal designs allow the test engineer to execute only a limited number of test runs, which is useful when the runs are expensive or time-consuming. However, there may still be a large subset of parameter combinations for which the outcome is unknown. Apart from reduction of the parameter set \mathcal{Q} , the sample complexity for the remainder of \mathcal{Q} must still be reduced.

5.3.2 Sequential estimation using the binomial bound

To reduce the sample complexity given by the Chernoff bound, we could try to use one of the other bounds, *e.g.*, the binomial bound N_{bin}^- , but this requires *a priori* knowledge of p . This causality problem can be solved by making an initial rough estimate of p using the Chernoff bound⁴ $N_1 = N_{\text{Ch}}^-$ and modified values for ϵ and δ . After this first sequence the real p is then considered to be less than or equal to the worst-case estimate $\hat{p}_{N_1} + \epsilon_1$ with a confidence interval $1 - \delta_1$. Based on this first estimate, we can then use the binomial bound with a significantly lower bound on the sample complexity in the second sequence.

In order to reduce the total number N for both sequences, while still obtaining the desired ϵ and δ , the values ϵ_i and δ_i in the i -th sequence must be chosen suitably. From (5.16) follows that the sample complexity N_{SS} for simple sampling is proportional to $\ln(1/\delta)$ and inversely proportional to ϵ^2 . Therefore, when δ is decreased and ϵ increased by a suitably chosen factor κ , a lower N can be obtained in a first sequence, by first using $\delta_1 = \delta/\kappa$, and $\epsilon_1 = \kappa\epsilon$. To obtain the desired confidence $1 - \delta$ for \hat{p}_N in a second sequence, δ_2 should be chosen such that $(1 - \delta_1)(1 - \delta_2) \geq 1 - \delta$ (that is, the combined confidence is at least as large as the desired confidence). This is true when $\delta_2 = \delta - \delta_1$, since $(1 - \delta/\kappa)(1 - \delta + \delta/\kappa) \geq 1 - \delta$, for all $\kappa \geq 1$. The accuracy in the second sequence ϵ_2 is set to its desired value ϵ . This procedure is formalized in Algorithm 5.2.

In order to choose a suitable value for κ , we investigate the bounds on $N_{\text{SS},1}$ and $N_{\text{bin},2}$ for different values of κ , p , δ , and ϵ . Figure 5.7 shows the relation between κ and p for fixed values of δ and ϵ . This figure shows that Algorithm 5.2 is optimal when $N_{\text{bin},2} = N_{\text{SS},1}$, since this requires no extra samples in the second sequence. Furthermore, κ can better be selected too large instead of too small. In the former case, only a small number of extra samples $N_{\text{SS},2} = N_{\text{bin},2} - N_{\text{SS},1}$ has to be drawn in the second sequence. In the latter case, the first sequence will draw too many samples $N_{\text{SS},1}$, increasing the total sample complexity. The curves are similar for other values of δ , ϵ and p . Furthermore, the optimal value for κ increases when δ increases, and when ϵ and p decrease. This in turn means that for lower values p (where the Chernoff bound becomes more conservative), this conservatism can be counteracted even more.

⁴In the following the subscripts ⁺ and ⁻, which indicate the one-sided bounds, are omitted for notational convenience. In the remainder of this chapter we will only use the one-sided bounds associated with the probability that $p - \hat{p}_N > \epsilon$, *i.e.*, superscript ⁻, unless indicated otherwise.

Algorithm 5.2 Sequential estimation using binomial bound

Given a desired $\epsilon, \delta \in (0, 1)$, a performance measure ρ , a threshold $\gamma \geq 0$, and the true PDF f_Q for \mathcal{Q} , this randomized algorithm returns with a probability of at least $1 - \delta$ an estimate \hat{p}_N for p , such that $p - \hat{p}_N \leq \epsilon$.

1. Draw $N_{SS,1} = \frac{1}{2\epsilon_1^2} \ln \frac{1}{\delta_1}$ iid samples \mathbf{q}_j , where $\delta_1 = \delta/\kappa$, and $\epsilon_1 = \kappa\epsilon$, and κ is a suitably chosen real number.

2. Return the empirical probability $\hat{p}_{N_{SS,1}} = \frac{1}{N_{SS,1}} \sum_{j=1}^{N_{SS,1}} J(\mathbf{q}_j)$.

The real p is always less than or equal to $\hat{p}_{bin,1} = \hat{p}_{N_{SS,1}} + \epsilon_1$ with confidence $1 - \delta_1$.

3. Determine the sample complexity $N_{bin,2}$ that is associated with $\hat{p}_{bin,1}$, $\delta_2 = \delta - \delta_1$, and $\epsilon_2 = \epsilon$, using (5.18) evaluated at

$$z^- = \frac{(\hat{p}_{bin,1} - \epsilon_2)N_{bin,2} - \hat{p}_{bin,1}N_{bin,2}}{\sqrt{N_{bin,2}\hat{p}_{bin,1}(1 - \hat{p}_{bin,1})}}.$$

4. IF $N_{bin,2} \leq N_{SS,1}$

Return the empirical probability $\hat{p}_N = \hat{p}_{N_{SS,1}}$.

ELSE

Draw $N_{SS,2} = N_{bin,2} - N_{SS,1}$ new samples \mathbf{q}_j .

Return the empirical probability

$$\hat{p}_N = \frac{1}{N_{SS,1} + N_{SS,2}} \sum_{j=1}^{N_{SS,1} + N_{SS,2}} J(\mathbf{q}_j)$$

with accuracy ϵ and confidence level $(1 - \delta_1)(1 - \delta_2) \geq 1 - \delta$.

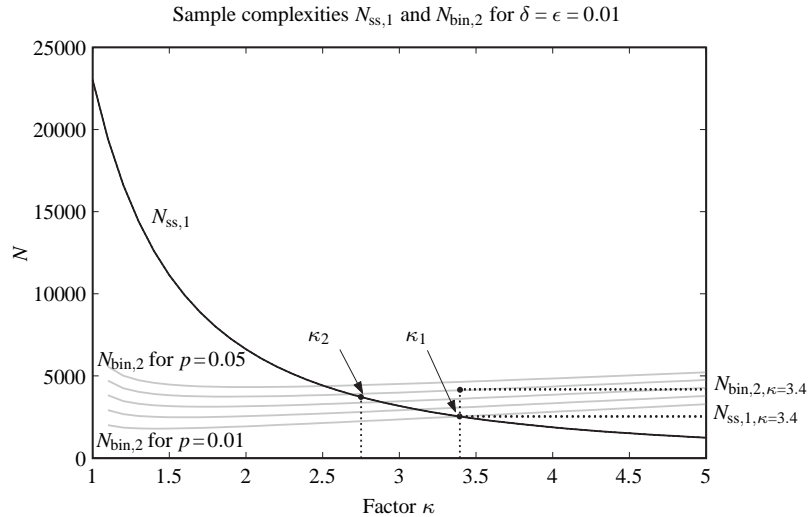


Figure 5.7: Sample complexities $N_{SS,1}$ and $N_{bin,2}$ for varying κ and p with $\delta = \epsilon = 0.01$.

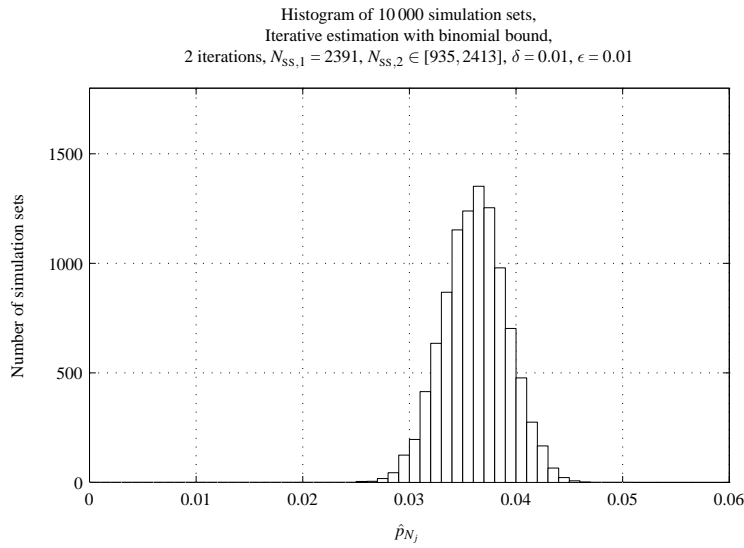


Figure 5.8: Histogram of 10 000 simulation sets, with $N_\ell \in [3326, 4804]$ each, using Algorithm 5.2.

It is therefore preferred to choose κ at a point where $N_{\text{bin},2} = N_{SS,1}$, based on a reasonable *a priori* estimate \hat{p}_0 for p , preferably with \hat{p}_0 as low as possible. Since \hat{p}_0 should be greater than ϵ (having $\hat{p}_0 < \epsilon$ implies the possibility of a negative probability p), \hat{p}_0 is lower bounded by ϵ . Therefore κ should be chosen such that $N_{\text{bin},2} = N_{SS,1}$ and $\hat{p}_0 = \epsilon$.

Example 5.3 Sequential estimation using binomial bound. The problem in Example 5.2 is repeated using Algorithm 5.2, for the same values $\delta = 0.01$, $\epsilon = 0.01$, and with the real p known to be 0.03630. However, since $\epsilon = 0.01$, p is assumed to be lower bounded at 0.01. We therefore choose $\kappa = 3.4$, using the procedure described above. Figure 5.8 shows the results for 10 000 simulation sets, each with a sample complexity range $N_\ell \in [3326, 4804]$, where $N_{SS,1,\ell} = 2391$ and $N_{SS,2,\ell} \in [935, 2413]$. The sample complexity $N_{SS,2}$ in the second sequence is quite large, caused by the fact that the factor κ is chosen larger than the optimal value (*i.e.*, $\kappa_2 = 3.4$ at $\hat{p}_0 = 0.01$, instead of $\kappa_1 = 2.75$ at $p = 0.03630$, see Figure 5.7).

The result is $\hat{p}_N = 0.03619$ with a variance of $8.8040 \cdot 10^{-6}$. Although the variance is larger than in Example 5.2, we should look at the obtained reduction in sample complexity. When the empirical confidence parameter is set $\hat{\delta} = 0.01$ the empirical accuracy is found to be $\hat{\epsilon} = 0.0070$, meaning that $N_\ell \in [935, 2413]$ is still slightly conservative, although it is much better than in Example 5.2 (where $N_\ell = 23026$ with $\hat{\epsilon} = 0.0028$). Compared to the initial Chernoff bound $N_{\text{Ch}}^- = 23026$ for $\epsilon = 0.01$ and $\delta = 0.01$, this means an efficiency improvement with a factor $\frac{N}{\max N_\ell} = 4.8$.

5.3.3 Sequential estimation for a multiplicative accuracy

In the previous sections we have focussed on problems dealing with an additive accuracy, that is: to check that the probability that $p - \hat{p}_N \leq \epsilon$, *i.e.*, the probability that the real probability p is smaller than or equal to the *estimated* $\hat{p}_N + \epsilon$, is larger than $1 - \delta$, where it is

assumed that p represents the probability of an undesired non-cleared parameter combination \mathbf{q} . However, the additive value ϵ may not always be an appropriate measure for the accuracy. What if the empirical probability \hat{p}_N turns out to be much larger or much lower than ϵ ? In case $p \ll \epsilon$, we would be estimating a negative probability $p - \epsilon < 0$, whereas if $p \gg \epsilon$, the sample complexity will be much higher than would be sufficient.

Therefore, it is often more suitable to use a *relative* degree of accuracy, *i.e.*, to check that the probability that $p - \hat{p}_N \leq p\epsilon_r$ is larger than $1 - \delta$, where ϵ_r is the relative (or multiplicative) accuracy. In case of a desired relative accuracy ϵ_r , Algorithm 5.2 can be modified by using the multiplicative Chernoff bound [236], which gives

$$N_{\text{mult}}^- \geq \frac{2}{p\epsilon_r^2} \ln \frac{1}{\delta}. \quad (5.21)$$

An advantage over the additive bound is that the multiplicative bound is much less conservative for small values of p . Unfortunately, this bound again requires an *a priori* estimate of p . Similar to the strategy of Algorithm 5.2, it is proposed to first make a rough estimate \hat{p}_0 of p using the multiplicative Chernoff bound, and subsequently use the binomial bound. This procedure is formalized in Algorithm 5.3.

Algorithm 5.3 Sequential estimation for relative accuracy

Given a desired $\epsilon_r, \delta \in (0, 1)$, a performance measure ρ , a threshold $\gamma \geq 0$, and the true PDF f_Q for Q , this randomized algorithm returns with a probability of at least $1 - \delta$ an estimate \hat{p}_N for p , such that $p - \hat{p}_N \leq p\epsilon_r$.

1. Draw $N_{\text{SS},1} = \frac{2}{\hat{p}_0\epsilon_{r,1}^2} \ln \frac{1}{\delta_1}$ iid samples \mathbf{q}_j , where $\delta_1 = \delta/\kappa$, $\epsilon_{r,1} = \kappa\epsilon_r$ with the factor κ and the *a priori* estimate \hat{p}_0 suitably chosen.

2. Return the empirical probability $\hat{p}_{N_{\text{SS},1}} = \frac{1}{N_{\text{SS},1}} \sum_{j=1}^{N_{\text{SS},1}} J(\mathbf{q}_j)$.

The real p is less than or equal to $\hat{p}_{\text{bin},1} = \hat{p}_{N_{\text{SS},1}}(1 - \epsilon_{r,1})$ with a confidence level of $1 - \delta_1$.

3. Determine the sample complexity $N_{\text{bin},2}$ that is associated with $\hat{p}_{\text{bin},1}$, $\delta_2 = \delta - \delta_1$, and $\epsilon_{r,2} = \epsilon_r$, using (5.18) evaluated at

$$z^- = \frac{(\hat{p}_{\text{bin},1} - \hat{p}_{\text{bin},1}\epsilon_{r,2})N_{\text{bin},2} - \hat{p}_{\text{bin},1}N_{\text{bin},2}}{\sqrt{N_{\text{bin},2}\hat{p}_{\text{bin},1}(1 - \hat{p}_{\text{bin},1})}}.$$

4. IF $N_{\text{bin},2} \leq N_{\text{SS},1}$
 Return the empirical probability $\hat{p}_N = \hat{p}_{N_{\text{SS},1}}$.
 ELSE
 Draw $N_{\text{SS},2} = N_{\text{bin},2} - N_{\text{SS},1}$ new samples \mathbf{q}_j .
 Return the empirical probability

$$\hat{p}_N = \frac{1}{N_{\text{SS},1} + N_{\text{SS},2}} \sum_{j=1}^{N_{\text{SS},1} + N_{\text{SS},2}} J(\mathbf{q}_j)$$

with a relative accuracy ϵ_r and confidence level $(1 - \delta_1)(1 - \delta_2) \geq 1 - \delta$.

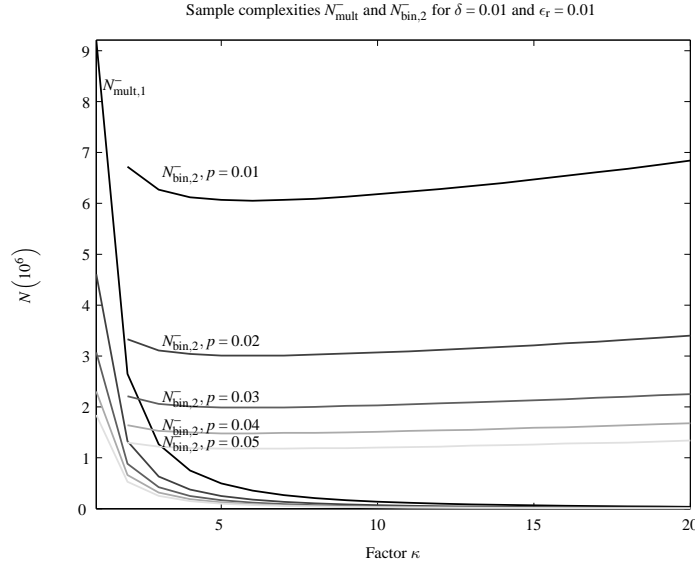


Figure 5.9: Sample complexities $N_{\text{mult},1}^-$ and corresponding $N_{\text{bin},2}^-$ as a function of varying κ , p , δ , and ϵ_r .

Although Algorithm 5.3 and Algorithm 5.2 are based on the same strategy, they differ in the following way, as illustrated by Figure 5.9:

- It can be seen that for typical values of p , ϵ_r , and δ (between 10^{-6} and 10^{-3}), p_{bin} has a minimum at around $\kappa = 6$. This means that the initial sample size $N_{\text{mult},1}^-$ should be chosen at $\kappa = 6$, and \hat{p}_0 at the lowest value considered possible. In case the empirical estimate $\hat{p}_{N_{\text{ss},1}}$ proves to be larger than \hat{p}_0 , the initial sample size $N_{\text{mult},1}^-$ might be too small to provide the required accuracy and confidence in the first simulation sequence.
- Still, with a worst-case estimate for \hat{p}_0 , the sample size for the initial sequence $N_{\text{mult},1}^-$ is relatively small compared to that of the second sequence $N_{\text{bin},2}^-$. This means that the second sequence can benefit from the significantly lower bound.
- In Figure 5.9 the sample complexity $N_{\text{bin},2}^-$ in the second sequence increases with decreasing p , as opposed to Algorithm 5.2. This is caused by the fact that the absolute accuracy ϵ also decreases for a fixed value of ϵ_r , since $\epsilon = p\epsilon_r$.
- Since $N_{\text{bin},2}^-$ increases with decreasing p , the worst-case estimate for p is less than or equal to $\hat{p}_{\text{bin},1} = \hat{p}_{N_{\text{ss},1}}(1 - \epsilon_r)$. Note the difference with Algorithm 5.2 where the real probability p is less than or equal to $\hat{p}_{\text{bin},1} = \hat{p}_{N_{\text{ss},1}} + \epsilon_1$.

Example 5.4 Sequential estimation for relative accuracy. As an example, for values around $\delta = 0.01$, $\epsilon_r = 0.01$, and $p = 0.01$, we can choose $\kappa = 6$. This will result in a sample complexity of 355 385 for the first sequence plus $6 \cdot 10^6$ for the second sequence. Note that the efficiency improvement is only a factor 2/3 compared to the one-sided

multiplicative bound $N_{\text{mult}}^- = 9 \cdot 10^6$ that holds for simple sampling (*i.e.*, $\kappa = 1$). This is caused by the fact that the multiplicative Chernoff bound is already less conservative than the additive bound. However, compared to the additive bound with $N_{\text{Ch}}^- = 2.30 \cdot 10^8$ for $\epsilon = p\epsilon_r = 0.0001$ and $\delta = 0.01$, Algorithm 5.3 means roughly an increase in efficiency of a factor 35!

5.3.4 Importance sampling

The previous section has shown that a significant reduction of N can be achieved using *a priori* information on \mathcal{Q} , although the samples \mathbf{q}_j themselves are not chosen more efficiently. For the one-dimensional ACC control problem the cleared and non-cleared subsets are often separated quite clearly by boundary curves. However, for multi-dimensional problems, the correlation between the individual elements q_i of the parameter vector \mathbf{q} becomes relevant. For example, it can be stated that a dangerous event is more *likely* with a lower value for the acceleration, but the actual occurrence of such an event also depends on other parameters, such as initial distance and relative velocity.

It therefore makes sense to give more attention to operating conditions that are more likely to become dangerous than others. A possibility is to use *importance sampling*, which is a technique to increase the number of occurrences of the event of which the probability p should be estimated [156]. Suppose that, given the parameter set \mathcal{Q} , we want to estimate the probability p , which is defined as:

$$p = \int_{\mathcal{Q}} J(\mathbf{q}) f_{\mathcal{Q}}(\mathbf{q}) d\mathbf{q} = \mathbb{E} \{ J(\mathbf{q}) \}, \quad (5.22)$$

where we sample \mathbf{q} from its joint PDF $f_{\mathcal{Q}}$, denoted as $\mathbf{q} \sim f_{\mathcal{Q}}$. In order to highlight the interesting subset \mathcal{Q}^{\bullet} it thus makes sense not to sample from the original PDF $f_{\mathcal{Q}}$, but instead use an artificial PDF that reflects the ‘importance’ of the events, and then reweigh the observations to get an unbiased estimate. We therefore define an importance sampling PDF ξ that is strictly positive on \mathcal{Q} . We can then rewrite (5.22) as

$$p = \int_{\mathcal{Q}} \frac{J(\mathbf{q}) f_{\mathcal{Q}}(\mathbf{q})}{\xi_{\mathcal{Q}}(\mathbf{q})} \xi_{\mathcal{Q}}(\mathbf{q}) d\mathbf{q} = \mathbb{E} \left\{ \frac{J(\boldsymbol{\varphi}) f_{\mathcal{Q}}(\boldsymbol{\varphi})}{\xi_{\mathcal{Q}}(\boldsymbol{\varphi})} \right\}, \quad (5.23)$$

where $\boldsymbol{\varphi} \sim \xi_{\mathcal{Q}}$. The importance sampling estimator based on $\xi_{\mathcal{Q}}$ is

$$\hat{p}_{N_{\text{IS}}} = \frac{1}{N_{\text{IS}}} \sum_{j=1}^{N_{\text{IS}}} \frac{J(\boldsymbol{\varphi}_j) f_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}{\xi_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}, \quad (5.24)$$

where $\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_{N_{\text{IS}}}$ are N_{IS} iid samples. Every sample $\boldsymbol{\varphi}_j$ drawn from $\xi_{\mathcal{Q}}$ can be seen as a realization of $\frac{\xi_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}{f_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}$ original samples $\mathbf{q}_j \sim f_{\mathcal{Q}}$, which must therefore be reweighed with the factor $\frac{f_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}{\xi_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}$. The expected value of $\hat{p}_{N_{\text{IS}}}$ is p , so we again have an unbiased estimator. The

variance σ_{IS}^2 of the importance sampling estimator is

$$\begin{aligned} \text{var} \{ \hat{p}_{N_{\text{IS}}} \} &= \text{var} \left\{ \frac{1}{N} \sum_{j=1}^N J(\boldsymbol{\varphi}_j) \frac{f_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}{\xi_{\mathcal{Q}}(\boldsymbol{\varphi}_j)} \right\} = \\ &= \frac{1}{N} \left(\text{E} \left\{ J^2(\boldsymbol{\varphi}_j) \frac{f_{\mathcal{Q}}^2(\boldsymbol{\varphi}_j)}{\xi_{\mathcal{Q}}^2(\boldsymbol{\varphi}_j)} \right\} - \left(\text{E} \left\{ J(\boldsymbol{\varphi}_j) \frac{f_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}{\xi_{\mathcal{Q}}(\boldsymbol{\varphi}_j)} \right\} \right)^2 \right). \end{aligned} \quad (5.25)$$

A randomized algorithm can then be formulated as follows.

Algorithm 5.4 Importance sampling

Given a desired $\epsilon, \delta \in (0, 1)$, a performance measure ρ , a threshold $\gamma \geq 0$, and the true PDF $f_{\mathcal{Q}}$ for \mathcal{Q} , this randomized algorithm returns with a probability of at least $1 - \delta$ an estimate $\hat{p}_{N_{\text{IS}}}$ for p , such that $p - \hat{p}_{N_{\text{IS}}} \leq \epsilon$.

1. INITIALIZATION: $j = 0$.
2. Determine an importance sampling PDF $\xi_{\mathcal{Q}}$ that is strictly positive on \mathcal{Q} .
3. Set $j \leftarrow j + 1$.
4. Draw a sample $\boldsymbol{\varphi}_j$ according to $\xi_{\mathcal{Q}}$.
5. Calculate the cumulative empirical probability

$$\hat{p}_{N_j} = \frac{1}{N_j} \sum_{j=1}^{N_j} \frac{J(\boldsymbol{\varphi}_j) f_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}{\xi_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}.$$

6. IF convergence is reached
Return the empirical probability $\hat{p}_{N_{\text{IS}}} = \hat{p}_{N_j}$.
- ELSE
GOTO Step 3.

Example 5.5 Importance sampling. Suppose that we again want to estimate the probability p of a dangerous event, as specified by Example 5.2. The goal is then to estimate

$$p = \int_{\mathcal{Q}} J(a_1) f(a_1) da_1 = \text{E} \{ J(a_1) \}, \quad (5.26)$$

where $a_1 \sim \mathcal{N}(0, 1.5)$. Assuming that this Gaussian PDF is correct, then with a simple sampling method relatively few samples will lie in the interval of interest $[-10, -2.69]$, as was observed in Example 5.2.

We therefore define a more suitable $\xi_{\mathcal{Q}}$ to sample from. We choose the linear PDF $\xi_{\mathcal{Q}}(a_1) = -0.005a_1 + 0.05$ with a_1 bounded on the interval $[-10, 10]$, as shown in Figure 5.10. Note that $\int_{-10}^{10} \xi_{\mathcal{Q}}(a_1) = \int_{-10}^{10} f_{\mathcal{Q}}(a_1) = 1$, such that the reweighting process gives an unbiased estimator. Again we use $N = 23026$ for a fair comparison to the simple sampling process in Example 5.2.

With Algorithm 5.4 more ‘important’ samples will be generated for every ℓ -th simulation set, thus decreasing the variance of \hat{p}_N . This result can be seen from the histogram

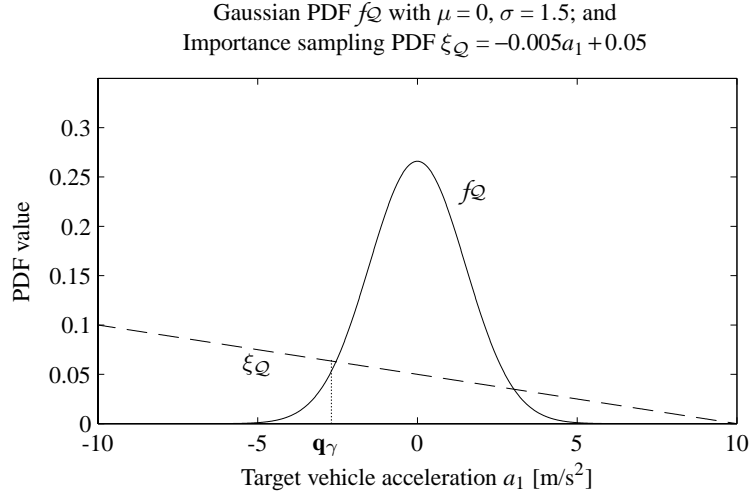


Figure 5.10: Probability density functions used in the examples: original Gaussian PDF $f_{\mathcal{Q}}$ and importance sampling PDF $\xi_{\mathcal{Q}}$. The boundary value $\mathbf{q}_{\gamma} = -2.69 \text{ m/s}^2$ is also indicated.

in Figure 5.11, where the empirical mean $\hat{p}_M = 0.03633$ and the variance $\sigma_{\text{IS}}^2 = 6.4 \cdot 10^{-7}$, based on $M \cdot N = 2.3 \cdot 10^8$ simulations. Note that \hat{p}_M is approximately equal to that in Example 5.2. However, the variance of a particular realization \hat{p}_{N_ℓ} has decreased by a factor of 2.4. Since the variance increases inversely proportional to N , as shown in (5.6), the sample complexity could have been decreased by the same factor.

Although the reduction in this simple example is quite limited, it has been shown by Gerlach [69] that the number of sufficient Monte Carlo simulations can be reduced by several orders of magnitude, especially with multi-dimensional distributions. In order to effectively reduce the variance, $\xi_{\mathcal{Q}}$ must be chosen proportional to $|J(\mathbf{q})f_{\mathcal{Q}}(\mathbf{q})|$. Based on (5.25), Gerlach shows that even a single sample is enough if $\xi_{\mathcal{Q}}$ is chosen as

$$\xi_{\mathcal{Q}} = \frac{\int_{\mathcal{Q}} J(\mathbf{q})f_{\mathcal{Q}}(\mathbf{q})d\mathbf{q}}{p}. \quad (5.27)$$

However, this is only a hypothetical possibility, since (5.27) requires *a priori* knowledge of p , as well as of the outcome of $J(\mathbf{q}_j)$ for every $\mathbf{q}_j \in \mathcal{Q}$. Moreover, a risk with importance sampling is that a poor choice of $\xi_{\mathcal{Q}}$ may result in a significant *increase* of the sample complexity.

Therefore, the performance of the importance sampling method heavily depends on the reliability of the PDF $\xi_{\mathcal{Q}}$ to generate random variables. Furthermore, the sample complexity cannot be determined *a priori*, thus requiring the iterative loop in Algorithm 5.4 and the need for a suitable stopping criterion in Step 6 that determines convergence of the algorithm. In order to make this algorithm work, good choices for the sample complexity N_{IS} and the importance sampling PDF $\xi_{\mathcal{Q}}$ must be made in advance, as well as an appropriate method to sample from this distribution. We will address these issues next.

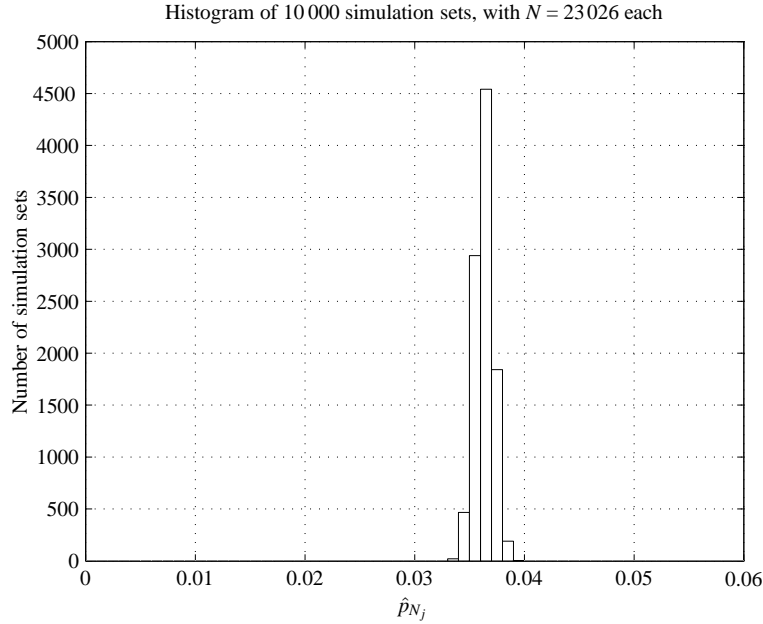


Figure 5.11: Histogram for the estimate \hat{p}_{N_ℓ} of 10 000 simulation sets, with $N = 23026$ each, where the acceleration profile is sampled from an importance sampling distribution $\xi_{\mathcal{Q}}(a_1) = -0.005a_1 + 0.05$.

5.3.5 Sample complexity of importance sampling

Existing literature relies on stopping criteria or thumb rules for Algorithm 5.4 [29, 38]. In this way it cannot be predicted when the simulation should be stopped, other than by using the initial (conservative) bound. Instead of using a stopping criterion, we would like to know the sufficient sample complexity to achieve a specified δ and ϵ in advance. Unfortunately, it is not possible to exactly calculate the minimum sample complexity N_{IS} beforehand [29]. However, here we will present a method that at least gives a reliable prediction of N_{IS} after an initial sequence with a small number of samples.

In order to provide an estimate of the reduction in sample complexity that can be achieved with importance sampling, we would like to know the *importance sampling reduction factor* [223], which indicates the reduction in the sample complexity that is sufficient to achieve the same level of accuracy and confidence as the Chernoff bound. It is defined as

$$\lambda_{\text{IS}} = \frac{\sigma_{\text{IS}}^2}{\sigma_{\text{SS}}^2}, \quad (5.28)$$

where σ_{SS}^2 and σ_{IS}^2 are the variances of the simple sampling estimator and the importance sampling estimator, respectively. Similar to (5.7), the accuracy of the resulting estimator $\hat{p}_{N_{\text{IS}}}$ can then be expressed in its relative RMS error

$$\sqrt{\frac{\text{var}\{\hat{p}_{N_{\text{IS}}}\}}{p^2}} = \sqrt{\frac{\lambda_{\text{IS}} \text{var}\{\hat{p}_N\}}{p^2}} = \sqrt{\frac{\lambda_{\text{IS}}(1-p)}{pN_{\text{IS}}}}. \quad (5.29)$$

The gain in efficiency can then be calculated by comparing (5.7) and (5.29). The same level of relative error can be achieved with the following reduction in samples for importance sampling:

$$N_{\text{IS}} = \lambda_{\text{IS}} N_{\text{SS}}. \quad (5.30)$$

In order to estimate N_{IS} , we therefore need to know λ_{IS} , and thus the variances σ_{SS}^2 and σ_{IS}^2 . The simple sampling variance σ_{SS}^2 can be approximated from the empirical estimate of the first sequence:

$$\hat{\sigma}_{\text{SS}}^2 = \frac{\frac{1}{N_{\text{SS},1}} \sum_{j=1}^{N_{\text{SS},1}} J(\mathbf{q}_j) - \left(\frac{1}{N_{\text{SS},1}} \sum_{j=1}^{N_{\text{SS},1}} J(\mathbf{q}_j) \right)^2}{N_{\text{SS},1}}. \quad (5.31)$$

The importance sampling variance σ_{IS}^2 can be approximated by the empirical estimate of (5.25), which is

$$\hat{\sigma}_{\text{IS}}^2 = \frac{\frac{1}{N_{\text{IS}}} \sum_{j=1}^{N_{\text{IS}}} J^2(\boldsymbol{\varphi}_j) \frac{f_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}{\xi_{\mathcal{Q}}^2(\boldsymbol{\varphi}_j)} - \left(\frac{1}{N_{\text{IS}}} \sum_{j=1}^{N_{\text{IS}}} J(\boldsymbol{\varphi}_j) \frac{f_{\mathcal{Q}}(\boldsymbol{\varphi}_j)}{\xi_{\mathcal{Q}}(\boldsymbol{\varphi}_j)} \right)^2}{N_{\text{IS}}}. \quad (5.32)$$

Unfortunately, (5.32) can only be calculated *a posteriori*, whereas we would like to know N_{IS} before we start with importance sampling. We therefore seek to estimate $\hat{\sigma}_{\text{IS}}^2$ *a priori*, thereby predicting the importance sampling reduction factor in (5.28), which in turn is used to predict $N_{\text{IS}} = \lambda_{\text{IS}} N_{\text{SS}}$.

We therefore use the limited set of samples that has already been evaluated in the first sequence. Consider the factor $\frac{f_{\mathcal{Q}}(\mathbf{q}_j)}{\xi_{\mathcal{Q}}(\mathbf{q}_j)}$, which is known for every sampled value \mathbf{q}_j in the first sequence. We then predict the number of ‘positives’ ($J(\mathbf{q}) = 1$) that would have corresponded to the use of the importance sampling PDF $\xi_{\mathcal{Q}}$ in the first sequence. We make this prediction by using the assumption that every single sample obtained with simple sampling ($\mathbf{q}_j \sim f_{\mathcal{Q}}$), corresponds to $\frac{\xi_{\mathcal{Q}}(\mathbf{q}_j)}{f_{\mathcal{Q}}(\mathbf{q}_j)}$ samples using importance sampling ($\mathbf{q}_j \sim \xi_{\mathcal{Q}}$), as shown in (5.24).

The expected importance sampling variance $\hat{\sigma}_{\text{IS}}^2$ can then be estimated using the first simple sampling sequence, where the first term in the numerator of (5.32) is multiplied with $\frac{\xi_{\mathcal{Q}}(\mathbf{q}_j)}{f_{\mathcal{Q}}(\mathbf{q}_j)}$. The second term is equal to $\hat{p}_{N_{\text{IS}}}^2$ and can be approximated by its simple sampling estimate. Rewriting the samples $\boldsymbol{\varphi}_j$ to \mathbf{q}_j we get

$$\hat{\sigma}_{\text{IS}}^2 = \frac{\frac{1}{N_{\text{SS},1}} \sum_{j=1}^{N_{\text{SS},1}} J^2(\mathbf{q}_j) \frac{f_{\mathcal{Q}}(\mathbf{q}_j)}{\xi_{\mathcal{Q}}(\mathbf{q}_j)} - \left(\frac{1}{N_{\text{SS},1}} \sum_{j=1}^{N_{\text{SS},1}} J(\mathbf{q}_j) \right)^2}{N_{\text{SS},1}}. \quad (5.33)$$

We can then estimate λ_{IS} from (5.28) by substitution of (5.31) and (5.33). After the first sequence we can then provide a prediction for N_{IS} . Moreover, using the results of this first sequence, we are also able to form a suitable importance sampling PDF $\xi_{\mathcal{Q}}$, as discussed next.

5.3.6 Kernel density estimation of the importance sampling PDF

As shown in Chapter 2, in practice the components q_i of \mathbf{q} are interdependent, such that the distribution of \mathbf{q} is governed by a *joint* PDF $f_{\mathcal{Q}} = f(q_1, q_2, \dots, q_n)$. The performance of Algorithm 5.4 heavily depends on the reliability of the PDF $\xi_{\mathcal{Q}}$ to generate random variables, and of the models used in the simulation. An efficient estimator of $\hat{p}_{N_{\text{is}}}$ is obtained by choosing $\xi_{\mathcal{Q}}$ proportional to the importance of the individual samples, with importance defined as $|J(\mathbf{q})f_{\mathcal{Q}}(\mathbf{q})|$ (for Boolean performance measures) or $|\rho_k(\mathbf{q})f_{\mathcal{Q}}(\mathbf{q})|$ (for continuous performance measures). A rare but dangerous event can thus be equally important as a frequent but less critical event. Conventional importance sampling methods consist of shifting the mean or variance of the original PDF $f_{\mathcal{Q}}$ to form the importance sampling PDF $\xi_{\mathcal{Q}}$, a so-called *parametric* approach [29, 223]. However, the optimal importance sampling PDF $\xi_{\mathcal{Q}}$ will most likely not be a standard type PDF (e.g., Gaussian), but reflect an irregular multi-dimensional surface in \mathcal{Q} . Parametric importance sampling methods can therefore bias the results if not carefully chosen. Instead, a *nonparametric* approach will be used, where the *entire* PDF is estimated. Since importance is related to the samples \mathbf{q}_j for which $J(\mathbf{q}_j) = 1$, we apply a multivariate kernel density estimation on these samples.

Kernel density estimation

A kernel density estimator can be regarded as a generalization of the well-known histogram \hat{f}_h for a one-dimensional parameter q , which can be formalized as:

$$\hat{f}_h(q) = \frac{1}{Nh} \sum_{j=1}^N \kappa_u\left(\frac{q-q_j}{h}\right), \quad (5.34)$$

where h is the histogram binwidth and $\kappa_u(u)$ is a uniform kernel function

$$\kappa_u(u) = \begin{cases} 1 & \text{if } |u| \leq \frac{1}{2} \\ 0 & \text{else} \end{cases}. \quad (5.35)$$

Figure 5.12 illustrates the construction of the histogram based on a set of ‘positive’ samples q^\bullet . The disadvantage of using a histogram for density estimation is the discontinuous form and the possibility of ‘gaps’ in the resulting PDF. For example, the value for $\hat{f}_h(q)$ on the interval $[-4.75, -4.5]$ is zero, whereas the adjacent intervals have non-zero values.

In order to ‘smoothen’ the histogram, a continuous instead of a discontinuous kernel function κ should be used. The kernel density estimator $\hat{\xi}$ based on the samples q_j, \dots, q_N is then defined as

$$\hat{\xi}_h(\mathbf{q}) = \frac{1}{N} \sum_{j=1}^N \kappa_h(\mathbf{q} - \mathbf{q}_j), \quad (5.36)$$

where

$$\kappa_h(u) = \frac{1}{h} \kappa(u/h), \quad (5.37)$$

with $\kappa(u)$ some kernel function and h the kernel bandwidth.

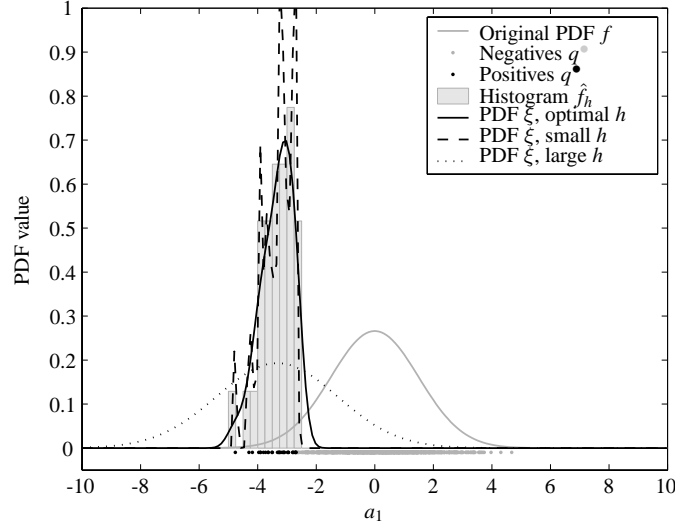


Figure 5.12: Examples of kernel density estimation for the purpose of construction of an importance sampling PDF, based on different values for the bandwidth.

Kernel selection

Hardle *et al.* [92] show that kernel density estimates based on any two different kernel functions $\kappa(u)$ can achieve the same degree of smoothness (in the limit) by adjusting one of the bandwidths h by multiplication with a constant factor. That means that, regardless of the *type* of kernel function, the same degree of smoothness can be reached with any kernel function, only by tuning the bandwidth. The differences between different kernel functions are shown to be at most a few percent: *e.g.*, the Gaussian kernel is 4 % less efficient than the most efficient kernel in terms of the number of samples that is necessary to obtain a representative PDF. Therefore, it is concluded that for practical purposes the choice of kernel function is almost irrelevant for the efficiency of the estimated PDF ξ , and we will use the simple Gaussian kernel function

$$\kappa(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}u^2\right). \quad (5.38)$$

Efficient kernel bandwidth selection

More important than the type of kernel, is to choose the most appropriate kernel bandwidth h . If the bandwidth is too small, the resulting kernel density estimate may be ‘undersmoothed’. However, if the bandwidth is too large, the resulting PDF will be ‘oversmoothed’ and not distinguish enough between important and unimportant subsets in \mathcal{Q} . Figure 5.12 illustrates this issue for different values of the bandwidth.

The selection of the bandwidth for the multivariate case becomes more critical with higher dimension. For the multivariate case, the goal of density estimation is to approximate the shape of a multivariate PDF. In order to account for the interdependency of the components q_i of \mathbf{q} , a multivariate kernel should be used. A general approach is to use a

nonsingular diagonal bandwidth matrix \mathbf{H} . The general form for the multivariate density estimator is then

$$\hat{\xi}_{\mathbf{H}}(\mathbf{q}) = \frac{1}{N} \sum_{j=1}^N \frac{1}{\det(\mathbf{H})} \kappa \left\{ \mathbf{H}^{-1}(\mathbf{q} - \mathbf{q}_j) \right\} = \frac{1}{N} \sum_{j=1}^N \kappa_{\mathbf{H}}(\mathbf{q} - \mathbf{q}_j), \quad (5.39)$$

where

$$\kappa_{\mathbf{H}}(u) = \frac{1}{\det(\mathbf{H})} \kappa(\mathbf{H}^{-1}(u)), \quad (5.40)$$

analogously to κ_h in the one-dimensional case. An equal bandwidth h in all dimensions corresponds to $\mathbf{H} = h\mathbf{I}_n$ where \mathbf{I}_n denotes the $n \times n$ identity matrix. Different bandwidths are equivalent to $\mathbf{H} = \text{diag}(h_1, \dots, h_n)$, a diagonal matrix with elements h_1, \dots, h_n .

A criterion for choosing the optimal bandwidth matrix \mathbf{H} is the mean integrated squared error (MISE) expressed as [43]

$$\text{MISE}(\hat{\xi}_{\mathbf{H}}) = \int \mathbb{E} \left\{ \hat{\xi}_{\mathbf{H}}(\mathbf{q}) - f(\mathbf{q}) \right\}^2 d\mathbf{q}. \quad (5.41)$$

In general two methods are available that provide a suitable bandwidth to be used for the kernel density estimation: plug-in methods and cross-validation methods. For details on the working principle of these methods the reader is referred to [92]. The plug-in method is the most widely used bandwidth selector, and results in the so-called ‘normal reference rule’. When data are observed from the multivariate density with dimension n and the diagonal bandwidth matrix, denoted by $\mathbf{H} = \text{diag}(h_1, h_2, \dots, h_d)$, is employed, the optimal bandwidth that minimizes the mean integrated squared error can be approximated by [23, 207]

$$h_i = \sigma_i \left\{ \frac{4}{(n+2)N^\bullet} \right\}^{\frac{1}{n+4}} \quad (5.42)$$

for $i = 1, 2, \dots, n$, where σ_i is the standard deviation of the i -th parameter variable that can be replaced by its sample estimator in practical implementations. The number of positive samples \mathbf{q}^\bullet is represented by N^\bullet .

Although (5.42) is often used in practice, it is only valid for Gaussian distributed data. In our case, the samples are not assumed to be Gaussian distributed, hence (5.42) will not yield the optimal bandwidth, but will cause the PDF to be oversmoothed rather than under-smoothed. Undersmoothing (using a small bandwidth h) increases the risk of an incorrect representation of the importance sampling PDF $\xi_{\mathcal{Q}}$, whereas oversmoothing (using a large bandwidth h) only decreases the efficiency of the importance sampling algorithm. To be on the safe side, a larger bandwidth is desired, and therefore oversmoothing is not considered problematic, such that we will use the optimal bandwidth given by (5.42).

5.3.7 Random number generation

To operate a randomized algorithm, samples must be generated from a PDF using a random number generator. Many methods for random number generation are available, such as direct, inversion, and rejection methods [236]. Since direct or inversion algorithms are unsuitable for nonparametric PDFs, we will use the rejection method to produce samples from the importance sampling PDF that is constructed by kernel density estimation. The numbers

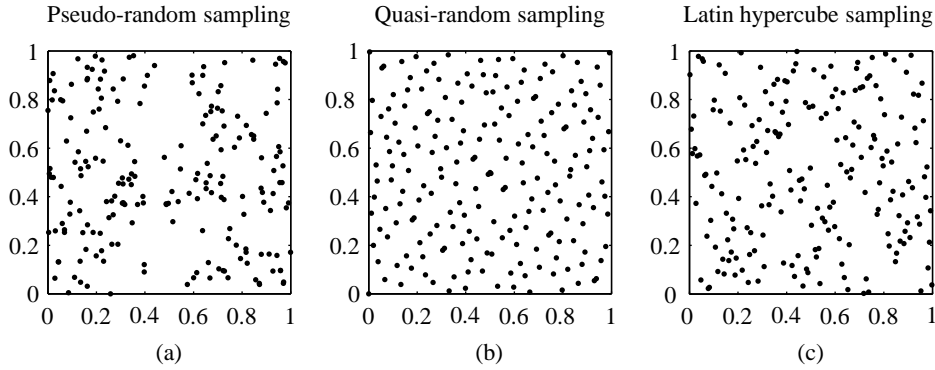


Figure 5.13: Methods for random number generation for a two-dimensional example.

that these algorithms produce are never truly random, since they are completely determined by a relatively small set of initial values, hence the more appropriate term *pseudo*-random number generation. Figure 5.13(a) shows a plot with pseudo-random samples, which clearly do not represent a ‘truly’ random behavior, due to the uneven spread of samples.

Alternatively, *quasi*-random numbers can be used, which are used to generate representative samples from the PDF that allow a faster convergence in a quasi-Monte Carlo simulation. These algorithms are based on deterministic series of samples that are selected using an optimality criterion, such that the resulting set is ‘evenly distributed’ in the parameter domain. An example of the algorithm by Faure (see [236]) is shown in Figure 5.13(b).

Unfortunately, for higher dimensions of \mathcal{Q} , quasi-Monte Carlo may become inferior to the simple sampling Monte Carlo with pseudo-random numbers. A solution is to use the Latin hypercube sampling method developed by Iman *et al.* [104], which combines a selection of pseudo-random random numbers and an evenly spread representation of the parameter set. An example is depicted in Figure 5.13(c). Based on this combination of characteristics, the Latin hypercube algorithm is selected for implementation of the sampling steps in the presented algorithms (*e.g.*, Steps 1 and 4 in Algorithm 5.3 or Step 4 in Algorithm 5.4). Various other efficient sampling algorithms (*e.g.*, the hit-and-run method) are available that enable to approximate the volume of a set in polynomial time [236]. However, these algorithms usually only apply to convex sets or require similar specific conditions, whereas the parameter set \mathcal{Q} will most likely be non-convex in practice.

5.3.8 Summary of methods for sample complexity reduction

As we have seen so far in Section 5.3, a reduction of the sample complexity can be achieved on several levels in the validation process:

- Reduction of the parameter set \mathcal{Q} by using *a priori* knowledge of the infeasible subset $\mathcal{Q}_{\text{infeas}}$, the cleared subset \mathcal{Q}_0^* , and the non-cleared subset \mathcal{Q}_0^\bullet .
- Use of the binomial bound through a sequential sampling procedure that gives a sufficient sample complexity, which is lower than the conservative Chernoff bound.
- Reformulation of the desired accuracy to a relative instead of an absolute measure.

- Use of kernel density estimation to construct an importance sampling PDF that emphasizes the relevant samples in Q .
- Improvement of the rate of convergence of the estimation by using a Latin hypercube for the sample generation.

Depending on the problem definition, the application, and the test tools that are used, one or more of these methods can be combined. Let us first use the sequential estimation algorithm to extend the importance sampling method, as discussed in the next section.

5.4 Adaptive importance sampling

In this section a new validation approach is presented based on an sequential randomized algorithm that uses *adaptive importance sampling* (AIS), where the importance sampling PDF is based on the results of the previous sequence.

5.4.1 A new algorithm for probabilistic validation

From the results of the *first* sequence with sample complexity $N_{ss,1}$ an importance sampling PDF $\xi_{Q,2}$ can be constructed using the approach of kernel density estimation. In addition, the importance sampling reduction factor λ_{IS} can be estimated by the method shown in Section 5.3.5. The remaining number of samples to be taken in the *second* sequence is then $N_{IS,2} = \lambda_{IS} (N_{bin,2} - N_{ss,1})$. If desired, these steps can be repeated to obtain a better importance sampling PDF ξ_Q . We therefore combine Algorithms 5.2 and 5.4 as follows.

Example 5.6 Sequential estimation with AIS. To illustrate the efficiency of Algorithm 5.5, the one-dimensional problem of Example 5.2 is again repeated. Figure 5.14 shows the results for 10 000 simulation sets, each with $N_\ell \in [2519, 2810]$, where in the first simple sampling sequence $N_{ss,1,\ell} = 2391$ and in the second importance sampling sequence $N_{IS,2,\ell} \in [128, 419]$. The sample complexity $N_{IS,2}$ in the second sequence is considerably smaller, since the use of importance sampling reduces $N_{IS,2}$ in the second sequence. The results are $\hat{p}_N = 0.03628$ and $\hat{\sigma}_{IS}^2 = 1.2140 \cdot 10^{-5}$. When the empirical confidence parameter is set to $\hat{\delta} = 0.01$ the empirical accuracy is found to be $\hat{\epsilon} = 0.0081$, meaning that $N_\ell \in [2519, 2810]$ is still slightly conservative, although much better than in Example 5.2.

The fact that the bound $N_{bin,2,\ell}$ is not strict (such that $\hat{\epsilon} = 0.01$) is caused by the fact that the factor κ is not chosen optimal, as discussed in Section 5.3.2. Nevertheless, there is an efficiency improvement of at least $\frac{N_{ss}}{\max_\ell(N_{bin,2,\ell})} = 8.2$ for this particular example.

This means roughly a twofold improvement with respect to Algorithm 5.2, where the improvement was a factor of 4.8. Obviously the resulting variance is slightly larger than in Example 5.3, but the accuracy and confidence are still within the desired values. In other words, the bound on the sample complexity is more strict (less conservative for given values of δ and ϵ).

Algorithm 5.5 Sequential estimation with AIS

Given a desired $\epsilon, \delta \in (0, 1)$, a performance measure ρ , a threshold $\gamma \geq 0$, and the true PDF f_Q for Q , this randomized algorithm returns with a probability of at least $1 - \delta$ an estimate $\hat{p}_{N_{\text{AIS}}}$ for p , such that $p - \hat{p}_N \leq \epsilon$.

1. Draw $N_{\text{SS},1} = \frac{1}{2\epsilon_1^2} \ln \frac{1}{\delta_1}$ iid samples \mathbf{q}_j , where $\delta_1 = \delta/\kappa$, and $\epsilon_1 = \kappa\epsilon$, and κ is a suitably chosen real number.
2. Return the empirical probability

$$\hat{p}_{N_{\text{SS},1}} = \frac{1}{N_{\text{SS},1}} \sum_{j=1}^{N_{\text{SS},1}} J(\mathbf{q}_j).$$

The real p is always less than or equal to $\hat{p}_{\text{bin},1} = \hat{p}_{N_{\text{SS},1}} + \epsilon_1$ with confidence $1 - \delta_1$.

3. Determine the sample complexity $N_{\text{bin},2}$ that is associated with $\hat{p}_{\text{bin},1}$, $\delta_2 = \delta - \delta_1$, and $\epsilon_2 = \epsilon$, using (5.18) evaluated at

$$z^- = \frac{(\hat{p}_{\text{bin},1} - \epsilon_2)N_{\text{bin},2} - \hat{p}_{\text{bin},1}N_{\text{bin},2}}{\sqrt{N_{\text{bin},2}\hat{p}_{\text{bin},1}(1 - \hat{p}_{\text{bin},1})}}.$$

4. Estimate the importance sampling PDF $\xi_{Q,2}$, based on the kernel density estimate (5.39) of the samples \mathbf{q}_j^* , for which $\rho_k(\mathbf{q}_j) \geq \gamma_k$.
5. Estimate the importance sampling reduction factor $\hat{\lambda}_{\text{IS}} = \frac{\tilde{\sigma}_{\text{IS}}^2}{\tilde{\sigma}_{\text{SS}}^2}$, with $\tilde{\sigma}_{\text{IS}}^2$ from (5.33) and $\tilde{\sigma}_{\text{SS}}^2$ from (5.31).
6. IF $N_{\text{bin},2} \leq N_{\text{SS},1}$
Return the empirical probability $\hat{p}_{N_{\text{AIS}}} = \hat{p}_{N_{\text{SS},1}}$.

ELSE

Draw $N_{\text{IS},2} = \hat{\lambda}_{\text{IS}}(N_{\text{bin},2} - N_{\text{SS},1})$ new samples from the importance sampling PDF $\xi_{Q,2}$.

Return the empirical probability

$$\hat{p}_{N_{\text{AIS}}} = \frac{1}{N_{\text{SS},1} + N_{\text{IS},2}} \left(\sum_{j=1}^{N_{\text{SS},1}} J(\mathbf{q}_j) + \sum_{j=1}^{N_{\text{IS},2}} \frac{J(\boldsymbol{\varphi}_j) f_Q(\boldsymbol{\varphi}_j)}{\xi_{Q,2}(\boldsymbol{\varphi}_j)} \right)$$

with accuracy ϵ and confidence $(1 - \delta_1)(1 - \delta_2) \geq 1 - \delta$.

For higher dimensions of Q , the rate of convergence of Algorithm 5.5 increases compared to Algorithm 5.2. However, the examples in this section were one-dimensional, and the question arises how these methods perform for the multi-dimensional case. These methods include parameter set reduction, kernel density estimation, and sample generation, as will be discussed next.

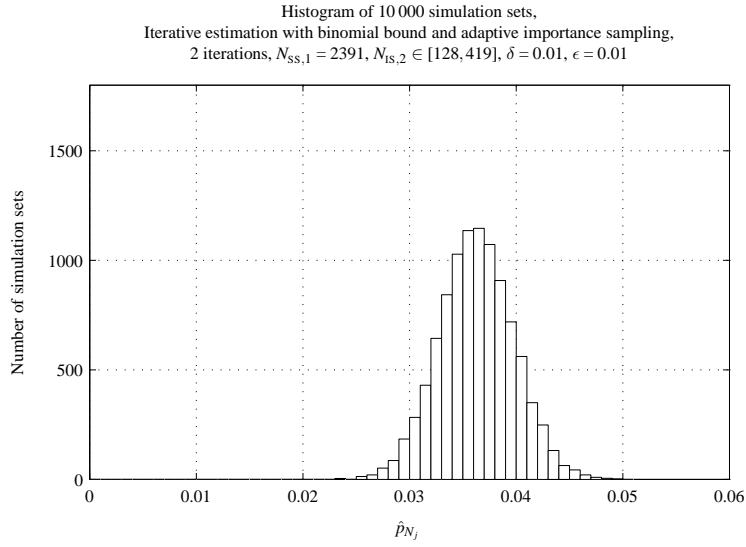


Figure 5.14: Histogram of 10 000 simulation sets, with $N_\ell \in [2519, 2810]$ each, using Algorithm 5.5.

5.4.2 Extension to the multi-dimensional problem

In practice, the controller validation problem is always affected by a multi-dimensional parameter set. The validation objective is to identify the cleared parameter set \mathcal{Q}^* and the non-cleared parameter set \mathcal{Q}^\bullet in a probabilistic sense. Therefore, we check whether the performance measure is satisfied ($\rho_k < \gamma_k$) with a probability p , within a given accuracy ϵ and with a given confidence level $1 - \delta$.

Consider again the ACC control problem. In the previous section the acceleration a_1 was considered the only disturbance. However, in practice the probability p depends also on the initial distance $x_r(0)$ and velocities of both vehicles $v_1(0)$, $v_2(0)$ at the moment of first detection by the sensor. Consider for instance a cut-in situation at close distance x_r , which is more dangerous than a vehicle cutting in at larger distance (with equal $v_1(0)$, $a_1(0)$, and $v_2(0)$). The performance measure ρ thus depends on the parameter vector $\mathbf{q} = [x_r(0) \quad v_1(0) \quad a_1(0) \quad v_2(0)]^T$ that specifies the single-lane scenario, as was defined in Chapter 2. In order to provide a benchmark for the AIS algorithm, let us first give the results for this multi-dimensional problem using the simple sampling algorithm.

Example 5.7 Simple sampling for a multi-dimensional parameter set. Consider the single-lane ACC problem, where a Boolean safety measure ρ_{coll} is used that indicates the occurrence of a collision ($\rho_{\text{coll}} = 1$) or no collision ($\rho_{\text{coll}} = 0$). A scalar threshold with a constant value is used: $\gamma_k = 1$. Suppose that we would like to know the probability that a driver must intervene to prevent a collision.

The ACC model perturbed by the multi-dimensional set \mathcal{Q} is validated using Algorithm 5.1. If the desired parameters for accuracy and confidence are again $\epsilon = 0.01$ and $\delta = 0.01$, the one-sided Chernoff bound (5.16) gives $N = 23\,026$. Since the results cannot be visually displayed for more than two dimensions, Figure 5.15 shows the positives and negatives as a function of the initial distance $x_r(0)$ and initial relative velocity $v_r(0) =$

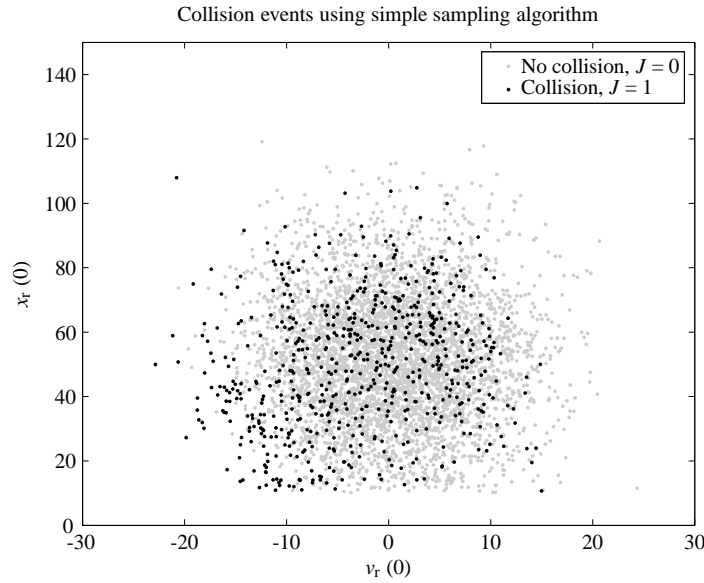


Figure 5.15: Safety measure ρ_{coll} as a function of the initial scenario conditions $x_r(0)$ and $v_r(0) = v_1(0) - v_2(0)$.

$v_1(0) - v_2(0)$. The dependence on the other parameter variables is illustrated by the overlap between the mapping of \mathcal{Q}^* and \mathcal{Q}^\bullet onto the two-dimensional range – range rate diagram.

There is a total of 2717 collision occurrences ($\rho_{\text{coll}} = 1$) out of $N = 23026$. This means that $\hat{p}_N = 0.118$ with $\epsilon = 0.01$ and $\delta = 0.01$. Although a collision becomes more likely with lower values for $x_r(0)$ and $v_r(0)$, as illustrated in Figure 5.15, a distinctive boundary between \mathcal{Q}^* and \mathcal{Q}^\bullet cannot be drawn in the two-dimensional representation, due to the joint effect of absolute vehicle speed and target vehicle acceleration.

Example 5.7 has shown that a huge number of samples is required for a multivariate problem. Although grid-based searches have their disadvantages in terms of sample complexity, a rough grid can already give good *a priori* information in a first sequence. In the *first* sequence with sample complexity $N_{\text{SS},1}$ it is therefore recommended to use a grid-based sampling strategy or Latin hypercube, in order to get a representative overview of the parameter set \mathcal{Q} , characterized by the uniform PDF $\xi_{\mathcal{Q},1}$. The results of this first sequence can then be used to form an importance sampling PDF $\xi_{\mathcal{Q},2}$ using kernel density estimation for the second sequence.

Example 5.8 Adaptive importance sampling for the multivariate case. This example illustrates the use of an initial gridding sequence, where we use a grid consisting of all combinations of the following four parameters: $x_r(0) \in \{10, 50, 90, 130\}$ m, $v_1(0) \in \{10, 20, 30, 40\}$ m/s, $a_1(0) \in \{-3, -2, -1, 0, 1, 2\}$ m/s², and $v_2(0) \in \{10, 20, 30, 40\}$ m/s. This results in a grid size $N_{\text{GS}} = 384$.

For visualization purposes, Figure 5.16 shows the results in the two-dimensional range – range rate diagram. Again the same conclusions can be drawn as for Figure 5.15.

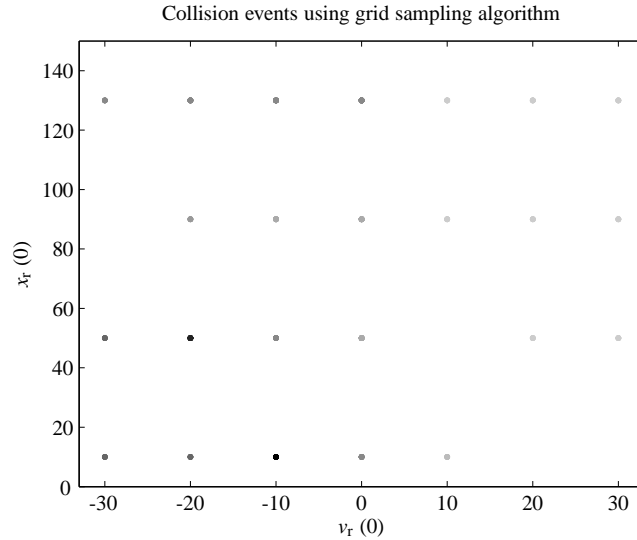


Figure 5.16: Scatter plot with the occurrence of collision events in the parameter set \mathcal{Q} , consisting of a grid with equidistant points. The effect of multivariate parameter combinations is taken into account by varying shades of grey: light gray points indicate that no collision occurs for any \mathbf{q}_j , whereas the dots become darker with increasing collision risk.

But now a more complete picture is obtained for the whole parameter set. It appears that there is structure in the parameter set that can be used to reduce the volume of \mathcal{Q} (for both cleared and non-cleared subsets): The collision probability increases with lower values for $x_r(0)$ and $v_r(0)$. Obviously, certain subsets can be excluded from the evaluation. For instance, subsets with positive relative velocity ($v_r(0) > 0$) and positive acceleration ($a_1(0) > 0$) will *never* result in a collision, so these subsets can be cleared *a priori* as \mathcal{Q}_0^* and can be excluded from the further validation.

Furthermore, the grid search results enable the construction of an importance sampling PDF $\xi_{\mathcal{Q}}$ through multivariate kernel density estimation. Since \mathcal{Q} has dimension 4, it is not possible to visualize the results for the entire volume of \mathcal{Q} . The results in Figure 5.17 therefore depict the resulting importance sampling PDF $\xi_{\mathcal{Q}}$ as a function of $x_r(0)$ and $v_1(0)$ for several operating points of $v_2(0)$ and $a_1(0)$. As can be expected, it can be seen that collisions are more likely for scenarios with a shorter initial distance to the target $x_r(0)$, lower target velocity $v_1(0)$, lower target deceleration $a_1(0)$, and higher host velocity $v_2(0)$.

In the second sequence of the AIS algorithm, samples are generated from the multivariate kernel density estimate $\xi_{\mathcal{Q}}$. The result is that relatively more samples are generated in areas that are deemed *important*, where ‘importance’ is currently defined in terms of the occurrence of a collision. For continuous performance functions, $\xi_{\mathcal{Q}}$ can be weighted according to the value of ρ_k , *i.e.*, proportional to $|\rho_k(\mathbf{q})f_{\mathcal{Q}}(\mathbf{q})|$. Figure 5.18 shows the samples generated from $\xi_{\mathcal{Q}}$. Indeed, the samples lie in areas with shorter distance $x_r(0)$, lower relative velocity $v_r(0)$, and lower decelerations $a_{1,0}$. These type of figures enable to identify the volume of the subsets \mathcal{Q}^* and \mathcal{Q}^\bullet (*e.g.*, characterized by convex hulls) and pinpoint problem areas in \mathcal{Q} .

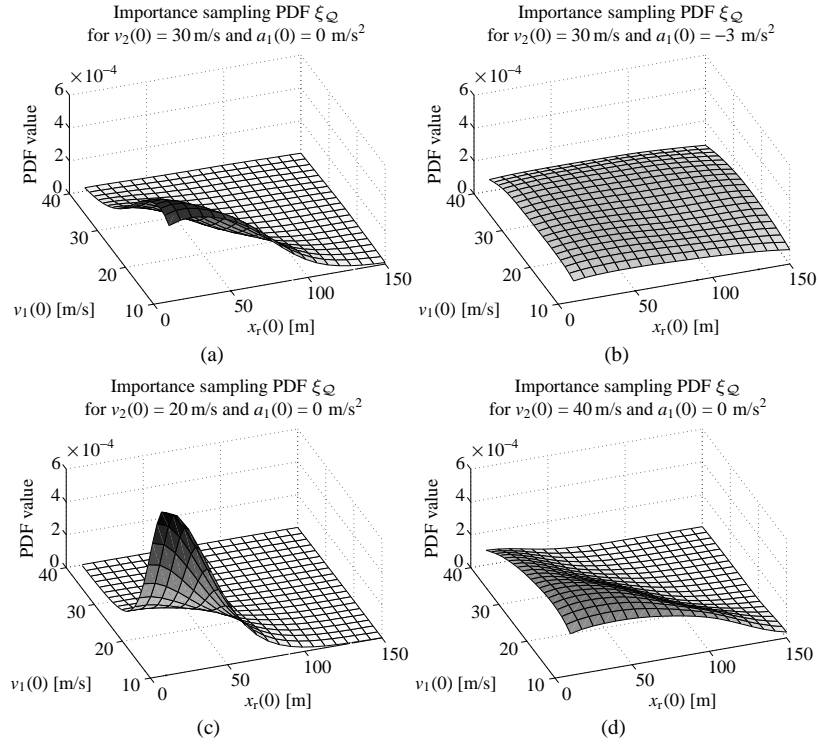


Figure 5.17: Importance sampling PDFs $\xi_{\mathcal{Q}}$ as a function of $x_r(0)$ and $v_1(0)$ for several operating points of $a_1(0)$ and $v_2(0)$.

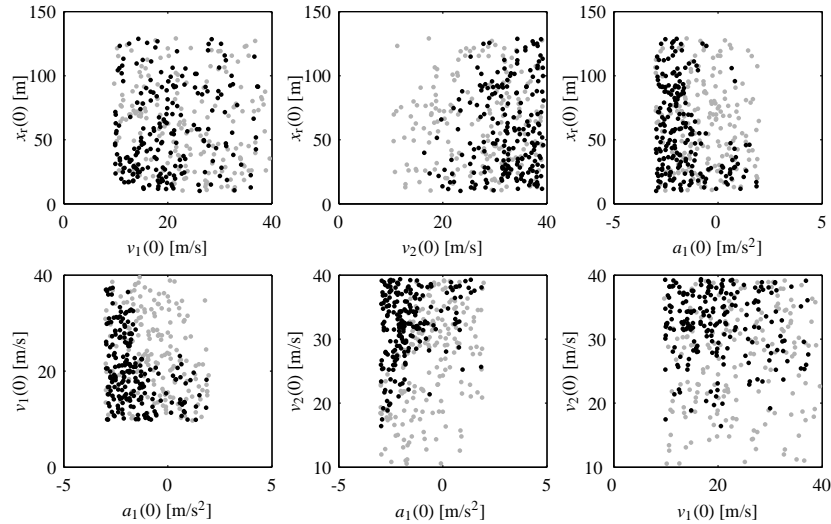


Figure 5.18: Scatter plots with the occurrence of collisions (black dots) and no collisions (grey dots) for the first AIS sequence of the parameter set \mathcal{Q} , using a two-dimensional representation for all parameter combinations.

5.4.3 Convergence of the AIS algorithm

The purpose of AIS is to reduce the variance of the Monte Carlo estimate or conversely, to reduce the sufficient sample complexity N to obtain a desired accuracy ϵ and confidence $1 - \delta$. However, due to the increased complexity of the simulation, it is not possible to perform $M = 10\,000$ sets of simulations, like in the examples of the previous section. So we cannot analyze the efficiency by checking the empirical $\hat{\epsilon}$ and $\hat{\delta}$. Instead, the potential for variance reduction is demonstrated by investigating the convergence of the AIS algorithm.

Example 5.9 Convergence of the AIS algorithm. To verify whether this is the case, the convergence of the estimate p can be checked for consecutive samples of the AIS algorithm. Figure 5.19 therefore shows the convergence of the cumulative estimate \hat{p}_{N_j} with the cumulative sample complexity $N_j \in [1, \dots, N]$, calculated as

$$\hat{p}_{N_j} = \frac{1}{N_j} \sum_{j=1}^{N_j} \frac{J(\varphi_j) f_{\mathcal{Q}}(\varphi_j)}{\xi_{\mathcal{Q}}(\varphi_j)}. \quad (5.43)$$

The comparison with simple sampling is also shown, which illustrates that the choice of a representative importance sampling PDF speeds up the convergence. The figure also illustrates that with each sequence convergence is reached earlier, because the importance sampling PDF $\xi_{\mathcal{Q}}$ gets closer to the *optimal* importance sampling PDF.

From Figure 5.19 can be seen that the AIS approach needs $N = 332$ samples to get within an accuracy of $\epsilon < 0.01$; and the simple sampling approach needs $N = 1015$ samples. This means that the importance sampling PDF introduces an efficiency gain of $1015/332 \approx 3$ in the *necessary* sample complexity. For this example it is known from a reference simulation set with high N that the real p is equal to 0.112 with very small ϵ and δ . The total estimate for the probability of clearance $1 - p$ for the other sampling methods are shown in Table 5.2, where we have selected equal sample sizes for a fair comparison. From this table follows that, compared to the Chernoff bound $N_{\text{ch}} = 23\,026$ for simple sampling, AIS needs only $4 \cdot 384 = 1\,536$ samples. This means an efficiency improvement of the *sufficient* sample complexity by a factor $23\,026/1\,536 = 15$.

Similar to Algorithm 5.3, it is also possible to use AIS for control validation problems, in case a relative accuracy is desired, as formulated in Algorithm 5.6. The application of this algorithm will be demonstrated with case studies, where the use of a relative accuracy is more appropriate (due to the low value of p in practice). However, these practical case studies also involve more realistic validation problems. It is therefore important to investigate the use of the testing tools PreScan, VeHIL, and test drives for application in these randomized algorithms. In the next section we will therefore present a methodological framework that integrates the validation methods with the testing tools that support them.

Table 5.2: Estimates \hat{p}_{N_j} for various sampling methods.

Sampling method	\hat{p}_{N_j}	Cumulative \hat{p}_{N_j}	N_j
Real p		0.112	
Simple sampling		0.094	384
Grid search		0.045	384
AIS sequence 1	0.123	0.123	384
AIS sequence 2	0.106	0.114	384
AIS sequence 3	0.103	0.111	384

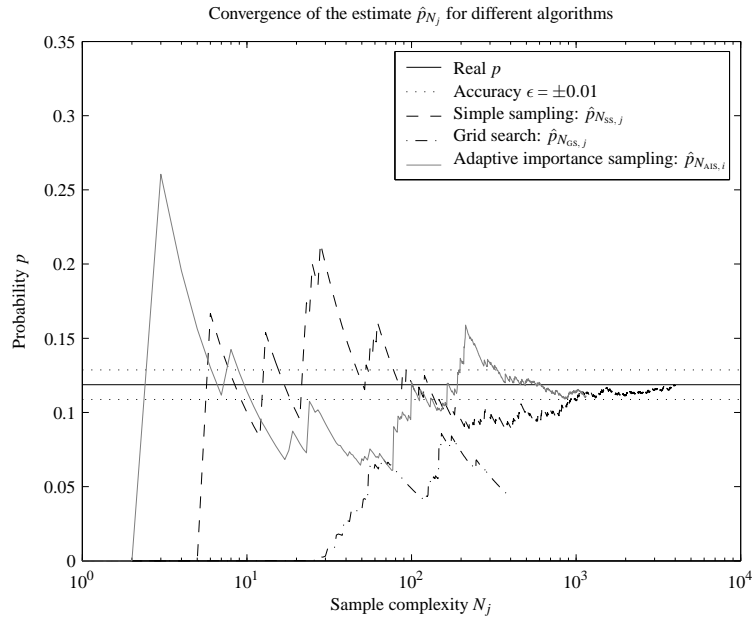


Figure 5.19: Convergence of the cumulative estimate \hat{p}_{N_j} with the number of samples N_j for simple sampling, grid search, and AIS. The real value with the accuracy interval of $\epsilon = \pm 0.01$ is also indicated.

5.5 A methodology using PreScan, VeHIL, and test drives

In the previous sections new algorithms for efficient validation of the performance and dependability of ADASs have been presented. These algorithms will be most useful for faster than real-time simulations, because of the large sample complexity (although N can be reduced to ‘only’ several thousands). However, the validity of the corresponding simulation results depends on the reliability of the simulation model and the underlying PDFs, from which the samples are drawn. It is therefore essential to validate these simulation results and the validity of the simulation model itself. Because of the model-based relation between PreScan simulations and VeHIL experiments that was presented in Chapter 4, the VeHIL laboratory will be an appropriate tool for this validation phase. Simultaneously, the model validation in VeHIL allows to improve the results of PreScan simulations. Consequently, the accuracy and confidence of the estimated performance and dependability measures can be improved in an iterative approach. Vice versa, the simulation phase can be used to provide a suitable test schedule for VeHIL. Unfortunately, the sample complexity based on the AIS algorithm is often still too large for testing in VeHIL. It should therefore be investigated how the number of VeHIL experiments can be further reduced.

The goal of this section is therefore to integrate the above mentioned activities in a *methodological framework for ADAS control system design and validation* using PreScan simulations, VeHIL experiments, and test drives. The phases that can be distinguished in this methodology are summarized in Algorithm 5.7. Each of the phases in Algorithm 5.7, including the tools that are used and the corresponding results, will be briefly described below.

Algorithm 5.6 Sequential estimation with AIS for a relative accuracy

Given a desired $\epsilon_r, \delta \in (0, 1)$, a performance measure ρ , a threshold $\gamma \geq 0$, and the true PDF f_Q for Q , this randomized algorithm returns with a probability of at least $1 - \delta$ an estimate $\hat{p}_{N_{AIS}}$ for p , such that $p - \hat{p}_N \leq p\epsilon_r$.

1. Draw $N_{SS,1} = \frac{2}{\hat{p}_0 \epsilon_{r,1}^2} \ln \frac{1}{\delta_1}$ iid samples \mathbf{q}_j , where $\delta_1 = \delta/\kappa$, $\epsilon_{r,1} = \kappa\epsilon_r$ with the factor κ and the *a priori* estimate \hat{p}_0 suitably chosen.

2. Return the empirical probability $\hat{p}_{N_{SS,1}} = \frac{1}{N_{SS,1}} \sum_{j=1}^{N_{SS,1}} J(\mathbf{q}_j)$.

The real p is less than or equal to $\hat{p}_{bin,1} = \hat{p}_{N_{SS,1}}(1 - \epsilon_{r,1})$ with confidence $1 - \delta_1$.

3. Determine the sample complexity $N_{bin,2}$ that is associated with $\hat{p}_{bin,1}$, $\delta_2 = \delta - \delta_1$, and $\epsilon_{r,2} = \epsilon_r$, using (5.18) evaluated at

$$z^- = \frac{(\hat{p}_{bin,1} - \hat{p}_{bin,1}\epsilon_{r,2})N_{bin,2} - \hat{p}_{bin,1}N_{bin,2}}{\sqrt{N_{bin,2}\hat{p}_{bin,1}(1 - \hat{p}_{bin,1})}}$$

4. Estimate the importance sampling PDF $\xi_{Q,2}$, based on the kernel density estimate (5.39) of the samples \mathbf{q}_j^* , for which $\rho_k(\mathbf{q}_j) \geq \gamma_k$.

5. Estimate the importance sampling reduction factor $\hat{\lambda}_{IS} = \frac{\hat{\sigma}_{IS}^2}{\hat{\sigma}_{SS}^2}$, with $\hat{\sigma}_{IS}^2$ from (5.33) and $\hat{\sigma}_{SS}^2$ from (5.31).

6. IF $N_{bin,2} \leq N_{SS,1}$

Return the empirical probability $\hat{p}_N = \hat{p}_{N_{SS,1}}$.

ELSE

Draw $N_{IS,2} = \hat{\lambda}_{IS}(N_{bin,2} - N_{SS,1})$ new samples from the importance sampling PDF $\xi_{Q,2}$.

Return the empirical probability

$$\hat{p}_{N_{AIS}} = \frac{1}{N_{SS,1} + N_{IS,2}} \left(\sum_{j=1}^{N_{SS,1}} J(\mathbf{q}_j) + \sum_{j=1}^{N_{IS,2}} \frac{J(\boldsymbol{\varphi}_j)f_Q(\boldsymbol{\varphi}_j)}{\xi_{Q,2}(\boldsymbol{\varphi}_j)} \right)$$

with a relative accuracy ϵ_r and confidence level $(1 - \delta_1)(1 - \delta_2) \geq 1 - \delta$.

The relations between these phases in the methodology are illustrated in Figure 5.20, although it should be noted that in practice the development process does not strictly follow all phases in this sequence. Also, this diagram is set up specifically from the viewpoint of system validation, whereas the traditional ‘V’-diagram of Figure 4.1 provides a more general view of the entire development process.

5.5.1 Definition of validation objectives

The first phase in the validation process is to define the validation objectives. As defined in Problem 5.4, the general validation problem is to estimate the probability p for insuffi-

Algorithm 5.7 Methodological framework for design and validation of advanced driver assistance systems

1. Definition of validation objectives.
2. Definition of the parameter set.
3. System design.
4. Generation of a simulation model of the system.
5. System construction, integration, and verification.
6. Sensitivity analysis of the performance measure.
7. Preliminary validation with adaptive importance sampling.
8. Improvement of the validation results with VeHIL.
9. Evaluation of the system benefit with test drives.

cient system performance on \mathcal{Q} , such that the cleared parameter set \mathcal{Q}^* and the non-cleared parameter set \mathcal{Q}^\bullet can be defined in probabilistic terms. In the previous sections efficient sampling methods have been investigated to estimate p . In this section, we will extend this to the investigation of the cleared and non-cleared subsets of \mathcal{Q} , and the definition of a corresponding test schedule. An unambiguous measure of clearance is provided by defining the requirements in terms of one or more performance measures ρ_k :

- Stability, *e.g.*, individual vehicle stability, string stability.
- Performance, *e.g.*, tracking error, time-to-collision.
- Comfort, *e.g.*, acceleration, naturalistic control performance, warning timeliness.
- Dependability, *e.g.*, false alarm rate, missed alarm rate, fault tolerance.

For each measure a corresponding criterion γ_k must be defined to assess whether the system is cleared, *i.e.*, $\rho_k(\mathbf{q}_j) < \gamma_k(\mathbf{q}_j)$, for a particular $\mathbf{q}_j \in \mathcal{Q}$, as summarized in Table 2.3.

The probability of clearance for all possible parameter combinations can only be found up to a desired level of accuracy and confidence. Suitable values for ϵ (or ϵ_r in case of a relative accuracy) and confidence $1 - \delta$ must therefore be defined.

5.5.2 Definition of the parameter set

The next phase is to identify the parameter set \mathcal{Q} , which consists of:

- Scenario parameters, *e.g.*, scenario configuration, inter-vehicle motion.
- Driver interaction, *e.g.*, car-following behavior, warning responsiveness.
- Disturbances, *e.g.*, sensor measurement noise, and weather and ambient conditions.
- Modeling errors, *e.g.*, parametric uncertainty and unmodeled dynamics.
- Failure modes, *e.g.*, faults in sensors, actuators, or controller hardware.

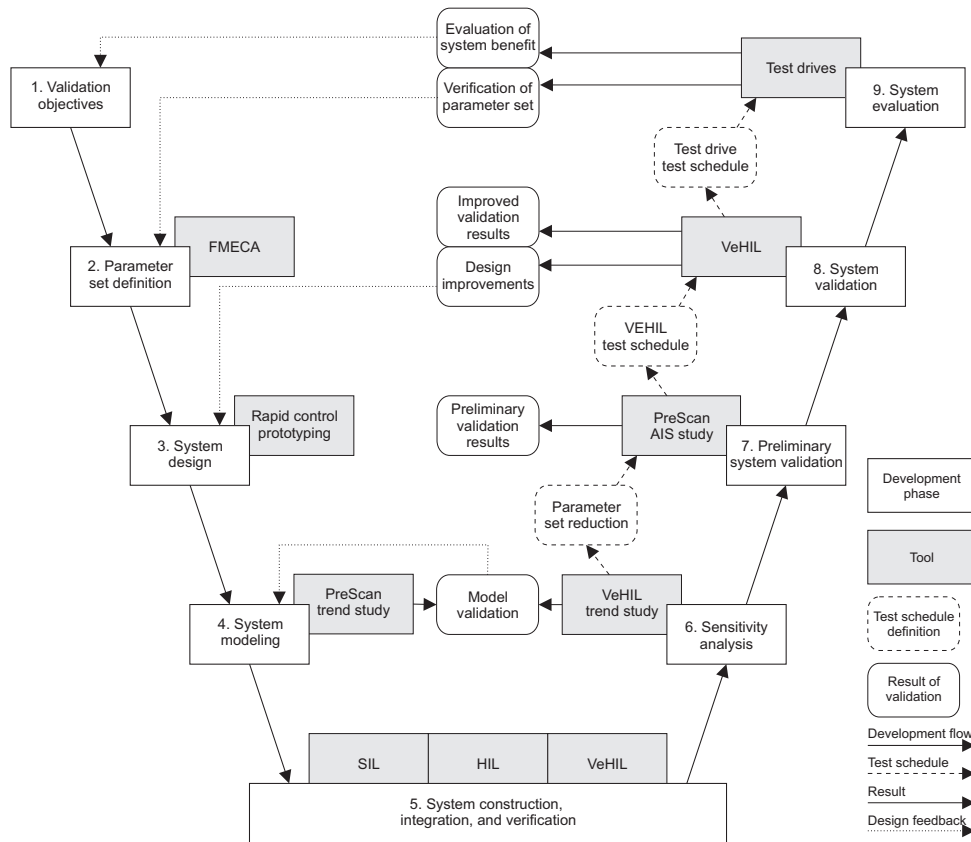


Figure 5.20: Schematic representation of the methodological framework for validation, which extends the traditional 'V'-diagram from the viewpoint of system validation.

The underlying distribution of Q in terms of a multivariate PDF f_Q must be found from experimental data, as was presented in Sections 2.4 and 2.5. Obviously, this PDF would result in relatively few positive samples, see Step 0 in Figure 5.21(a). FMECA can be used to identify the critical failure modes. Furthermore, scenario and accident analysis can help to identify the most relevant scenarios. This reduction of the parameter set (also illustrated in Figure 5.21(a)) is used to speed up the subsequent process of simulation and testing.

5.5.3 System design

After the definition of the system requirements and parameter set, the design phase covers the definition of system specifications and the design of the system architecture, followed by detailed module design. Design issues are not further discussed here, although some elements of ADAS design will be treated in the case studies. Nevertheless, the design phase strongly depends on the generation of a simulation model that is used for specification development and controller synthesis. Furthermore, this simulation model is useful for preliminary validation purposes, as discussed next.

5.5.4 Generation of a simulation model of the system

For clearance of the ADAS, it must be demonstrated that the performance ρ meets the criteria γ for all traffic scenarios and system variations. This means that the assessment must be performed not only for the nominal model but also for all possible parameter deviations, operating conditions, and failure modes. Some of these variations are known, whereas others are uncertain and known only within certain bounds. In this modeling phase, the bounded uncertainty set $\mathcal{D} \subset \mathcal{Q}$ is defined with minimum and maximum values for each uncertainty Δ_i in the uncertainty vector $\Delta \in \mathcal{D}$, representing model uncertainty, unmodeled dynamics, and parametric uncertainty.

Although a reliable model is obviously required for retrieving useful validation results, the model should not be overly complex. This requires a trade-off between simulation speed and transparency versus complexity and reliability. Obviously, system elements that affect the relation between system inputs and outputs should be modeled. For example, in case of an ACC system, it makes sense to include engine and brake system dynamics, but steering dynamics could be neglected. A PreScan simulation model can then be developed that incorporates suitable models for the vehicle, the sensor, and a microscopic traffic scenario.

The effect of model uncertainty on performance and dependability varies with ADAS type, control system, and sensor systems. Before the actual validation is started, the relative importance of the different uncertainties on the validation results should be investigated in a sensitivity analysis. Parameters with minor effects can thus be excluded from further assessment in order to reduce the amount of required calculations.

For each Δ_i a mean, minimum, and maximum value is considered in the D-optimal design, see Figure 5.21(b). When the analysis of variance shows that an uncertainty Δ_i has no effect on the performance ρ_k , the set \mathcal{D} can be reduced by neglecting this uncertainty. The number of runs for the D-optimal design is given by Step 1 of Algorithm 5.6.

5.5.5 System construction, integration, and verification

After implementation and verification of the individual hardware and software modules, the complete system is assembled from its components. This system integration phase is carried out in parallel to the validation phases, as the integration of the system progresses.

5.5.6 Sensitivity analysis of the performance measure

Since the results of the validation process are subject to the validity of the models used, model validation is an important element of the methodological framework. Because of the model-based design framework, validation of PreScan simulation models can be done straightforward by executing the same test schedule in the VeHIL laboratory, based on the D-optimal design, as illustrated in Figure 5.21(c). From this initial test program, preliminary conclusions can be drawn on the validity of the simulation models and the importance of specific test vectors \mathbf{q}_j . If a major discrepancy occurs between the simulation model and the VeHIL test, the simulation model should be adapted and rerun until the simulation output matches the VeHIL test results. This comparison between VeHIL and PreScan simulation, as well as the test analysis itself, is made in terms of the performance measures ρ_k .

In addition, these preliminary VeHIL results allow model reduction, which speeds up simulation time. Furthermore, from the response surfaces $\hat{\rho}(\mathbf{q})$ generated by the D-optimal design, and subsequent sensitivity analysis, the subsets \mathcal{Q}_0^* and \mathcal{Q}_0° , for which the outcome

(cleared or non-cleared) is *a priori* known, can be neglected for the remaining phases of the validation.

5.5.7 Preliminary validation with adaptive importance sampling

When a reduced parameter set and a validated simulation model have been identified, a simulation study is used to provide a preliminary clearance of \mathcal{Q} . The results of the simulation phase are estimated measures for the performance and dependability of the ADAS with respect to the criteria defined.

This simulation study is carried out through adaptive importance sampling, using Algorithm 5.6. The importance sampling PDF for the AIS simulation strategy is formed by the response surface $\hat{\rho}(\mathbf{q})$ that is constructed from the results of the sensitivity analysis of the previous phase. The AIS simulation strategy allows a representative set of samples $\mathbf{q} \in \mathcal{Q}$ to be investigated, the controller behavior $\rho(\mathbf{q})$ to be analyzed, and important areas of the parameter set to be identified.

Since the performance of Algorithm 5.6 depends heavily on the reliability of the models and the PDFs used in the simulation phase, the robustness of the probability estimate \hat{p}_N to model uncertainty Δ should be considered. The experimental relation between ρ_k and \mathcal{Q} from the simulations is then bounded, such that

$$\rho_{k,\text{AIS}}(\mathbf{q}, \Delta^-) \leq \rho_k(\mathbf{q}) \leq \rho_{k,\text{AIS}}(\mathbf{q}, \Delta^+) \quad \text{for all } \mathbf{q} \in \mathcal{Q} \text{ and all } \Delta \in \mathcal{D}. \quad (5.44)$$

This bounded estimate for the performance has been illustrated for the two-dimensional case in Figure 5.21(d). Correspondingly, the AIS estimate \hat{p}_N has an upper and lower bound, depending on the value of the uncertainty: $\hat{p}_{\text{AIS},\Delta^-} < \hat{p}_{\text{AIS}} < \hat{p}_{\text{AIS},\Delta^+}$. This means that the real boundary $\rho = \gamma$ is expected to lie within the subset $\mathcal{Q}^\circ \subset \mathcal{Q}$ that is bounded by $\rho_{\text{AIS}}(\mathbf{q}, \Delta^-) = \gamma$ and $\rho_{\text{AIS}}(\mathbf{q}, \Delta^+) = \gamma$. Based on these preliminary findings, a test schedule for VeHIL experiments and test drives with the real ADAS-equipped vehicle can be drawn up.

5.5.8 Improvement of the validation results with VeHIL tests

After the randomized validation using AIS, VeHIL tests should be performed to validate the simulation results, and provide an estimated probability \hat{p} with higher accuracy and confidence than is possible with simulations. The most relevant scenarios have to be identified, using thresholds for the performance criteria. This relates to the area between the curves $\rho_{\text{AIS}}(\mathbf{q}, \Delta^-) = \gamma$ and $\rho_{\text{AIS}}(\mathbf{q}, \Delta^+) = \gamma$ of Figure 5.21(d). We therefore select a limited number of parameter combinations \mathbf{q}_j to be reproduced in the VeHIL facility, as illustrated by the corresponding shaded area in Figure 5.21(e).

In VeHIL parameter variations, disturbances and failure modes are introduced very accurately. The model uncertainty is reduced, because of the replacement of the vehicle and sensor model by real hardware. The boundary between \mathcal{Q}° and \mathcal{Q}^\bullet and the effect of disturbances on this boundary can therefore be investigated more precisely. Furthermore, an estimate \hat{p}_{VeHIL} can be achieved that is more reliable than \hat{p}_{AIS} :

$$\hat{p}_{\text{VeHIL}} = \frac{1}{N_{\text{AIS}}} \left(\sum_{j=1}^{N_{\text{AIS}}} \frac{J(\boldsymbol{\varphi}_{\text{AIS},j}^\bullet) f_{\mathcal{Q}}(\boldsymbol{\varphi}_{\text{AIS},j}^\bullet)}{\xi_{\mathcal{Q},\text{AIS}}(\boldsymbol{\varphi}_{\text{AIS},j}^\bullet)} \right) + \frac{1}{N_{\text{VeHIL}}} \left(\sum_{j=1}^{N_{\text{VeHIL}}} \frac{J(\boldsymbol{\varphi}_{\text{VeHIL},j}) f_{\mathcal{Q}}(\boldsymbol{\varphi}_{\text{VeHIL},j})}{\xi_{\mathcal{Q},\text{VeHIL}}(\boldsymbol{\varphi}_{\text{VeHIL},j})} \right), \quad (5.45)$$

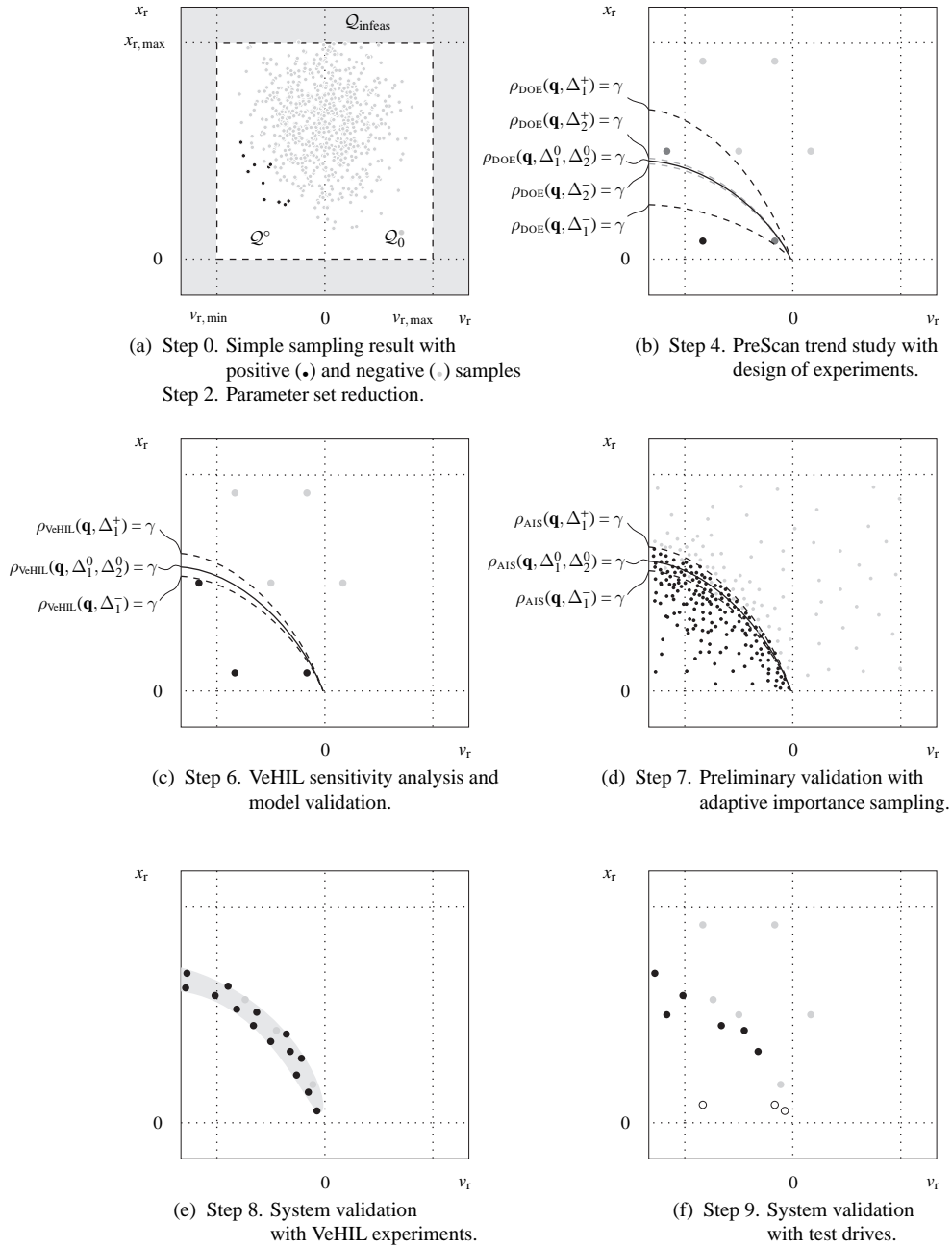


Figure 5.21: Illustration of several steps of the methodological framework of Algorithm 5.7 using the two-dimensional range – range rate diagram.

where $\boldsymbol{\varphi}_{\text{AIS},j}^\bullet$ are the N_{AIS}^\bullet samples of Step 7 from Algorithm 5.7, for which the indicator function J is always positive, that is: $\rho_{\text{AIS}}(\boldsymbol{\varphi}_j) > \gamma$ for all $\boldsymbol{\Delta} \in \mathcal{D}$. The estimate \hat{p}_{VeHIL} and the estimated volume of the non-cleared set \hat{Q}^\bullet may indicate necessary improvements in the system design regarding fine-tuning of the controller parameters.

5.5.9 Evaluation of the system benefit with test drives

Finally, an important issue is to investigate whether the results of VeHIL experiments and PreScan simulations are representative of the ‘real’ performance. Test drives are therefore always the final link in the system validation phase. They are also used to investigate the validity of the VeHIL and PreScan results. Similarly to the definition of the VeHIL test schedule from the simulation results, we can define a test drive program from the VeHIL results. This validation phase is done by executing the same grid as used in Steps 4 and 6 to validate the simulation model and representativeness of VeHIL. Furthermore, a selection of test vectors \mathbf{q}_j can be chosen from the set $\mathcal{Q}_{\text{VeHIL}}$, to be further investigated on the test track, as indicated in Figure 5.21(f). Unfortunately, some tests will be too difficult or dangerous to perform on a test track, *e.g.*, tests with short initial distance $x_r(0)$, indicated by the symbol “o” in Figure 5.21(f).

5.6 Summary

We have presented a methodological framework for probabilistic validation of advanced driver assistance systems (ADASs), based on a new randomized algorithm for adaptive importance sampling. This probabilistic approach cannot *prove* that the system has adequate performance and dependability. However, when we accept a (small) risk of failure, this probabilistic approach *is* able to obtain an efficient estimate of the performance and dependability of the system. Especially compared to conventional grid-based validation and Monte Carlo simulation, our approach is significantly more efficient in terms of the number of experiments that is sufficient to obtain a desired level of accuracy and confidence. In contrast to traditional μ -analysis methods, performance measures can include time-domain and Boolean criteria, instead of only frequency-domain criteria, which makes it more useful for practical application to ADASs.

The level of confidence in the estimate \hat{p} depends on many factors. First, the sufficient sample complexity N is an important prerequisite for proper validation. It has been reduced in a sequential approach by reformulation of the validation problem, such that a more strict bound on N is obtained. In addition, the definition of the parameter set \mathcal{Q} and its PDF $f_{\mathcal{Q}}$ that are used for the validation process play an important role, as the simulations are sampled from this distribution. The use of a randomized algorithm for adaptive importance sampling allows to use a more efficient representation of \mathcal{Q} . The validation of the PreScan simulation results with VeHIL experiments allows to reduce this model uncertainty and improve the estimate \hat{p} in an iterative approach.

Another important element of the methodology is that *a priori* information on the system behavior can be included in the test schedule, such that the most relevant subsets in the parameter set are investigated. Especially design of experiments theory can be efficiently used to investigate the sensitivity of the performance measure to particular parameters or combinations thereof. Critical parameter combinations are then subsequently emphasized

Table 5.3: Application of elements of the methodological framework to the case studies.

	Tools					Development phases								
	PreScan model-in-the-loop	PreScan software-in-the-loop	Rapid control prototyping	Vehicle hardware-in-the-loop	Test drives	1. Validation objectives	2. Parameter set definition	3. System design	4. System modeling	5. System construction	6. Sensitivity analysis	7. Preliminary validation	8. System validation	9. System evaluation
Driver warning system	•		•	•	•	•	•		•			•	•	
Cooperative adaptive cruise control	•		•	•	•	•	•	•	•	•	•			•
Pre-crash system	•			•		•	•		•		•			•

in the validation process.

A major advantage of the methodology is that the sufficient sample complexity for PreScan simulations and VeHIL experiments can be predicted *a priori*. This allows to allocate the appropriate time and resources that are required for the validation of a control system. This is an improvement over the trial-and-error methods that are currently used in practice, and for which the required time and resources can only be approximated beforehand, based on engineering judgement. The sample complexity is based on the adaptive importance sampling reduction factor that is calculated using a prediction of the number of positive samples that are representative of the importance sampling PDF ξ_Q . This method considerably reduces the *sufficient* sampling complexity, almost to the minimum possible number (*i.e.*, the *necessary* sample complexity). The only excess conservatism in the sample complexity is caused by a suboptimal choice for the factor κ , which reflects the ratio between the sample complexities in the subsequent steps of the AIS algorithm.

In summary, the purpose of PreScan is to provide a preliminary validation of the ADAS using the AIS algorithm. The role of VeHIL is to validate the simulation model, as well as refine the simulation results to a more reliable value. Finally, test drives are used to confirm the PreScan and VeHIL results, and for obtaining practical knowledge of Q and f_Q .

The methodology has been illustrated with a simple case study, involving an ACC-equipped vehicle following a target vehicle. Depending on the validation objective, this case study has shown a more than tenfold increase in efficiency compared to conventional methods. It is expected that the relative efficiency increases with a higher dimension of the parameter set Q . This will be demonstrated in the next chapters, where the methodology will be applied to practical case studies, involving a driver warning system (Chapter 6), a cooperative adaptive cruise control system (Chapter 7), and a pre-crash system (Chapter 8). Although it was not possible within the framework of this thesis to apply all tools and all phases of the methodology to each of these case studies, each phase and tool is applied to at least one of the cases. Table 5.3 gives a preliminary outlook on the development phases of these case studies and the tools that are applied.

Chapter 6

Case study: Validation of a driver information and warning system

This chapter is the first of three that demonstrate the application of the validation methodology that was developed in the previous chapter. The subject in the present case study is a novel driver information and warning system for safe speed and safe distance (SASPENCE). This system communicates the appropriate velocity and inter-vehicle distance to the driver in potentially dangerous situations. This case study investigates the steps of Algorithm 5.7 that are indicated as dark grey blocks in Figure 6.1. Section 6.1 introduces the functional requirements and operating conditions of the SASPENCE system, as well as the implementation of the system in a demonstrator vehicle. Sections 6.2 and 6.3 define the validation objectives, evaluation criteria, and parameter set for this case study. Subsequently, Sections 6.4 and 6.5 present the simulation model and the validation results of the randomized simulation study. From this study, critical scenarios are chosen to be replayed in VeHIL, as demonstrated in Section 6.6. The preliminary results of subjective driving tests are briefly reviewed in Section 6.7. Finally, Section 6.8 summarizes the results of this chapter.

6.1 A system for safe speed and safe distance

Many traffic accidents in the EU are caused by inappropriate vehicle speed or inter-vehicle distance. It has been estimated that excessive speed accounts for one-third of all road accidents, and contributes to around 1200 fatalities and over 100 000 injuries on European roads every year [222]. In addition, another 15 % of all road accidents are rear-end collisions, which is usually caused by drivers keeping insufficient distance [129].

6.1.1 Functional requirements

One of the initiatives to improve traffic safety by intelligent vehicle systems is the Integrated Project PReVENT (Preventive and Active Safety Applications), co-funded by the European Commission under the Sixth Framework Programme [105]. The goal of the PReVENT sub-project SASPENCE is the development and evaluation of a ‘Safe Speed and Safe Distance’ application that supports the driver in avoiding potentially dangerous situations and that im-

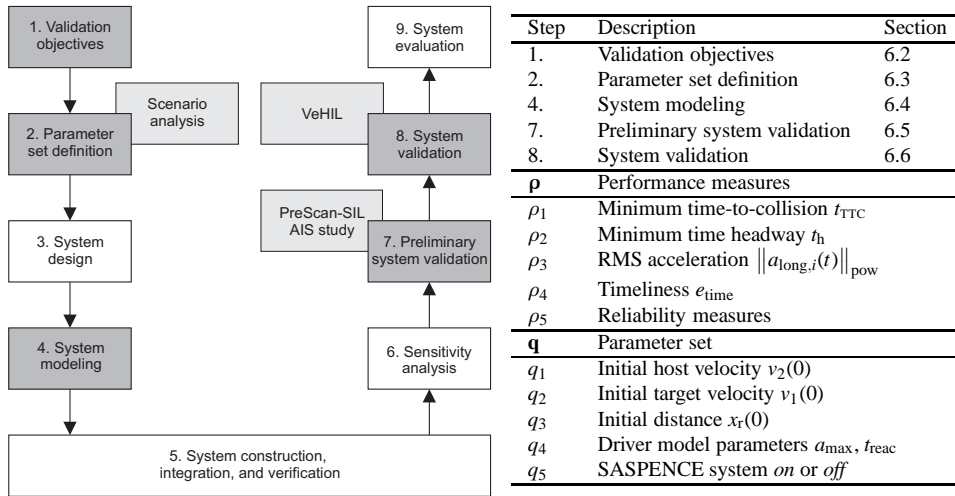


Figure 6.1: Steps in the methodology that are investigated for the SASPENCE case study.

proves driver comfort [196]. The SASPENCE system should cooperate unobtrusively with the driver by suggesting a safe speed and safe distance to keep, relative to the vehicle in front [234]. In addition, the system gives a speed advice, taking into account speed limits, road infrastructure, and weather conditions.

6.1.2 Relevant scenarios for driver warning

Since speeding and tailgating are widespread traffic hazards, the SASPENCE system considers a wide range of traffic scenarios and operating conditions, as listed in Table 6.1. Some of these scenarios are illustrated in Figure 6.2. The table also indicates the corresponding system response in terms of the warning level w . Warning level 0 means that no warning is given, level 1 is an information mode, level 2 a mild warning, and level 3 an emergency warning.

Since the safety potential that is expected from a system capable to appropriately warn the driver in case of excessive speed and small headway looks very promising [235], it is of paramount importance to accelerate the deployment of such an ADAS. In order to have a system ready for the short-term market, the SASPENCE project aims to develop a low-cost system by combining ADAS components that are already available in modern passenger cars (such as components for ACC, lane departure warning, and satellite navigation). The corresponding system architecture based on existing hardware components is therefore presented next.

6.1.3 System architecture and prototype description

The SASPENCE system is installed in a Fiat Stilo Multiwagon that serves as one of the two demonstrator vehicles for this project. In the system architecture of Figure 6.3 several modules can be distinguished. The sensor array of the SASPENCE system consists of a long-range ACC radar for obstacle detection, mounted on the front of the vehicle. Lane

Table 6.1: Traffic scenarios and operating conditions for the SASPENCE system.

No. ^a	Scenario description	Warning level w
1	Host vehicle breaks speed limit	1
2	Critical weather conditions present (e.g., fog, heavy rain, snow)	1
3	Obstacle appears ahead, but not on the host path	0
4	Obstacle appears ahead on the host path, without being dangerous	0
5	Obstacle appears ahead and could become dangerous	2
6	Obstacle appears ahead and is dangerous	3
7	On-coming vehicles approaching (on one-way rural road)	1
8	Host vehicle approaches hazardous infrastructure too fast (e.g., sharp bend, traffic light, or pedestrian crossing)	1

^aSee Figure 6.2 for an illustration of these scenarios (except scenario 7).

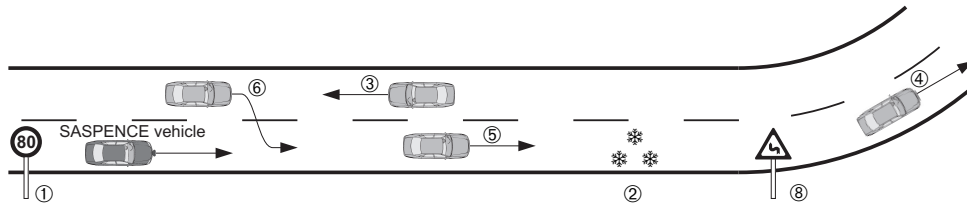


Figure 6.2: Overview of scenarios for the SASPENCE system.

recognition by video image processing is used to distinguish potentially dangerous obstacles from objects in adjacent lanes and on the side of the road. In addition, vehicle-to-vehicle communication (VVC) enhances the selection of relevant targets. Differential GPS (DGPS) combined with digital map navigation is used for global state estimation and for providing information on speed limits and relevant infrastructure [6]. Several human-machine interface (HMI) channels are available to provide information and warnings to the driver: a haptic accelerator pedal (through force feedback and pedal vibration), a visual warning display, seat belt vibration, and audio signals. In-vehicle networking between the sensor array, the signal processing modules, and the HMI is primarily provided by a dedicated CAN bus.

6.1.4 Sensor fusion and scenario assessment

Sensor data is fused at multiple levels to provide an enhanced view of the environment. Sensor fusion of DGPS and vehicle state sensors based on extended Kalman filtering, similar to the system presented in Chapter 3, provides an estimate of the host vehicle's global state $\hat{\mathbf{x}}$. A precise estimation of the road course ahead is created by fusion of navigational map points and lane detection information [261]. All detected vehicles are projected into the estimated road geometry to determine their relative positions to the host vehicle and their predicted paths [63].

The output of the sensor fusion and path prediction modules is then used to compute an optimal reference maneuver by solving the optimization problem

$$J = \frac{1}{S} \int_0^S f(\mathbf{x}(s), \mathbf{u}(s)) ds, \quad (6.1)$$

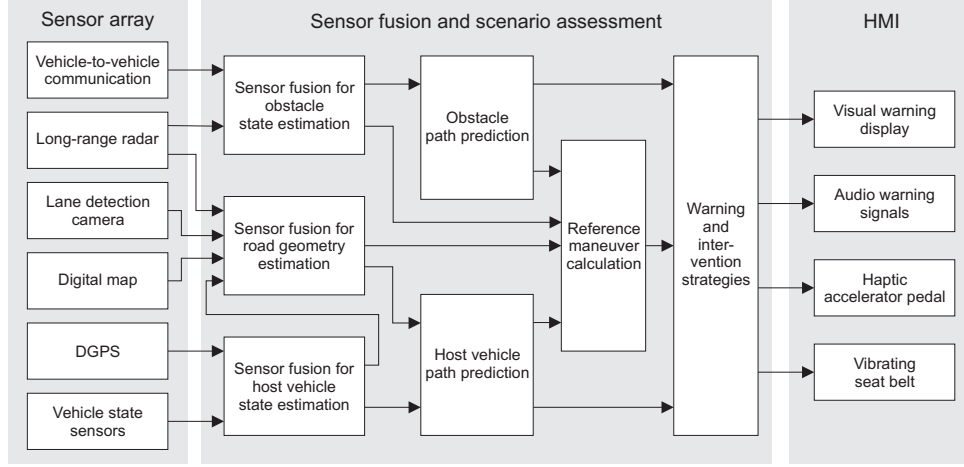


Figure 6.3: System architecture of the SASPENCE system.

subject to a given set of inequality constraints (used to impose trajectory constraints) and equality constraints (includes the vehicle model with $\mathbf{x}(s)$ the vehicle state vector). The functional J has to be minimized over the planning distance interval $[0, S]$ by finding the control functions $\mathbf{u}(s)$ (steering, throttle, and brake input). The penalty function $f(\mathbf{x}(s), \mathbf{u}(s))$ is used to define the driving style. In addition, the penalty function can be considered a risk performance measure, where the integral is a measure for the risk level of the maneuver. Apart from these safety considerations, requirements for user acceptance and mobility are included in the penalty function. This allows the SASPENCE system to compute an appropriate speed and safe distance to the preceding vehicle, as well as to consider speed limits and weather conditions.

The optimal reference maneuver is then compared to the predicted path of the host vehicle. In case the difference crosses a threshold, the system intervenes by giving information and/or warnings to the driver. The warning and intervention module computes the appropriate warning type and warning level, and directs this to the available HMI channels. For more information on the calculation of the reference maneuver for the SASPENCE system, we refer to the work by Biral *et al.* [17]. In the remainder of this chapter we focus on the validation of the system's capability for giving appropriate and reliable warnings in response to potentially dangerous obstacles ahead.

6.2 Definition of the validation objectives

The first step in the validation process is to define suitable evaluation criteria ρ from the system requirements, with emphasis on measures that relate to an appropriate interaction with the driver. The dependability is assessed through the missed alarm rate

$$\hat{p}_{FN} = \frac{1}{\sum T_j} \sum_{j=1}^N \frac{\bar{J}_{FN,j}(\varphi_j) T_j f_Q(\varphi_j)}{\xi_Q(\varphi_j)}, \quad (6.2)$$

where $\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_j, \dots, \boldsymbol{\varphi}_N$ are N randomized samples from \mathcal{Q} with importance sampling PDF $\xi_{\mathcal{Q}}$, as discussed in Chapter 5. The indicator function \bar{J} for the sample-based missed alarm rate for scenario j with time duration T_j , defined by $\boldsymbol{\varphi}_j$, is given as (see also Figure 2.7):

$$\bar{J}_{\text{FN},j}(\boldsymbol{\varphi}_j) = \frac{1}{T_j} \int_0^{T_j} J_{\text{FN},j}(t) dt, \quad (6.3)$$

where for each sample k

$$J_{\text{FN},j}(k) = \begin{cases} 1 & \text{if } w_j(k) = 0 \text{ and } w_{\text{ref},j}(k) = 1, \\ 0 & \text{else.} \end{cases} \quad (6.4)$$

For each individual scenario j , this performance measure is calculated by comparison of the output of the SASPENCE system w_j with a reference system $w_{\text{ref},j}$, which gives information on the warning level and when it *should* be given from the perspective of an average attentive driver. This reference warning is based on empirical data from the CAMP project [132], that indicates the most appropriate warning time and warning level of a forward collision warning system for a representative set of drivers. This database has been summarized in the algorithm of (2.5), which is repeated here as:

$$a_{\text{ref}} = \begin{cases} 0.685a_1 - 0.086(v_1 + a_2 t_{\text{reac}}) - 1.617 & \text{if } t_{\text{stop},1} \leq t_{\text{reac}}, \\ 0.685a_1 - 0.086(v_r + a_r t_{\text{reac}}) - 0.833 & \text{if } t_{\text{stop},1} > t_{\text{reac}}. \end{cases} \quad (6.5)$$

A warning is issued when a_{ref} falls below the thresholds discussed in [132].

Similarly, the false alarm rate is defined by replacing (6.4) with

$$J_{\text{FP},i}(t) = \begin{cases} 1 & \text{if } w_i(t) = 1 \text{ and } w_{\text{ref},i}(t) = 0, \\ 0 & \text{else.} \end{cases} \quad (6.6)$$

The associated probabilities can then be estimated according to the methodology of Chapter 5, considering the underlying probability function $f_{\mathcal{Q}}$ and the importance PDF $\xi_{\mathcal{Q}}$ for the individual scenarios. Furthermore, the timeliness of a scenario j indicates to what extent the warning is given too soon or too late:

$$e_{\text{time},j} = t_{\text{warn},j} - t_{\text{ref},j} \quad (6.7)$$

with $t_{\text{warn},j}$ the first sample for which $w_j = 1$ and $t_{\text{ref},j}$ the first sample for which $w_{\text{ref},j} = 1$.

The impact of the system on traffic safety and driver comfort is validated by comparing a scenario where a driver is assisted by the SASPENCE system to the same scenario where the SASPENCE system is not operational. The safety is measured in terms of the minimum time-to-collision (TTC) during the scenario:

$$\min_{t \in [0, T]} t_{\text{TTC},i}(t), \quad (6.8)$$

and the comfort in terms of the RMS value of the longitudinal acceleration:

$$\|a_{\text{long},i}(t)\|_{\text{pow}}. \quad (6.9)$$

With respect to the dependability of the system, we require a maximum false alarm rate p_{FP} in the order of $1 \cdot 10^{-4}$, and a maximum missed alarm rate p_{FN} in the order of $1 \cdot 10^{-3}$.

Table 6.2: Values for the Gipps parameters a_{\max} and t_{reac} , as well as the reaction time to unexpected events t_{unexp} .

Driver	Without SASPENCE			With SASPENCE		
	Conservative	Intermediate	Aggressive	Conservative	Intermediate	Aggressive
a_{\max} [m/s ²]	1.17	1.55	2.05	1.17	1.55	2.05
t_{reac} [s]	0.95	0.73	0.56	1.15	0.93	0.76
t_{unexp} [s]	1.61	1.28	0.95	0.99	0.73	0.47

Furthermore, we require that the combination of the correct alarm rate p_{CP} and the true positive rate p_{TP} in the generalized reliability measure p_{rel} is at least 0.99. These values are chosen in accordance with the state-of-the-art values discussed on page 31.

The objective of this case study is to validate the comfort, performance, and dependability of the warning functions of the SASPENCE system against the system requirements. The system must conform to the requirements for a representative set of traffic scenarios, operating conditions, and driver characteristics. Let us further assume a desired relative accuracy $\epsilon_r = 0.1$ of the validation results and an associated confidence level $1 - \delta$ with $\delta = 0.01$.

6.3 Definition of the parameter set

The value of the performance measures $\rho_k(\mathbf{q}_j)$ for a particular traffic scenario j obviously depends on the perturbations \mathbf{q}_j imposed by that scenario. Based on the system specifications [6], a parameter set \mathcal{Q} is defined, composed of traffic scenarios, operating conditions, sensor characteristics, and driver characteristics.

For the traffic scenario parameters we use the microscopic traffic model of Chapter 2. The measurement noise of the environment sensors is taken from the system specifications [234]. The driver is modeled for conventional car-following behavior after the Gipps model (2.36). Table 6.2 gives values for the Gipps parameters for three different driver types: conservative, intermediate, and aggressive drivers. The maximum acceleration level a_{\max} for these three driver types is calculated for the 5 %, 50 %, and 95 % value of the cumulative distribution function that corresponds to (2.44), respectively. Similarly, the reaction time t_{reac} that these driver take into account during car-following are given by (2.47). Note that that this value represents the parameter that is used in the Gipps car-following model and relates to the reaction time that the driver anticipates (and subsequently to the headway that the driver applies). It is *not* equivalent to the reaction time t_{unexp} to *unexpected* events, such as a suddenly appearing target vehicle, which is obtained from [56].

6.4 Software-in-the-loop simulation in PreScan

The simulation model is developed in PreScan and consists of the SASPENCE-equipped host vehicle, a target vehicle and a radar model. The sensor fusion, scenario assessment, and HMI modules of the SASPENCE system are emulated by the actual real-time code in a software-in-the-loop (SIL) simulation. In addition, the driver model is implemented in the SASPENCE vehicle in order to provide a desired car-following behavior, whereas the target vehicle executes a simple longitudinal velocity profile. The simulation model is set up in

the PreScan environment in such a way that it can be called from MATLAB automatically in a randomized fashion. For this purpose, the adaptive importance sampling Algorithm 5.5 has been implemented in a MATLAB routine that communicates with PreScan.

In this case study it is assumed that the vehicle model, driver model, and sensor model are correct, and the model validation of Step 6 in Algorithm 5.7 is not carried out. Instead, this case study focusses on Steps 7 and 8 of the methodological framework, where we assume that *a priori* knowledge on the parameter set \mathcal{Q} is available. First, \mathcal{Q} is reduced by neglecting all situations for which $v_r(0) > 0$, since these will never result in a warning from the SASPENCE system. Secondly, in this chapter we consider only approach scenarios with two scenario parameters: $x_r(0)$ and $v_r(0)$. This restriction on \mathcal{Q} is introduced in order to facilitate validation of the simulation results with VeHIL experiments, as discussed later on. Next, the remaining subset of \mathcal{Q} is sampled using the Latin hypercube method, instead of using the original PDF $f_{\mathcal{Q}}$. Every simulation is carried out six times with the same initial conditions: for three driver types (conservative, intermediate, and aggressive), each with the SASPENCE system either *on* or *off*.

Since it was not possible in this case study to execute Steps 1 to 3 of Algorithm 5.6, we assume that the sample complexity for the adaptive importance sampling phase is in the same order as for the other examples of the previous chapter. We therefore choose $N_{\text{IS},2} = 10^3$.

6.5 Preliminary validation with PreScan-SIL simulation

This section presents the results of the simulation study for only one traffic configuration (approach scenario) and for a limited set of performance measures. Figures 6.4, 6.5, and 6.6 illustrate the impact of the SASPENCE system on the minimum time-to-collision t_{TTC} , the minimum time headway t_{h} , and the RMS value of the longitudinal acceleration $\|a_{\text{long},i}(t)\|_{\text{pow}}$, respectively. Results are shown for different drivers (conservative, intermediate, and aggressive), both with and without support of the SASPENCE system. It can be observed that the minimum TTC and minimum time headway, which are important safety indicators in car-following situations, increase when the SASPENCE system is active. Note that these figures only show a two-dimensional cross-section of the n -dimensional parameter set \mathcal{Q} . The reader is referred to [72] for results on other traffic configurations and scenario parameters.

Another aspect in the simulation study is an assessment of the comfort in terms of the RMS values of the acceleration. It can be seen that this performance measure is reduced, especially for the intermediate and aggressive driver, which means that comfort has increased. Considering these results, the SASPENCE system is expected to offer a significant benefit for traffic safety and driver comfort.

6.6 Functional validation with VeHIL tests

Obviously, in practice simulations have their limitations with regard to the credibility of the results. Driving simulator tests have therefore been carried out in [27], but these do not take into account the actual hardware of the vehicle and the SASPENCE system. On the other hand, driving tests with the demonstrator vehicles have also been performed [72], but they are limited in their ability to test safety-critical scenarios. To provide a preliminary

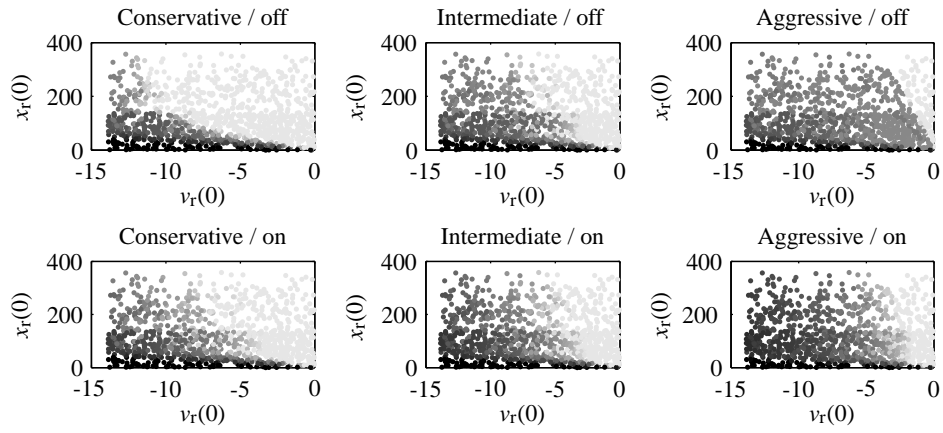


Figure 6.4: SASPENCE simulation results for the minimum TTC.

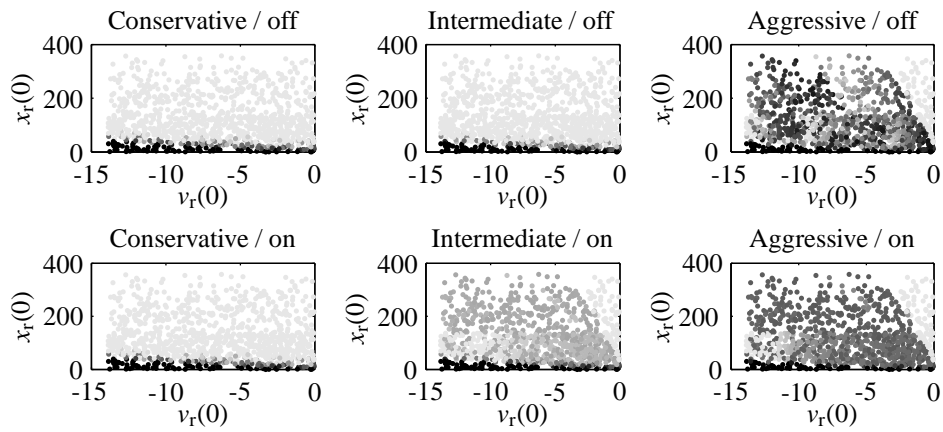


Figure 6.5: SASPENCE simulation results for the minimum time headway.

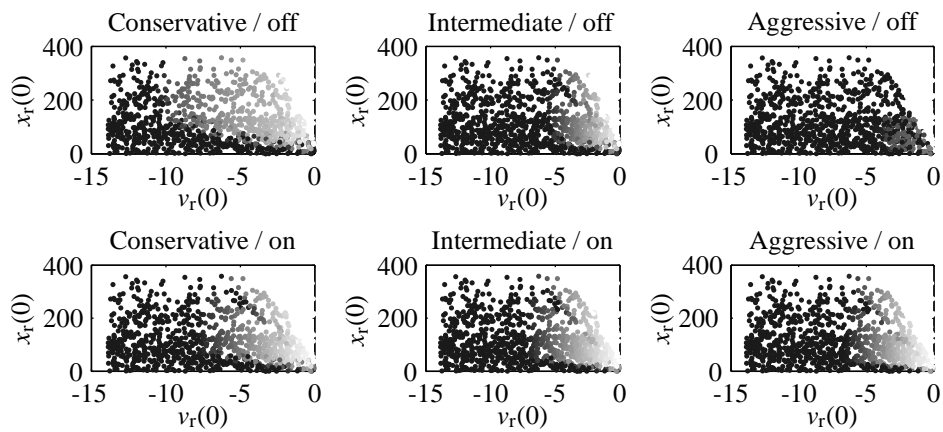


Figure 6.6: SASPENCE simulation results for the RMS value of the longitudinal acceleration.



Figure 6.7: VeHIL laboratory setup: (a) the SASPENCE vehicle is set up on the chassis dyno (beneath the floor), and approaches another road user, represented by the moving base; (b) the real-time traffic scene is projected on a display in front of the camera system (located behind the rear-view mirror), while a preliminary HMI display shows the output of the SASPENCE system.

functional validation of the SASPENCE system in an early stage of its development, the most critical scenarios that were identified with the simulation study are therefore selected to be reproduced in the VeHIL laboratory. First we present the experimental setup for testing the SASPENCE system [72].

6.6.1 Experimental setup

As shown in Figure 6.7(a), the Fiat demonstrator vehicle is mounted on the chassis dyno to emulate the tire-road interaction and the moving base is used to emulate the preceding vehicle. The visual input to the camera system is emulated by projecting a real-time animation of the traffic scene in front of the camera, as shown in Figure 6.7(b). Similarly, DGPS satellite navigation and vehicle-to-vehicle communication are emulated by a real-time ethernet link from the real-time simulation environment.

Since the SASPENCE system is a driver warning system, a closed-loop configuration requires that a driver reacts to warnings and takes appropriate action. The prototype vehicle is therefore instrumented with a driving robot, consisting of two actuators to control the brake and throttle pedal positions. The driving robot is linked to the driver model with the characteristics of Table 6.2. The driver model receives real-time information on the host absolute state ${}^G\mathbf{x}_2$ and the relative target motion ${}^L\mathbf{x}_1$ from the simulation environment. The driver model also receives the warning level w from the SASPENCE system, such that it can calculate a desired speed v_{ref} , which is sent to the actuator controller of the driving robot. Hence, the experiment is a *closed-loop* hardware-in-the-loop simulation. The flow of information between these components is illustrated in the schematic diagram in Figure 6.8.

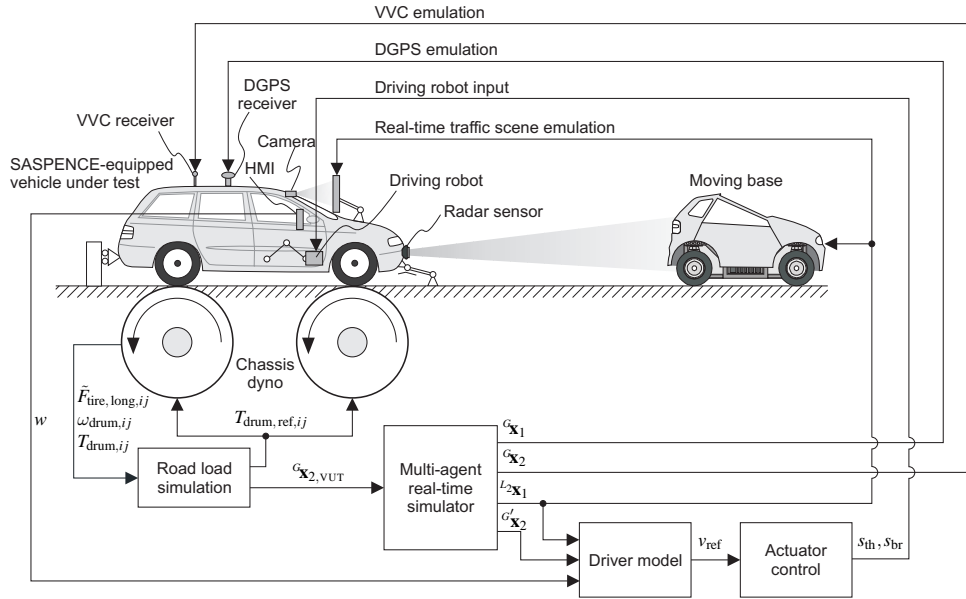


Figure 6.8: Schematic representation of the closed-loop VeHIL setup for the SASPENCE vehicle. The inputs from the driver, camera, DGPS, and VVC are emulated from the real-time simulation environment, whereas the relative motion for the radar sensor is emulated by the moving base.

6.6.2 Definition of an efficient test schedule

Using the results of the simulation study, a test schedule has been developed for a representative set of traffic scenarios. The test schedule is biased towards scenarios that are considered more critical, *i.e.*, with a lower value for the minimum TTC, as obtained from the simulation study. Correspondingly, scenario parameters are selected, such as the velocities $v_1(0)$ and $v_2(0)$, target acceleration a_1 , initial distance $x_r(0)$, and initial lateral offset $y_r(0)$, as summarized in Table 6.3. Figure 6.9 shows an overview of these parameters, where the SASPENCE vehicle, driving at a velocity v_2 , approaches a slower target vehicle, driving at v_1 . The resulting relative motion for the moving base is $v_{MB} = v_r = v_1 - v_2$. Around 150 tests were carried out, for each of which the performance criteria ρ were calculated. For comparison of the VeHIL experimental results with the PreScan simulation results, the approach scenarios (Scenario 5) were carried out both *with* and *without* the SASPENCE system.

6.6.3 Experimental results

Figure 6.10 illustrates these VeHIL test results for a typical approach scenario with the SASPENCE-equipped vehicle approaching a slower target vehicle. The initial velocities for this test are $v_{VUT}(0) = v_2 = 33.3$ m/s and $v_1(0) = 22.2$ m/s, and the initial distance $x_r(0) = 0$. It can be seen that at $t = 5.1$ s, the time-to-collision drops below $\gamma_{TTC} = 6$ s, which causes the reference algorithm to give a warning signal w_{ref} . However, only at $t = 7.5$ s the SASPENCE

Table 6.3: Test schedule with initial conditions for the SASPENCE system.

Scenario parameter	Symbol ^a	Unit	Min	Max ^b	Number of tests with SASPENCE system	
					on	off
<i>Scenario 3: Approach of target on adjacent lane, not relevant</i>					19	
Host vehicle velocity	v_2	m/s	33.3			
Target vehicle velocity	v_1	m/s	19.4	33.3		
Relative velocity	v_r	m/s	-13.9	0		
Acceleration profile target vehicle	a_1	m/s ²	-4	0		
Initial distance to target	x_r	m	50	150		
Initial lateral offset	y_r	m	-3.5	3.5		
<i>Scenario 4: Cut-in of target into host lane, not dangerous</i>					18	
Host vehicle velocity	v_2	m/s	22.2	33.3		
Target vehicle velocity	v_1	m/s	22.2	47.2		
Relative velocity	v_r	m/s	0	13.9		
Acceleration profile target vehicle	a_1	m/s ²	0	2		
Initial distance to target	x_r	m	10	50		
Initial lateral offset	y_r	m	-3.5	3.5		
<i>Scenario 5: Approach of target on host lane, potentially dangerous</i>					60	40
Host vehicle velocity	v_2	m/s	13.9	33.3		
Target vehicle velocity	v_1	m/s	0	30.6		
Relative velocity	v_r	m/s	-13.9	-2.8		
Acceleration profile target vehicle	a_1	m/s ²	0			
Initial distance to target	x_r	m	150			
Initial lateral offset	y_r	m	0			
<i>Scenario 6: Tailgating behind target vehicle, potentially dangerous</i>					10	
Host vehicle velocity	v_2	m/s	13.9	33.3		
Target vehicle velocity	v_1	m/s	13.9	33.3		
Relative velocity	v_r	m/s	0			
Acceleration profile target vehicle	a_1	m/s ²	0			
Initial distance to target	x_r	m	3	20		
Initial lateral offset	y_r	m	0			

^aSee Figure 6.9 for an illustration of the parameters.

^bWhen no maximum value is given, only the minimum value has been tested.

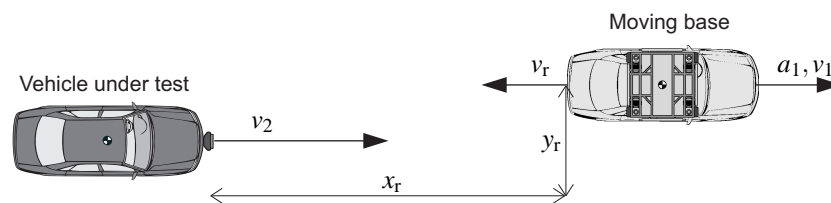


Figure 6.9: Overview of the scenario parameters in an approach scenario.

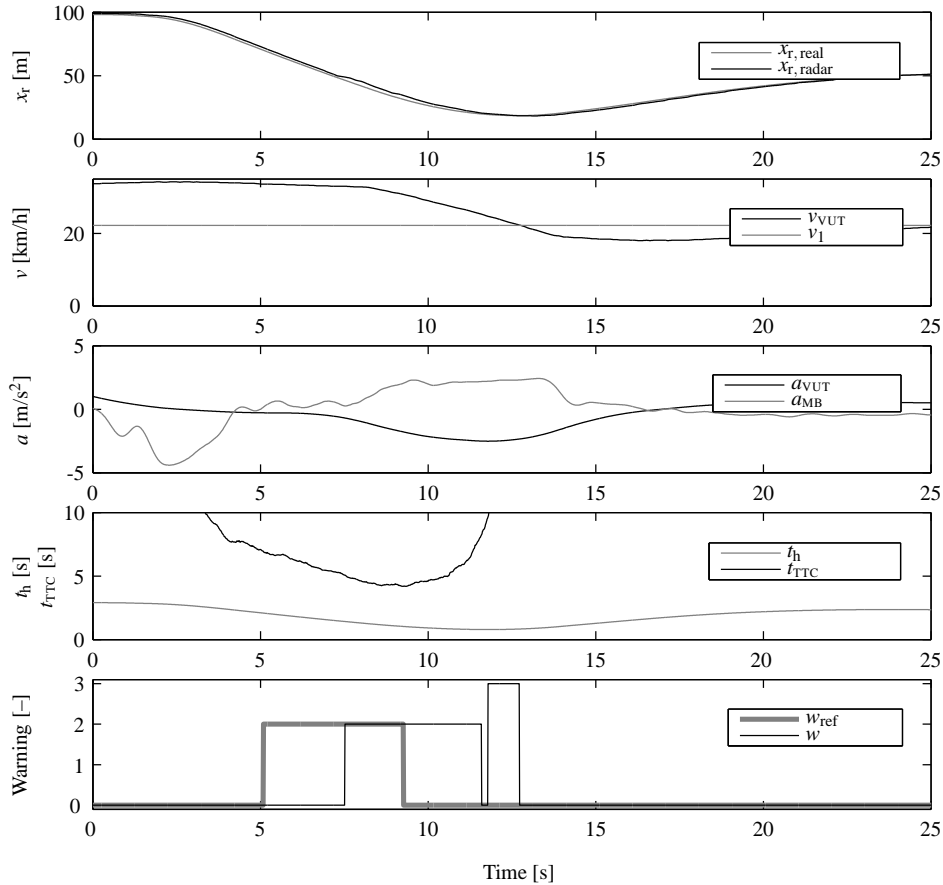


Figure 6.10: VeHIL test results for an approach scenario.

system gives a warning, after which the driving robot decelerates the vehicle according to the settings in Table 6.2. As the vehicle decelerates, the TTC rises again from $t = 9.3$ s onwards, indicating that a collision is being averted. Correspondingly, the reference signal w_{ref} disappears. However, the SASPENCE warning signal w remains present, and the warning level is even increased at $t = 11.8$ s, even though the distance x_r is increasing again.

Of course, the choice of the reference model is quite arbitrary, and a more conservative or a more sensitive algorithm might be selected. Nevertheless, the difference between the reference warning w_{ref} and the SASPENCE warning w is used to illustrate the dependability validation. Figure 6.11 shows the distribution of the timeliness of the warning. On average, the SASPENCE warning is given at approximately the same time instance as the reference warning, which indicates that the reference warning w_{ref} has acceptable behavior. However, the time difference ranges between 5 s too soon and 5 s too late. This variety in warning timeliness means that the presence or absence of a warning might be interpreted by the driver as a false or missed alarm.

If we look again at Figure 6.10, the lack of a SASPENCE warning in the time interval $t =$

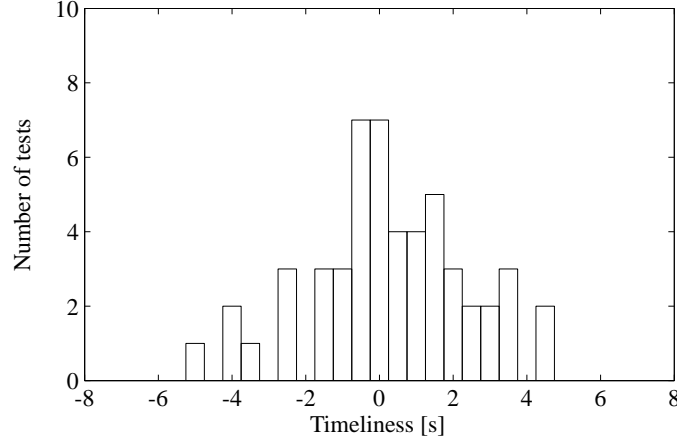


Figure 6.11: Distribution of the timeliness of the warning (negative means too soon, positive too late).

Table 6.4: Estimated reliability measures for the SASPENCE system, using Table 2.2.

Rate	Definition	Estimate
Accuracy p_{accuracy}	$(N_{\text{TN}} + N_{\text{TP}}) / (N_{\text{TN}} + N_{\text{FP}} + N_{\text{FN}} + N_{\text{TP}})$	0.982
Precision p_{CP}	$N_{\text{TP}} / (N_{\text{FP}} + N_{\text{TP}})$	0.994
True positive rate p_{TP}	$N_{\text{TP}} / (N_{\text{FN}} + N_{\text{TP}})$	0.979
False negative rate p_{FN}	$N_{\text{FN}} / (N_{\text{FN}} + N_{\text{TP}})$	0.021
True negative rate p_{TN}	$N_{\text{TN}} / (N_{\text{TN}} + N_{\text{FP}})$	0.989
False positive rate p_{FP}	$N_{\text{FP}} / (N_{\text{TN}} + N_{\text{FP}})$	0.011
Reliability p_{rel}	$\sqrt{N_{\text{TP}}^2 / (N_{\text{FP}} + N_{\text{TP}})(N_{\text{FN}} + N_{\text{TP}})}$	0.986

[5.1, 7.5] indicates a missed alarm, according to (6.4). Using (6.3) and a scenario duration of 30 s, the missed alarm indicator $\bar{J}_{\text{FN},j}(\varphi_j)$ for this scenario is calculated as 0.08. However, we should also take into account the importance sampling factor $\frac{f_{\mathcal{Q}}(\varphi_j)}{\xi_{\mathcal{Q}}(\varphi_j)}$ for this scenario j , as well as all other tested scenarios, such that the estimate for the missed alarm rate p_{FN} will be much less. Similarly, the time interval [9.3, 12.7] indicates a false alarm.

By combining all these test results and taking the importance sampling factor into account through (5.45), it is possible to obtain a representative overview of the dependability of the SASPENCE system. The results show that the estimated missed alarm rate is $\hat{p}_{\text{FN}} = 0.021$ and the estimated false alarm rate $\hat{p}_{\text{FP}} = 0.011$. The corresponding reliability p_{rel} (i.e., the geometric mean of the precision and the true positive rate p_{TP}) is estimated at 0.986. As summarized in Table 6.4, the reliability measures are all in the order of 10^{-2} .

These results show that in practice the dependability and timeliness of the SASPENCE warning function must be improved. This could also be observed during the VeHIL tests, where occasionally the SASPENCE system did not consider safety-critical scenarios threatening. Vice versa, warnings were sometimes given, even when the preceding vehicle did not pose any threat (it was far away or driving away from the vehicle under test). Because of the inherent trade-off between false and missed alarms for a given detection accuracy, it

Table 6.5: *VeHIL* test results for the minimum TTC [s] at different initial relative velocities $v_r(0)$ [m/s] for different drivers.

Driver type	SASPENCE	Initial relative velocity $v_r(0)$ [m/s]				
		-2.8	-5.6	-8.3	-11.1	-13.9
Conservative	off	*	7.35	7.22	2.89	4.76
	on	20.22	13.79	9.41	6.85	4.84
Intermediate	off	11.83	7.15	5.33	3.87	1.86
	on	16.00	9.77	7.21	5.18	3.87
Aggressive	off	12.17	2.01	3.40	4.18	*
	on	*	*	*	1.84	2.98

*Data not available.

Table 6.6: *VeHIL* test results for the RMS of the longitudinal acceleration [m/s^2] at different initial relative velocities $v_r(0)$ [m/s] for different drivers.

Driver type	SASPENCE	Initial relative velocity $v_r(0)$ [m/s]				
		-2.8	-5.6	-8.3	-11.1	-13.9
Conservative	off	*	0.64	0.87	0.71	0.88
	on	0.17	0.15	0.26	0.70	0.99
Intermediate	off	0.48	0.10	0.80	0.90	1.00
	on	0.11	0.17	0.26	0.69	1.05
Aggressive	off	0.24	0.66	0.64	0.97	*
	on	*	*	*	0.62	1.12

*Data not available.

is not possible to simply lower or raise the obstacle detection thresholds. Instead, the path prediction and reference maneuver algorithms should be further fine-tuned to reduce the above-mentioned probabilistic values.

Despite the fact that the dependability of the system must be improved, the effectiveness of SASPENCE in terms of traffic safety and driver comfort can still be validated for the scenarios with a correct alarm. For this purpose the experimental results are compared for scenarios *with* and *without* the SASPENCE system, as well as for different driver types (conservative, medium or aggressive), considering *equal* initial conditions for all six experiments.

In Table 6.5, the minimum TTC that occurs during an approach scenario is displayed for different driver types. From the table it can be concluded that the SASPENCE system has a positive effect on the safety of conservative and medium drivers, since the minimum TTC increases for them. Not enough consistent results for the aggressive driver were available to validate the benefit of the system for these drivers.

The effect that the SASPENCE system has on comfort is expressed in the performance measure $\|a_{\text{long},i}(t)\|_{\text{pow}}$. *VeHIL* test results are given in Table 6.6. This table shows that the RMS value of the acceleration generally decreases when using the SASPENCE system, which means that the SASPENCE system also increases driver comfort.

6.7 The role of test drives

Obviously, in the end outdoor test drives are still necessary to evaluate the system's performance on the road and to provide a subjective assessment of the SASPENCE system, *i.e.*, Step 9 of Algorithm 5.7. However, these tests can now be focussed on specific problem areas, since the system has already been thoroughly tested for a large number of scenarios in PreScan and VeHIL. These test drives can be used to evaluate the performance and dependability over a longer period of time. This will serve as validation of the expected probabilistic values from the simulation study and the VeHIL test results. In addition, test drives will be used to assign a subjective rating to the system and to test the HMI. It is a topic of ongoing research to carry out these test drives and compare the test results with those presented in this chapter [72].

6.8 Summary

This chapter has presented the system validation process of a driver information and warning system for safe speed and safe distance (SASPENCE). The system architecture and the experimental setup of the demonstrator vehicle have been described. A preliminary validation of the SASPENCE system has been carried out, according to the methodological framework that was presented in the previous chapter. Results of the VeHIL experiments show that the warning and intervention strategies need to be fine-tuned to further improve the dependability of the system.

Based on these findings, the scenario assessment modules of the SASPENCE system can be modified. Furthermore, ongoing research focusses on subjective evaluation of the SASPENCE system with test drives. It is expected that, despite the preliminary development stage of the SASPENCE system, these tests will show the benefit of the system in terms of traffic safety and driver comfort.

The effect of driver warning systems on traffic safety must always rely on an appropriate reaction by the driver, and much effort must therefore be placed on the design of the HMI. Instead of providing only a warning, a next step in longitudinal driver support might be to enforce a safe speed or safe distance, *e.g.*, intelligent speed adaptation [30]. In general, automation of the longitudinal control of a vehicle poses several issues that must be addressed in the design and validation of such systems. The next chapter therefore investigates a system with an automated longitudinal control function.



Figure 7.1: VeHIL laboratory setup: the Smart equipped with the CACC algorithm is following one of the moving bases.



Figure 7.2: Test drive set up at the ATP proving ground.

Chapter 7

Case study: Validation of a cooperative adaptive cruise control system

In the previous chapter a new driver *warning* system for safe speed and safe distance was presented and validated. Despite the occurrence of false and missed alarms, the system was shown to have a positive effect on driver comfort and traffic safety. As discussed in Chapter 2, a longitudinal vehicle *control* system, such as adaptive cruise control (ACC), has even more potential to increase safety and comfort, although it requires a much higher level of dependability. This chapter will therefore present the design and validation of a fault-tolerant control system for cooperative adaptive cruise control, based on the demonstrator vehicles and the sensor fusion strategy of Chapter 3.

Section 7.1 starts with a general approach to longitudinal vehicle control where gain scheduling is used to reproduce human driving behavior. A string stability analysis of the baseline ACC controller reveals several constraints on the control performance. Section 7.2 then presents a new cooperative control algorithm that further improves string stability, while increasing the tracking performance of the controller. The validation objectives and parameter set for this system are defined in Sections 7.3 and 7.4. According to the methodology of Chapter 5, Section 7.5 presents the results of a PreScan trend study. The corresponding VeHIL experiments to validate these results and investigate the sensitivity of the system are presented in Section 7.6. To complete the validation procedure, Section 7.7 presents results from test drives on a proving ground, which in turn are used to validate the simulations and VeHIL experiments. Finally, Section 7.8 concludes the chapter. The application of the methodology to this case study is summarized in Figure 7.3.

7.1 Introduction to longitudinal vehicle control

The main objective of automated longitudinal vehicle control is to provide comfortable, thus human-like, but also attentive driving behavior, as was discussed in Chapter 2. This is most apparent during car-following situations, where the host vehicle has to smoothly reach the same speed as the preceding vehicle at a desired safe distance. This section will

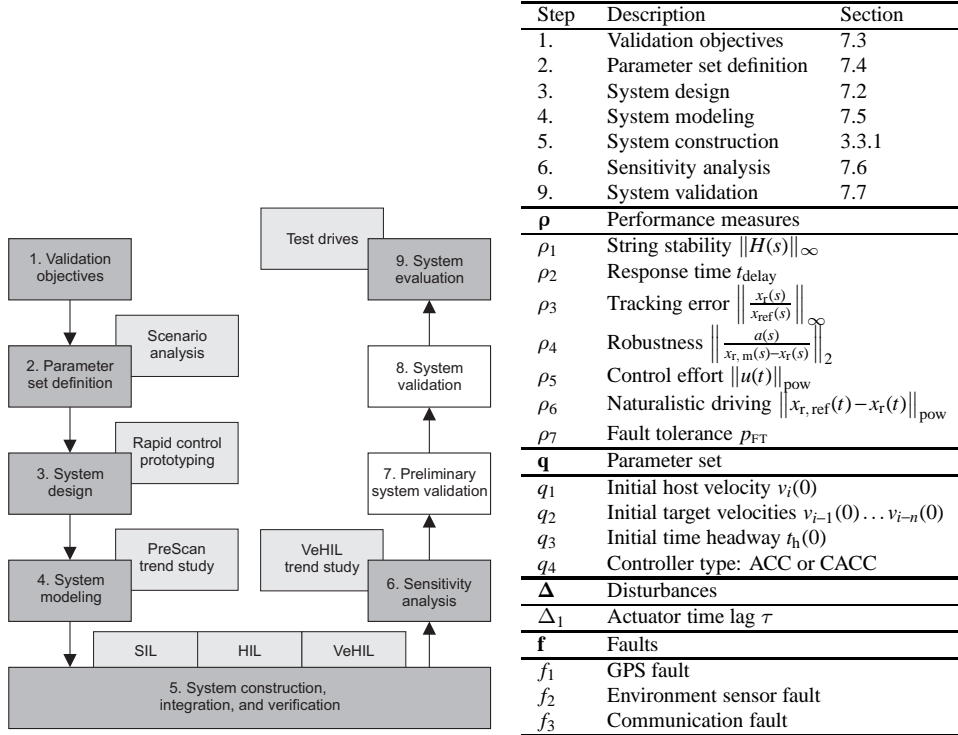


Figure 7.3: Steps in the methodology that are investigated for the CACC case study.

present a control algorithm for automatic car-following and investigate its string stability characteristics.

7.1.1 Gain scheduling for longitudinal vehicle control

In the ACC control architecture two cascaded levels are often distinguished, as was illustrated in Figure 2.5 on page 25. The outer loop (high-level controller) consists of the cooperative longitudinal controller that computes a reference acceleration a_{ref} based on the sensor information. The inner loop (low-level controller) consists of an acceleration controller that tracks the acceleration command a_{ref} from the outer loop as well as possible. The advantage of this configuration is that these two loops can be designed separately, reducing the overall complexity of the design. Furthermore, this structure can also be well motivated from the relation between driver and vehicle. The inner loop corresponds to the vehicle dynamics and the outer loop corresponds to the driving behavior.

First, let us first recall the ACC algorithm presented in (2.12), where the desired acceleration a_{ref} is given by feedback control of the spacing error $e_x = x_{\text{ref}} - x_r$ and speed tracking error $e_v = v_{\text{ref}} - v_r$:

$$a_{\text{ref}} = -K_2 e_v - K_1 e_x, \quad K_1, K_2 > 0. \quad (7.1)$$

This desired acceleration a_{ref} should control both e_x and v_r to zero. The performance of this basic feedback control law is shown in Figure 7.4(a) for a scenario that is composed of a cut-in, a car-following, and an emergency braking subscenario. In this scenario a target

vehicle 1 suddenly makes a cut-in maneuver in front of the host vehicle 2 at $t = 1$ s. The host then has to slow down to a car-following situation. Finally, the target makes an emergency braking maneuver to a lower velocity at $t = 15$ s. In car-following situations it would be uncomfortable for the driver when automatic braking would start immediately when the throttle is released. This undesirable behavior can be observed in Figure 7.4(a), where braking occurs instantly after release of the throttle, both during the cut-in and during the emergency braking maneuver.

In order to create a more comfortable car-following behavior and prevent frequent switching between control of the throttle pedal position s_{th} and brake pedal position s_{br} , the computation of the desired acceleration a_{ref} is treated separately for brake and throttle. The acceleration to be tracked by the brake system, denoted as $a_{ref, ACC, br}$, is always larger than that of the engine control $a_{ref, ACC, th}$, such that it takes some time before the brakes are activated after the throttle is released. Control law (7.1) is then split into

$$a_{ref, ACC, th} = K_2 v_r + K_1 (x_r - x_{ref, max}), \quad (7.2)$$

$$a_{ref, ACC, br} = K_2 v_r + K_1 (x_r - x_{ref, min}), \quad (7.3)$$

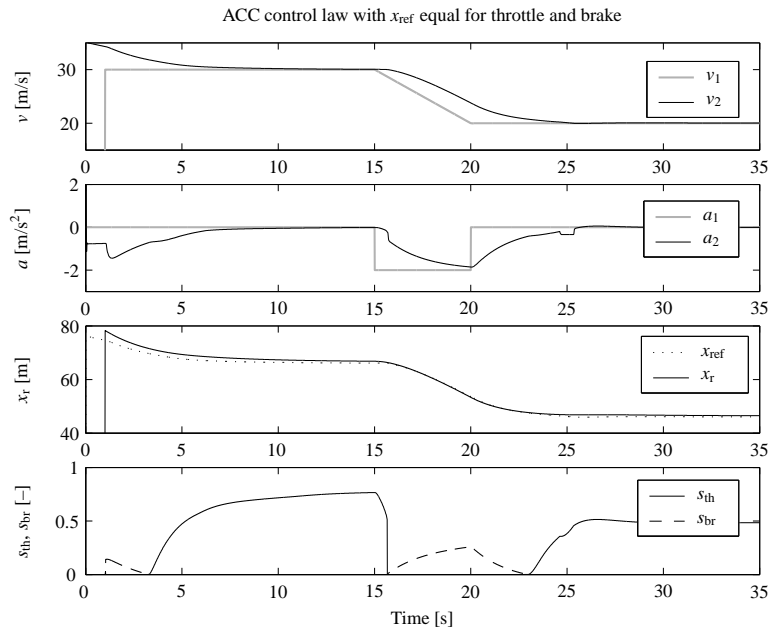
where the desired distance for throttle and brake control equal

$$x_{ref, max} = t_h v_2 + s_0, \quad (7.4)$$

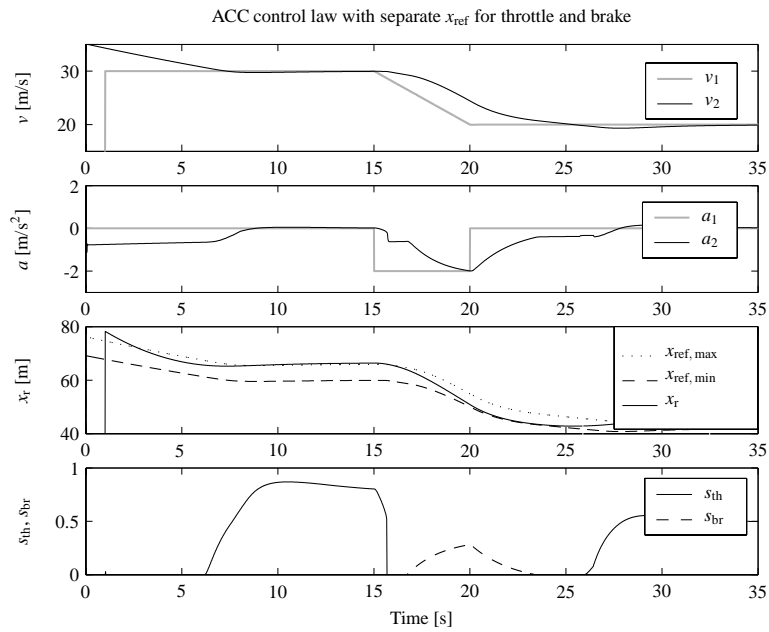
$$x_{ref, min} = (t_h - t_{h, diff}) v_2 + s_0, \quad (7.5)$$

with time headway t_h and safety margin s_0 . The parameter $t_{h, diff}$ reduces the desired time headway, effectively introducing a hysteresis between throttle and brake actuation, and making sure that they are never applied simultaneously. Figure 7.4(b) shows that the resulting longitudinal control is more comfortable. During the cut-in maneuver the controller only releases the throttle instead of braking, and during the emergency braking maneuver the controller applies the brakes more smoothly.

The selection of appropriate values for the feedback gains in (7.1) has been a heavily investigated research topic. Several authors (see *e.g.*, [150, 272]) describe the use of an optimal control procedure to select these gains, usually tested and fine-tuned with simulation studies. However, such a controller does not reflect human driving behavior, which is typically nonlinear and can hardly be described by an optimal control problem. Figure 7.5 demonstrates the tracking problem that occurs with constant values for K_1 and K_2 , when the distance x_r is significantly smaller or larger than the desired distance x_{ref} . In Figure 7.5(a) the host approaches the target from a large distance. However, because of the large distance separation e_x , the controller applies too much throttle, after which excessive braking is necessary to react to the large speed tracking error e_v . Conversely, Figure 7.5(b) depicts a situation where a target vehicle makes a cut-in with equal velocity, but at a short distance. The controller reacts to the negative spacing error e_x with an abrupt deceleration, after which the vehicle has to accelerate again to make up for the speed tracking error e_v . In both cases a mild use of the vehicle's actuators would have been more appropriate. A fixed setting for the feedback gains K_1 and K_2 should therefore not be considered for practical implementation. In practice, drivers look ahead and anticipate oncoming infrastructure (*e.g.*, traffic lights) and the behavior of other road users. In this chapter we will therefore use a gain scheduling approach to adjust K_1 and K_2 by nonlinear functions $K_1 = f_1(v_2, x_r, x_{ref})$ and $K_2 = f_2(v_2, x_{ref})$, similar to the approach followed by Yanakiev and Kanellakopoulos [269].

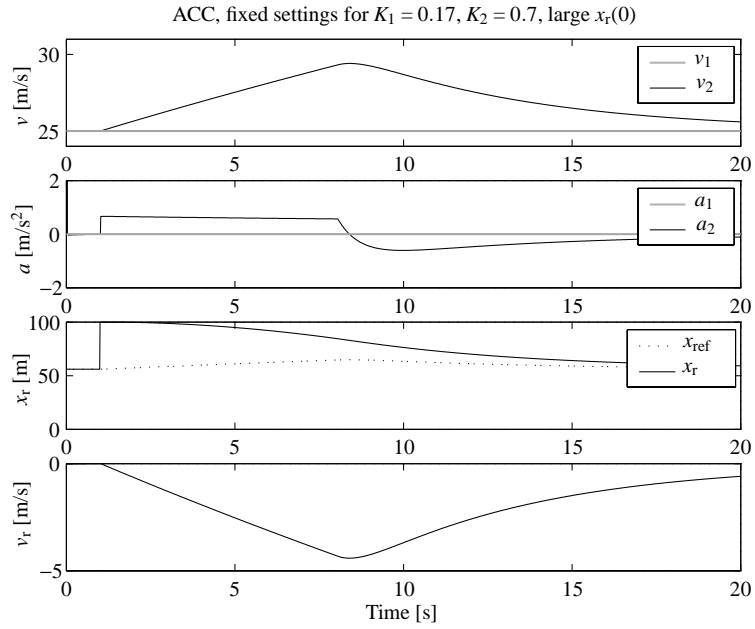


(a)

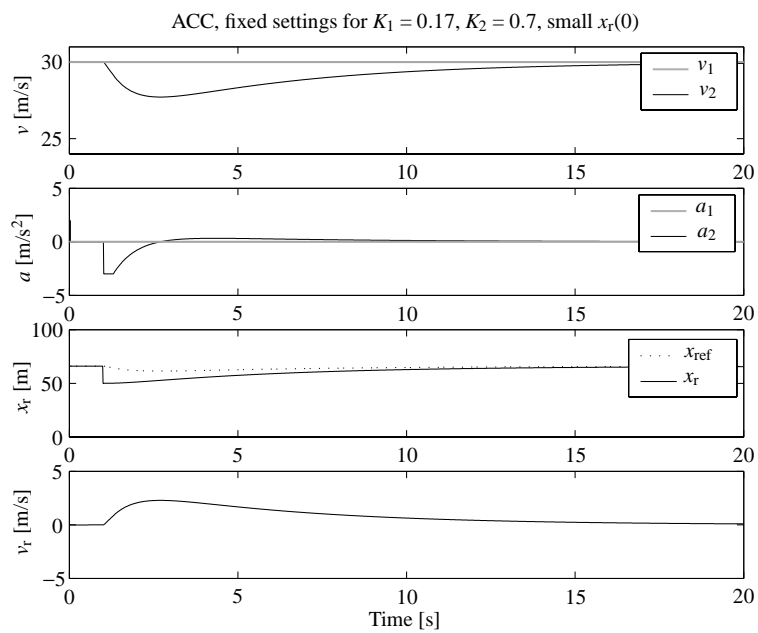


(b)

Figure 7.4: Actuator control during cut-in, car-following, and emergency braking scenarios: (a) using ACC control law (7.1); or (b) using control law (7.2)-(7.3). From top to bottom: velocity v_i ; acceleration a_i ; desired and actual distance x_{ref} , x_r ; and actuator use s_{th} , s_{br} .



(a)



(b)

Figure 7.5: Tracking of a target vehicle using constant values for K_1 and K_2 : (a) with large initial distance $x_r(0)$; (b) at a small distance $x_r(0)$.

The gain for the distance error K_1 is made adaptive by multiplication of two parameters:

$$K_1 = K_{1,v} K_{1,x}, \quad (7.6)$$

where $K_{1,v}$ depends on the vehicle speed v_2 and $K_{1,x}$ depends on the distance error e_x . In stop-and-go traffic at low speed, *e.g.*, in traffic jams, it is important to reach the desired distance with high accuracy, whereas at cruising speed a smooth following behavior is more important. Therefore, $K_{1,v}$ has a high value at low speeds and an asymptotically decreasing value at higher speeds. As shown in Figure 7.5, in case the actual distance x_r is significantly smaller or larger than the desired distance x_{ref} , the distance error feedback gain $K_{1,x}$ should be reduced to prevent too strong acceleration or deceleration. The relative velocity feedback gain K_2 has a high value for negative relative velocity (closing situations), which gradually decreases as the relative velocity increases. The value of the coefficients K_1 and K_2 (depending on v_2 , x_r , and x_{ref}) can be designed using curvatures in the range – range rate diagram [269]. For this study we have tuned these parameters with emphasis on adequate following behavior. Note that the objective of this case study is not the design and fine-tuning of the control algorithm itself, but that the controller is merely used to illustrate the validation methodology. We therefore accept that the performance may not be perfect, and instead focus on the ability to characterize and validate this (possibly imperfect) performance.

7.1.2 Control during transitional maneuvers

In addition to vehicle-following and speed control, the controller must perform a number of transitional maneuvers from one subscenario to another, similar to the transitions in the diagram of Figure 2.10 on page 39:

- *Cruise control to car-following.* This concerns ensuring a smooth transition from free-flow to steady-state car-following and vice versa. Every time a new target is detected, the desired acceleration a_{ref} is gradually blended from the old reference $a_{\text{ref,CC}}$ to the new reference value $a_{\text{ref,ACC}}$.
- *Maximum acceleration and deceleration.* Since an ACC system only intends to assist the driver during normal driving and not during emergency maneuvers, the desired acceleration a_{ref} is limited by a lower bound $a_{\text{ref,min}}$, usually -3 m/s^2 , and an upper bound $a_{\text{ref,max}}$, usually 1.5 m/s^2 . In case $a_{\text{ref}} < a_{\text{ref,min}}$, the driver is alerted by a visual warning display to intervene and avoid a collision.
- *Cut-in maneuver.* As we have seen in the previous section, it is undesirable for the vehicle to brake abruptly if another vehicle cuts-in at short distance, but with equal or higher relative velocity. Therefore, in case the preceding vehicle is driving away from the host vehicle ($v_r \geq 0$), $a_{\text{ref,ACC,br}}$ is set to zero and the host decelerates smoothly by only releasing the throttle.
- *Curve speed control.* To prevent uncomfortable cornering behavior the maximum lateral acceleration is limited by the parameter $a_{\text{lat,max}}$. The corresponding maximum longitudinal velocity $v_{\text{long,max}}$, given the vehicle's yaw rate $\dot{\psi}$, is then computed by:

$$v_{\text{long,max}} = \left| \frac{a_{\text{lat,max}}}{\dot{\psi}} \right|. \quad (7.7)$$

- *Target tracking in curves.* In addition, an environment sensor is not always able to track objects in sharp curves due to the limited azimuth angle of the sensor. The curve radius R_{curve} is estimated by

$$\hat{R}_{\text{curve}} = \left| \frac{v_{\text{long}}}{\dot{\psi}} \right|. \quad (7.8)$$

When the sensor loses a target in a curve with a radius $\hat{R}_{\text{curve}} < R_{\text{curve, min}}$, the controller is set to maintain the current vehicle speed v_{long} , in order to prevent potentially dangerous acceleration in the curve. This mode is disabled if the sensor detects a target vehicle again or the corner radius becomes larger than $R_{\text{curve, min}}$.

- *Stop-and-go control.* The control laws (7.2) and (7.3) are not adequate for stop-and-go situations, since the controller's objective is to exactly reach the desired headway. Uncomfortable braking behavior during stop-and-go situations may occur using (7.3), in case the host vehicle stops too early behind a stopped target vehicle. On standstill, control law (7.2) will cause the vehicle to shortly accelerate again to correct the small remaining distance error. Vice versa, use of (7.2) will also cause the vehicle to accelerate as soon as the vehicle in front accelerates. A transitional mode is therefore introduced to handle these situations, using a tolerance on the distance error e_x at standstill and a hysteresis on the response to acceleration of the preceding vehicle.

7.1.3 String stability considerations for longitudinal control

As discussed in Chapter 2, string stability is an important requirement for safety, performance, and comfort of longitudinal control systems. Although for adaptive feedback gains K_1 and K_2 , a linear stability analysis is not necessarily true, it may still be useful for many practical cases, where the distance errors from the operating point are small [269]. Using a similar approach as in Example 2.2 we will investigate the string stability for several types of controllers of the form (7.1). Implementation of the linearized controller for a steady-state condition leads to the spacing error transfer function from e_{i-1} to e_i :

$$H(s) = \frac{E_i(s)}{E_{i-1}(s)} = \frac{K_2 s + K_1}{s^2 + (K_2 + K_1 t_h) s + K_1}. \quad (7.9)$$

Analysis of the magnitude $H(j\omega)$ of this transfer function, and requiring this to be less than 1, reveals that, after simplification, we should have

$$(K_2 + K_1 t_h)^2 \omega^2 + (K_1 - \omega^2)^2 > K_2^2 \omega^2 + K_1^2. \quad (7.10)$$

After neglecting higher order terms, and assuming that $\omega > 0$, this leads to the following requirement to guarantee string stability:

$$K_2 > \frac{2 - K_1 t_h^2}{2 t_h}. \quad (7.11)$$

This implies that the time headway t_h should be large enough to guarantee string stability, since K_1 and K_2 are limited for practical reasons. Figure 7.6 illustrates this requirement, where plot (a) shows string-stable behavior for $t_h = 2$ s and (b) string-unstable behavior for $t_h = 0.5$ s.

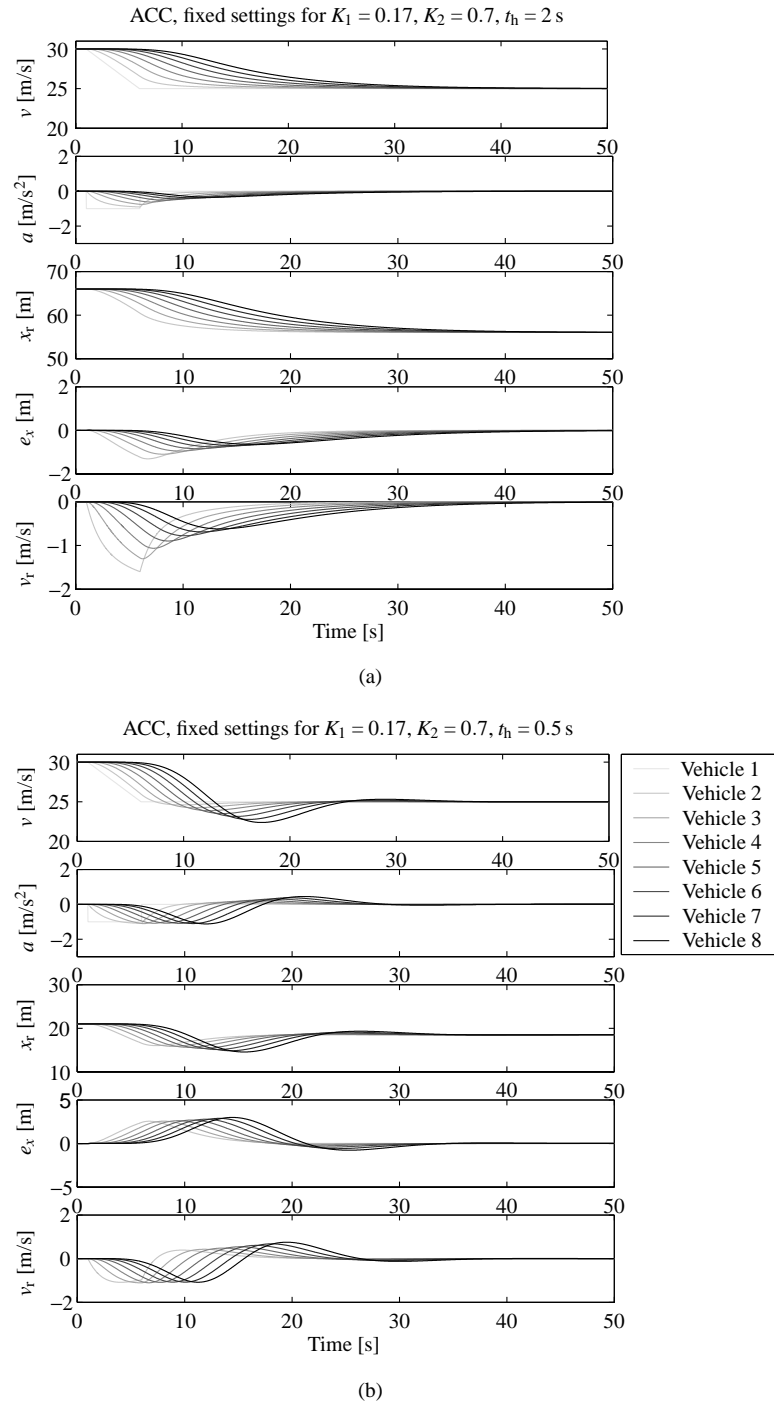


Figure 7.6: String stability analysis for ACC control law (7.1) with (a) $t_h = 2$ s (string-stable), and (b) $t_h = 0.5$ s (string-unstable).

Unfortunately, in practice, the actuator dynamics of vehicle i exhibit a time lag τ in the realization of a_{ref} , such that

$$\tau \dot{a}_i + a_i = a_{\text{ref},i}. \quad (7.12)$$

The corresponding string stability transfer function then becomes:

$$H(s) = \frac{K_2 s + K_1}{\tau s^3 + s^2 + (K_2 + K_1 t_h) s + K_1}. \quad (7.13)$$

It can be shown [197] that the magnitude $H(j\omega)$ is always less than or equal to unity at all frequencies if and only if:

$$t_h \geq 2\tau. \quad (7.14)$$

However, in practice the inherent time delay in acquiring and processing the sensor signals decreases the string stability even further [153]. Using a time lag on the realized acceleration, as well as a time delay t_{delay} in the sensor processing, we will derive the transfer function for the spacing error propagation between two vehicles. Using $e^{-t_{\text{delay}}s} X(s)$ as the Laplace transform for a time delayed signal $x(t - t_{\text{delay}})$, this leads to the transfer function

$$H(s) = 1 + \frac{-t_h \tau s^3 - t_h s^2 - t_h^2 K_1 s}{t_h \tau s^3 + t_h s^2 + (t_h K_2 e^{-t_{\text{delay}}s} - t_h t_{\text{delay}} K_1 + t_h^2 K_1) s + K_1 t_h}. \quad (7.15)$$

Requiring $|H(j\omega)|$ to be smaller than 1, leads to the necessary condition:

$$t_{\text{delay}} < \frac{t_h^2 K_1 + 2t_h K_2 - 2}{2t_h K_1}. \quad (7.16)$$

This means that for small values of the desired headway time t_h , any time delay t_{delay} will further decrease the capability of any ACC algorithm of the type (7.1) to provide string-stable car-following behavior.

Furthermore, for large errors resulting from challenging maneuvers, a linear stability analysis will not be valid anymore. In that case the control configuration may indeed become string-unstable, even if the conditions (7.11), (7.14), and (7.16) are met. Since we would like to validate the overall level of string stability, and to identify the stability boundaries in the parameter set, an analytical solution using linear stability analysis is not possible.

7.2 Cooperative vehicle control

Although the ACC control system described in the previous section is able to provide a satisfactory performance in common driving situations, one of the main drawbacks is that current environment sensors have a maximum range of around 200 m, which is insufficient to detect traffic jams or other potential danger further ahead. In addition, the controller only reacts to the directly preceding vehicle that is sensed by the environment sensor, whereas potential targets may be blocked by road infrastructure or other vehicles. Furthermore, the sensor can only distinguish a maximum amount of vehicles, and often returns unreliable signals to the controller, due to multi-path reflections, weather conditions, and sensor noise. Moreover, safety considerations and the need for string stability imposes restrictions on the time headway maintained by the ACC, as was shown in the previous section. Usually, the minimum possible time headway of ACC is around 1.5 s. However, it can be shown that ACC with such a large time headway would have a negative impact on traffic throughput with future higher market penetration levels [247].

7.2.1 Added value of vehicle-to-vehicle communication

ACC could therefore be greatly enhanced when the field-of-view of the sensorial platform is extended to include information from other preceding vehicles. This can be achieved by implementation of vehicle-to-vehicle communication (VVC), as was demonstrated in Chapter 3. Several authors (see *e.g.*, [87, 154]) have therefore investigated an extension of ACC systems to cooperative adaptive cruise control (CACC) systems. In CACC systems the inter-vehicle distance is accurately estimated using VVC and environment sensors. The advantage of CACC, compared to ACC, is that it has an increased control bandwidth and dependability. This improves string stability, which in turn has a positive effect on road capacity and traffic safety.

Furthermore, CACC systems are expected to improve traffic flow and increase road capacity, since the increased control bandwidth and string stability allows to maintain a time headway of only 0.5 s. However, more detailed simulations by Van Arem *et al.* [9] show that the extent of this improvement depends heavily on the traffic-flow conditions of the specific road section and the penetration level of CACC. Practical applications of CACC have only been tested in limited demonstrator setups using two vehicles and current control algorithms do not take into account traffic disturbances beyond the directly preceding vehicle. CACC systems addressing multiple vehicles further ahead have not been studied in depth, due to the high necessary penetration level of vehicles equipped with VVC. The objective of this chapter is therefore to develop and demonstrate an experimental CACC system, which combines the ACC function with a cooperative system that looks multiple vehicles ahead. This system is based on the fault-tolerant estimation scheme for the host vehicle state and the relative motion that was developed in Chapter 3. This facilitates accurate and reliable car-following with a time headway of only 0.5 s.

7.2.2 An algorithm for cooperative adaptive cruise control

A control law for CACC can be designed similar to (7.2)-(7.3). However, the main advantage of cooperative control is that more information is available, such as the acceleration of the preceding vehicle. Using VVC, the acceleration of the preceding vehicle $i-1$ (which is difficult to estimate with only an environment sensor) can be communicated to the host vehicle i . With information on the acceleration a_{i-1} , as well as more reliable estimates for the range and range rate, the ACC control law (7.2)-(7.3) can be modified to

$$a_{\text{ref,CACC,th},i} = K_3 a_{i-1} + K_2 v_{r,i} + K_1 (x_{r,i} - x_{\text{ref,max},i}), \quad (7.17)$$

$$a_{\text{ref,CACC,br},i} = K_3 a_{i-1} + K_2 v_{r,i} + K_1 (x_{r,i} - x_{\text{ref,min},i}), \quad (7.18)$$

$$(7.19)$$

where again K_1 , K_2 , and K_3 are positive adaptive feedback gains $K_1 = f_1(v_i, x_{r,i}, x_{\text{ref},i})$, $K_2 = f_2(v_i, x_{\text{ref},i})$, and $K_3 = f_3(v_i, x_{\text{ref},i})$. The availability of an acceleration signal in the feedback control law provides an opportunity to react faster to emergency braking of a preceding vehicle. In addition, the higher reliability of the signal allows to maintain a shorter time headway to preceding vehicles, as well as a string-stable performance.

In Chapter 2 it was argued that human drivers exhibit multi-anticipative car-following behavior with respect to n preceding vehicles, usually with n equal to 2 or 3, see (2.35). In order to use (7.17)-(7.18) with respect to multiple preceding vehicles, we include multi-anticipative behavior of drivers by considering the necessary deceleration a_{ref} with respect

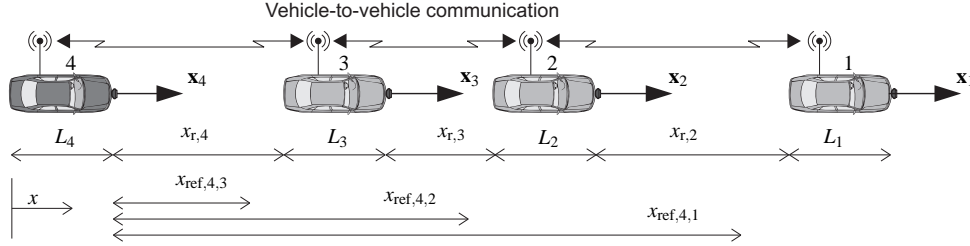


Figure 7.7: Configuration of four CACC-equipped vehicles in a car-following scenario.

to all n preceding vehicles, as illustrated in in Figure 7.7. The idea is that the host vehicle i should not only keep a distance $x_{\text{ref},i,i-1}$ to vehicle $i-1$, but also a distance $x_{\text{ref},i,i-j}$ to the other j target vehicles:

$$x_{\text{ref},i,i-j} = t_h v_i + j s_0 + (j-1)L_j. \quad (7.20)$$

This equation states that the desired distance to preceding vehicles is equal to the conventional headway $t_h v_2$ plus a safety margin s_0 and vehicle length L_j for each vehicle j further ahead.

Correspondingly, the CACC system computes a desired acceleration $a_{\text{ref},i-j}$ with respect to each preceding vehicle j , according to (7.20). The desired acceleration a_{ref} to be tracked by the vehicle's lower-level controller is then calculated by taking the minimum of all $a_{\text{ref},i-j}$ for all j preceding vehicles

$$a_{\text{ref}} = \min(a_{\text{ref},i-1}, \dots, a_{\text{ref},1}). \quad (7.21)$$

7.2.3 Hybrid automaton for CACC

In order to combine the various controllers presented in this section, we implement them in a hybrid automaton, as depicted in Figure 7.8. This allows to design and fine-tune each controller for a specific mode of operation. The switching logic is included in a supervisor, which also includes a fault management system to detect and isolate faults, as was presented in Chapter 3. In this way, graceful degradation of control functions in the presence of a fault in the environment sensor can be distinguished. For example, when the communication fails ($f_{\text{comm}} = 1$), the cooperative control algorithm (7.17)-(7.21) can degrade to a conventional ACC using (7.2)-(7.5). Vice versa, when the environment sensor fails ($f_{\text{lidar}} = 1$), the system can use the communication to obtain a pseudo-range measurement and use (7.17)-(7.21), though with a larger time headway t_h to accommodate the less accurate inter-vehicle motion estimation.

The setpoint for $a_{\text{ref,ACC,th}}$ and $a_{\text{ref,ACC,br}}$ that is given by the active mode of Figure 7.8 is fed to the actuator control of the vehicle. For the deceleration control by the brakes and the acceleration control by the throttle, an adaptive feed forward algorithm is used, based on an approximation of the inverse vehicle dynamics, including engine, brake, and gearbox dynamics. In addition, feedback of the acceleration signal and a simple PI controller realize the desired acceleration a_{ref} accurately. The supervisor switches between engine braking and additional braking to prevent simultaneous activation of throttle and brake. Furthermore, any additional driver input at the throttle pedal ($s_{\text{th}} > 0$) or brake pedal ($s_{\text{br}} > 0$) should always overrule the control commands generated by the control algorithms. Therefore, the

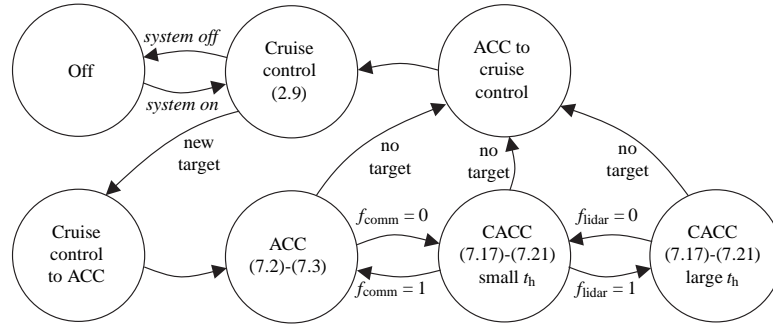


Figure 7.8: Hybrid automaton with the modes and transitions between cruise control, adaptive cruise control (ACC), and cooperative adaptive cruise control (CACC).

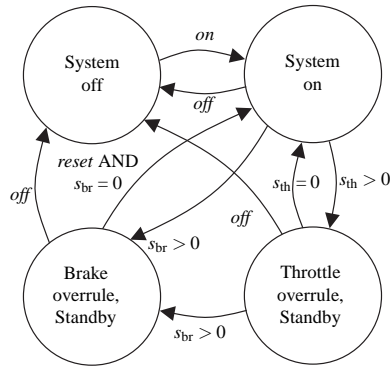


Figure 7.9: Hybrid automaton for the lower level control, depending on the driver input.

supervisor also includes a hybrid automaton for the lower-level controller, as depicted in Figure 7.9. This system also handles driver inputs that are given through a control lever: The *on* switch activates on the CACC system and sets the current velocity as the desired cruise control velocity $v_{\text{ref,CC}}$. Conversely, the *off* switch disables the system. When the system is on, the *set* and *reset* switches allow the driver to increase and decrease $v_{\text{ref,CC}}$, respectively. When the system is in standby (due to a temporary throttle or brake overrule condition), the *set* and *reset* can be used to set a new cruise control velocity or resume the old reference $v_{\text{ref,CC}}$, respectively.

The integration of several controllers into a switched system creates a very complex system. Even if all control algorithms separately provide a stable behavior, stability of the switched system cannot be guaranteed [200]. The control system therefore needs to be validated not only for the separate states, but especially for the switching between those states. Since stability of a complex hybrid system is difficult to prove in practice, string stability of the switched system will be validated using the randomized approach of Algorithm 5.7.

Table 7.1: Car-following scenario parameters for validation of the CACC algorithm.

Parameter	Unit	Distribution	μ	$\sigma, \varsigma, \mathbf{S}$	Min.	Max.
$v_r(0)$	m/s	Normal	-0.25	1.68	-10	10
$\lambda_v = v_{\text{ref}}/v_0$	-	Laplace	1.03	0.09	0.8	1.2
$\log(x_r(0))$	m	Bivariate normal	3.36	$\begin{bmatrix} 0.37 & 1.18 \\ 1.18 & 14.37 \end{bmatrix}$	1	5
v_0	m/s		29.03		20	40
a_{max}	m/s ²	Log-normal	0.44	0.17	0.5	4
t_{reac}	s	Log-normal	-0.31	0.16	0.1	3

7.3 Definition of the validation objectives

The main validation objective for CACC is to test whether the system provides string-stable behavior, which is beneficial for both driver comfort and safety. In addition, fault tolerance is validated by analyzing the extent to which the supervisor is able to provide graceful degradation of control functions in the presence of sensor or communication faults. The driving comfort will be evaluated by comparing the controller response to a human driver, which is modeled according to the Gipps model (2.36) of page 30.

With respect to the dependability of the system, we require the fault tolerance of the system p_{FT} to be at least 0.999, *i.e.*, the desired maximum probability for failure p is 10^{-3} . Since dependability is a stringent requirement, and the real probability p is most probably a very low value, we would like to validate these measures with a maximum error of $\pm 10\%$, which means that the relative accuracy is set at $\epsilon_r = 0.1$. Usually a 99% confidence interval is required, such that we set the confidence parameter δ at 0.01.

7.4 Definition of the parameter set

These validation criteria should be evaluated for a representative set of operating conditions. Since CACC is concerned with longitudinal vehicle control, we consider single-lane traffic with the statistical distribution and scenario parameters, as was investigated in Chapter 2. This concerns free-flow, car-following, cut-in, cut-out, approach, separate, and lane-change subscenarios. Table 7.1 summarizes the scenario parameters of the car-following subscenario for the sensitivity analysis (Steps 4 and 6 in Algorithm 5.7).

Disturbances in terms of model uncertainty Δ are taken into account, where the relevant uncertainties Δ_i are related to the vehicle mass, the temporal behavior of the drivetrain, and the tire dynamics. This uncertainty is grouped in the uncertainty on the actuator time lag τ . Additional disturbances caused by measurement noise are taken into account by adequate sensor modeling. Similarly, in Section 2.5 we have identified several failure modes in the environment sensor, GPS, and VVC, which are modeled additively as complete, but intermittent, failures.

7.5 PreScan trend study

7.5.1 Modeling of cooperative vehicle control

The control system that was presented in Section 7.2 is designed using MATLAB/SIMULINK with blocksets from the MATLAB toolbox CACCLib, which was developed within the framework of this thesis to support the design of ADAS controllers. The supervisor that was depicted in Figures 7.8 and 7.9 is implemented using MATLAB/STATEFLOW. The CACC algorithm uses a time headway $t_h = 0.5$ s. The integrated model is implemented in the modeling environment PreScan, which also supplies sensor models for the lidar sensor, vehicle dynamics models from ADVANCE, and modules for simulation of GPS and VVC. The traffic environment is animated using PreScan's 3D visualization software.

7.5.2 Definition of a test schedule for sensitivity analysis

Following the methodology of Chapter 5, the sensitivity of the performance measures to various combinations of scenario parameters is first assessed by performing an initial simulation study, based on design of experiments theory. For this purpose a fractional factorial design is constructed using the MATLAB Statistics toolbox, using the minimum, mean, and maximum values from Table 7.1. A PreScan simulation is run for each of the selected points.

7.5.3 Simulation results

The sensitivity analysis was carried out with 280 simulated single-lane scenarios, by which a preliminary analysis of the performance of the CACC algorithm can be done. Here we restrict the discussion of the simulation results to the analysis of the effect of the disturbance τ on the performance of the CACC algorithm. The reader is referred to [241] for more information on the simulation setup and the simulation results. Figure 7.10(a) shows the performance of the CACC algorithm for a car-following and slow-down maneuver of four vehicles equipped with CACC, each with a time headway of 0.5 s and with $\tau = \Delta_1^- = 0.1$ s. Compared to the similar scenario of Figure 7.6(b), it can be seen that the feedback of the acceleration signal has a positive impact on string stability. As soon as the lead vehicle makes an emergency braking maneuver, all following vehicles react. Figure 7.10(b) shows the same scenario with $\tau = \Delta_1^0 = 0.5$ s. It can be seen that the behavior is more or less similar.

Next, the effect of the minimum, mean, and maximum (not shown in figure) value for this disturbances on the CACC performance can be investigated. This is often carried out by the analysis of variance (often abbreviated as ANOVA), which reflects the statistical significance of the sensitivity of each parameter. Based on the analysis of variance it can be determined which scenario parameters, disturbances, and failure modes have a statistically significant effect on the performance measures. More importantly, the analysis of variance can reveal interaction between the parameters. This sensitivity analysis is the topic of ongoing research, but considering the preliminary results it is concluded that this model uncertainty can be neglected. Furthermore, it was shown that measurement noise and sensor faults did not influence the simulation results. This suggests that the fault management system of Chapter 3 performs adequately, although this should still be validated in practice.

Based on the sensitivity analysis, the dimension of the parameter set \mathcal{Q} can be reduced by leaving out parameters or parameter combinations that are not considered to be relevant.

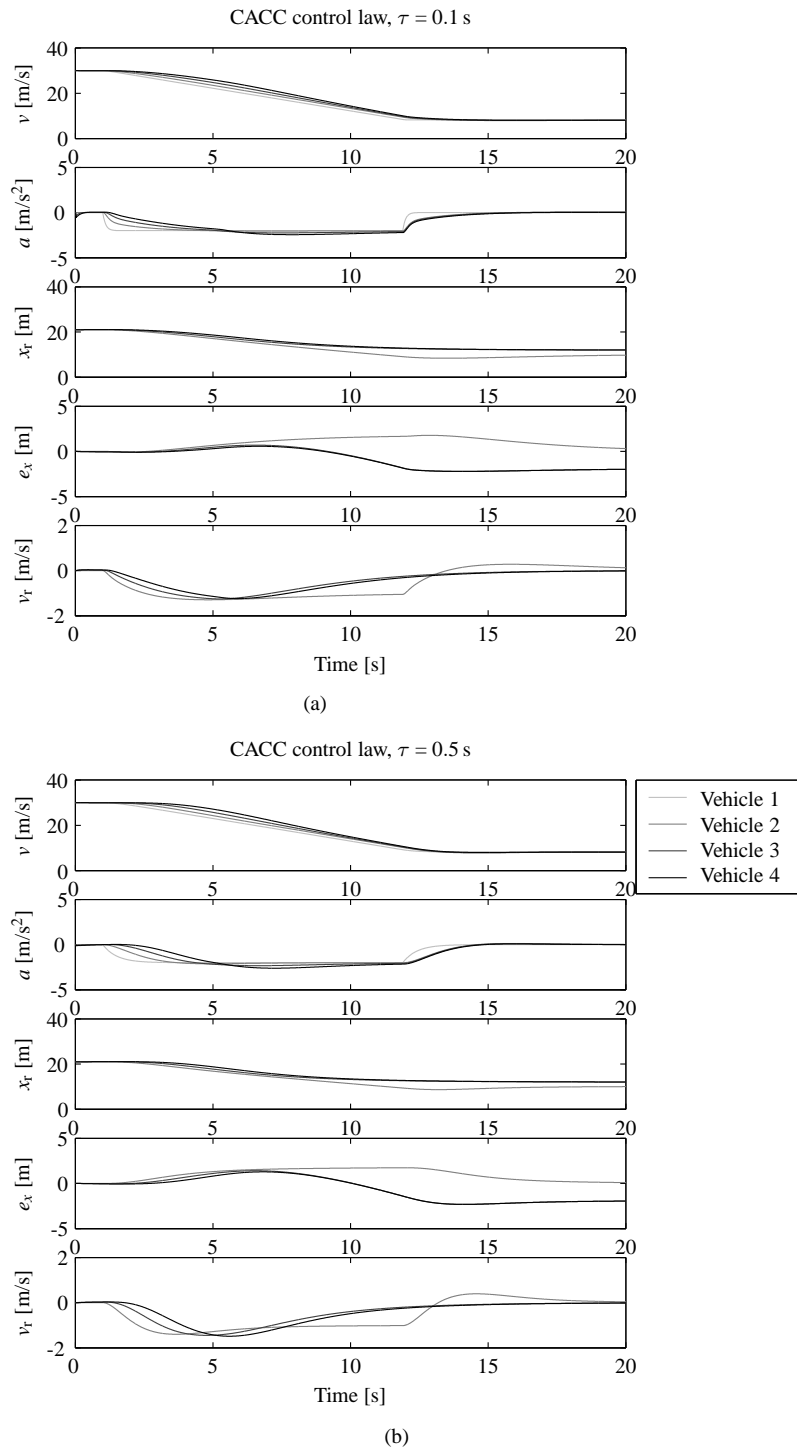


Figure 7.10: PreScan simulation results for the CACC control law (7.17)-(7.21) with (a) $\tau = 0.1$ s, and (b) $\tau = 0.5$ s.

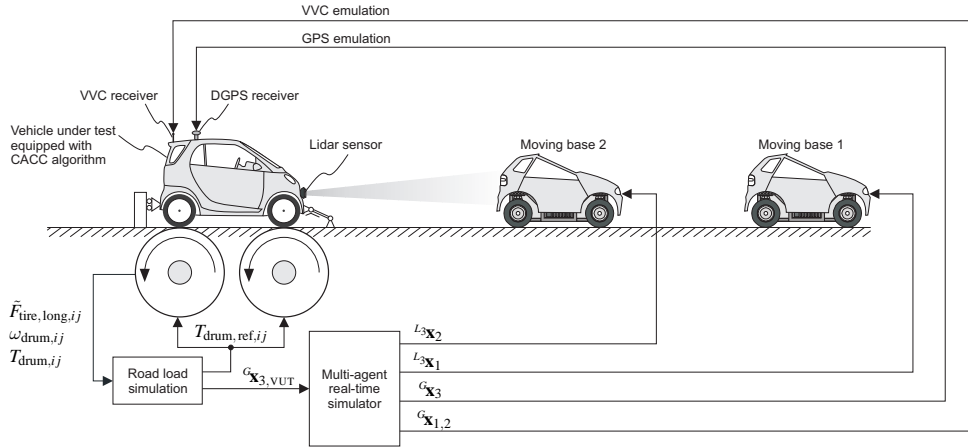


Figure 7.11: Schematic representation of the closed-loop VeHIL setup for the CACC system. The inputs from the GPS and VVC are emulated from the real-time simulation environment, whereas the relative motion for the lidar sensor is emulated by the moving bases.

Furthermore, irrelevant subspaces of Q can be neglected, which further decreases the number of possible interactions between different components q_i in the experimental design. These preliminary results can also be used to construct a response surface that serves as the importance sampling PDF for the first sequence of the simulation study.

7.6 Sensitivity analysis with VeHIL experiments

The experimental setup of the CACC tests in VeHIL is shown in Figure 7.1 on page 158, where one of the Smart vehicles is mounted on the chassis dyno. The surrounding traffic environment is emulated by the two moving bases, and can be detected by the on-board lidar sensor. Other vehicles are only present in the simulation environment, and their state $G\mathbf{x}$ is available to the host vehicle by VVC. This communication between the host vehicle and each of the target vehicles, as well as GPS information, is emulated by a wired connection between the host and the traffic simulation, as schematically shown in Figure 7.11.

A suitable VeHIL schedule is defined by the relevant points in the fractional factorial design. The results can then be compared to those of the simulations, such that the model uncertainty can be reduced, according to the methodology that was presented in Chapter 5.

Figure 7.12 gives an example of these VeHIL tests, where the CACC-equipped Smart demonstrator vehicle is driving on the chassis dyno at 80 km/h. It is following three preceding vehicles, two of which are represented by moving bases and one emulated by the multi-agent real-time simulator. Suddenly the lead vehicle makes an emergency brake maneuver due to an oncoming traffic jam and slows down to 30 km/h. As can be seen from the test results, the host vehicle follows quickly and also slows down to 30 km/h. Note that the velocity drop from 80 km/h to 30 km/h was chosen to accommodate the maximum moving base velocity of 50 km/h. When a stable string of vehicles has been reached at 30 km/h, the lead vehicle stops, and the test is finished.

These type of VeHIL experiments are carried out for a limited number of parameter

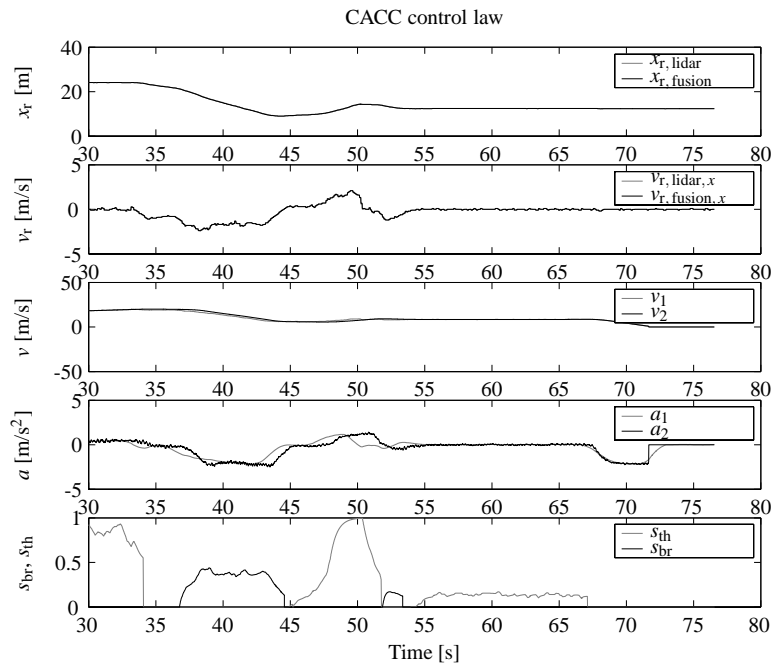


Figure 7.12: Test results for an approach scenario in VeHIL with the CACC system.

variations to identify the sensitivity of the CACC system to various disturbances when using the actual hardware. Similar to the PreScan simulations, analysis of these VeHIL tests are the topic of ongoing research. Nevertheless, using preliminary results from these VEHIL experiments insight in the behavior of the controller can be obtained. Subsequently, the control feedback gains can be fine-tuned to obtain an adequate and safe following behavior. In the next validation phase this controller is tested with test drives, as discussed next.

7.7 Test drives with the CACC system

The CACC system is tested with four vehicles on the ATP proving ground in Papenburg, Germany. Each vehicle i is equipped with satellite navigation, VVC, and the fault-tolerant estimation scheme of Chapter 3 to estimate its vehicle state ${}^G\mathbf{x}_i$ and the relative motion ${}^L\mathbf{x}_j$ of other vehicles $j \neq i$. Furthermore, all vehicles were equipped with the CACC algorithm (7.17)-(7.21), except for the manually driven lead vehicle 1. A photograph of the test setup is shown in Figure 7.2 on page 158.

The test drives are meant to confirm and improve the results from the PreScan simulations and VeHIL experiments that were carried out earlier. The test schedule is therefore defined according to the critical points in the fractional factorial design, complemented with several other interesting scenarios that appeared to be necessary for sensitivity analysis. The tests were executed on the high speed oval of the proving ground, where each test run included approach, car-following, cut-in, cut-out, lane-change, and emergency braking maneuvers. To validate the added value of CACC over conventional ACC, these tests were

carried both with the ACC algorithm (7.2)-(7.5) and the CACC algorithm (7.17)-(7.21).

For the ease of comparison with the VeHIL and PreScan results, we present the test drive results for the car-following and emergency braking maneuver from 80 km/h to standstill. The evaluation is based on the performance measures mentioned in Figure 7.3. Figure 7.13(a) shows the test results for an emergency brake test using the ACC control algorithm (7.2)-(7.3) in every vehicle. The most important observation is that the ACC controller is not string-stable. As can be seen in Figure 7.13(a), the spacing error e_x between subsequent vehicles decreases to the rear of the string, similar to the situation in Figure 7.6(b). Near the end of maneuver, vehicle 3 even has to make a manual evasive maneuver to avoid an actual collision. Furthermore, the deceleration levels result in uncomfortable behavior.

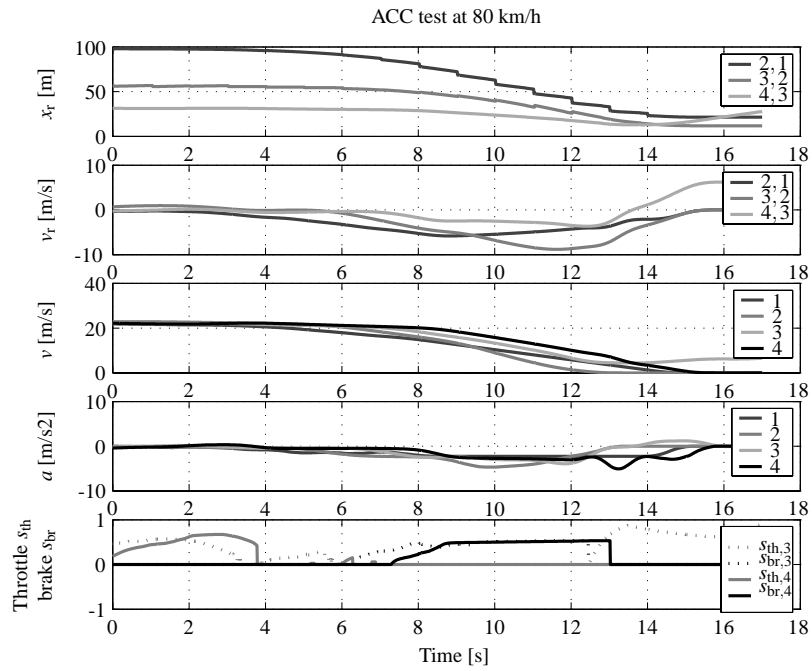
Similarly, the results for the same scenario with the CACC algorithm (7.21) are shown in Figure 7.13(b). Because of the direct communication between all vehicles in the string, the vehicles react earlier on disturbances. Subsequently, the response time for each vehicle in the string is significantly shorter, as can be seen from the actuator control in the bottom subplots of both figures. This results in significantly increased string stability as can be seen from the increased spacing, similar to Figure 7.10. Similarly, the control effort in terms of the RMS value of the actuator pedal positions is lower for the CACC algorithm than for the ACC algorithm. Furthermore, compared to the ACC controller, the RMS value of the acceleration has decreased, resulting in improved comfort.

During the test drives the fault-tolerant state estimation system proved to work adequate with respect to the vehicle state estimation of each vehicle. The sensor fusion between environment sensing and information from vehicle-to-vehicle communication also worked properly, but occasionally false and missed alarms did occur, due to the preliminary settings of the sensor fusion system. The required fault-tolerance of 0.999 was therefore not obtained, but ongoing research is focussed on fine-tuning the sensor fusion system, and subsequent improvement of the dependability.

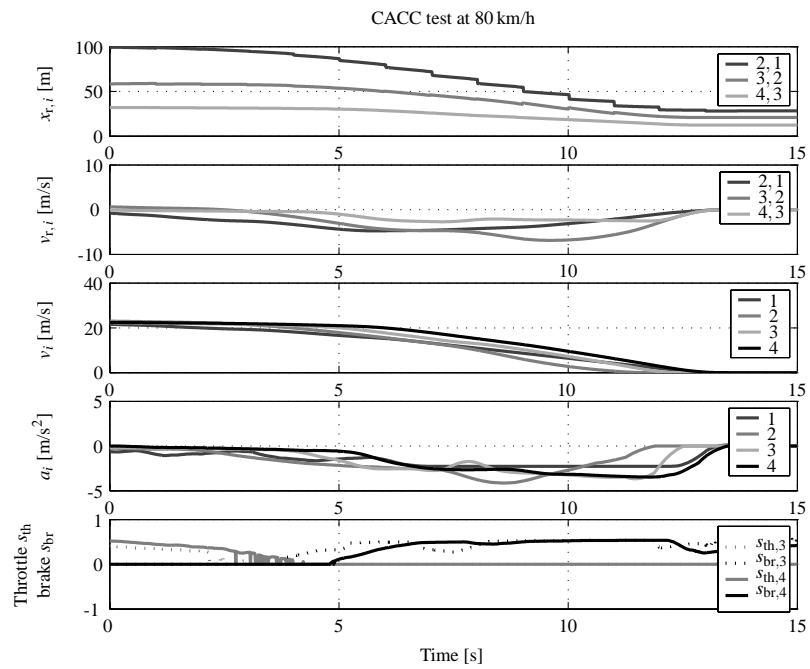
7.8 Summary

This chapter has presented and demonstrated a cooperative adaptive cruise control (CACC) system. This system requires state estimation of individual vehicles, combined with vehicle-to-vehicle communication and an environment sensor. This system is able to extend the field-of-view beyond the directly preceding vehicle and provides increased performance, safety, and string stability with respect to conventional ACC algorithms. In addition, the redundancy in state estimation enables fault-tolerant longitudinal control in case of sensor faults or communication outage. The performance of this longitudinal vehicle control system for CACC has been partly validated according to the probabilistic validation methodology that was presented in Chapter 5. Test drives on a proving ground have confirmed that the use of vehicle-to-vehicle communication allows significantly better performance due to a faster response to disturbances of vehicles in front.

Ongoing research focusses on completing the validation procedure using the adaptive importance sampling strategy for the PreScan simulations, and providing an efficient test schedule for future VeHIL testing. Subsequently, the dependability of the control system can be validated by investigating the false and missed alarm rates during a field-operational test. In addition, more sophisticated cooperative control algorithms are under development for enhanced string stability and more naturalistic car-following behavior.



(a)



(b)

Figure 7.13: Test drive results for a traffic jam approach at 80 km/h: (a) using the ACC algorithm; and (b) using the CACC algorithm.

Chapter 8

Case study: Validation of a pre-crash system

Although advanced driver assistance systems (ADASs) have the potential to improve traffic safety significantly, as shown in the previous two chapters, not every traffic accident can be prevented. Instead of relying on *active* and *passive* safety systems separately in case of a crash, *integrated safety* systems can contribute to a further improvement in passenger safety. As was discussed earlier in Section 2.1.3, pre-crash systems (PCSs) are therefore developed that use environment sensing to improve the effectiveness of passive safety restraints by activating them before a collision occurs. A schematic representation of a PCS is given in Figure 8.1.

The objective of this chapter is to adapt and demonstrate the ADAS development process for PCSs. Section 8.1 starts with an overview of the subsequent design and validation phases for PCSs. The necessary modification of the vehicle hardware-in-the-loop (VeHIL) concept to address the challenges of pre-crash testing is presented in Section 8.2. Using an accident study, relevant pre-crash scenarios are identified in Section 8.3. A suitable PCS to address these scenarios is developed in Section 8.4. Section 8.5 briefly discusses the development of a simulation model and results of model validation in VeHIL. Using input from the accident study, appropriate VeHIL test scenarios for the validation of this PCS are defined in Section 8.6. Corresponding test results are presented in Section 8.7. In Section 8.8 the benefit of PCSs is assessed with MADYMO simulations, which complement the methodological framework. Finally, Section 8.9 concludes the chapter.

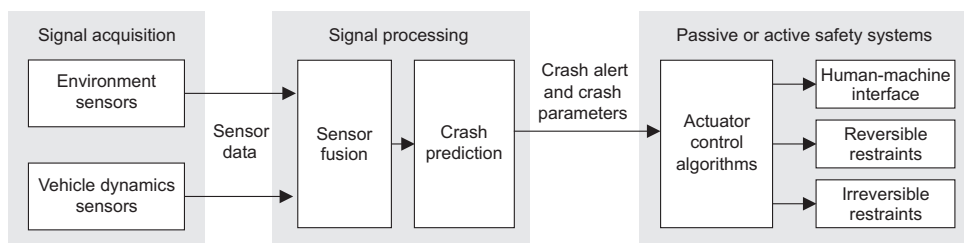


Figure 8.1: Schematic representation of a pre-crash system.

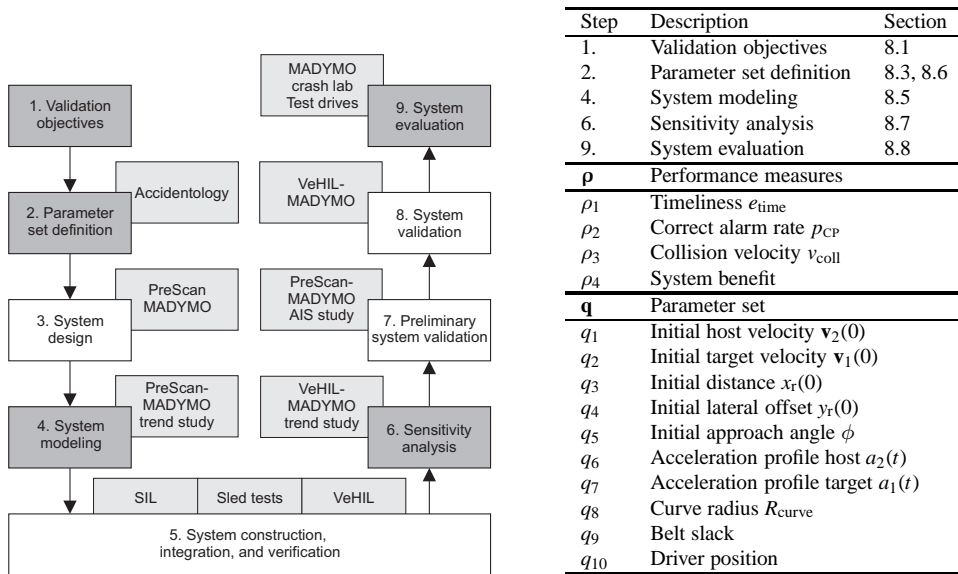


Figure 8.2: Schematic representation of the methodological framework for validation of pre-crash systems, including the use specific tools, such as MADYMO simulations, crash lab, and sled tests. The steps that are investigated in the PCS case study are shaded dark grey.

8.1 Tools and methods in the design and validation of PCSs

PCSs are *safety-critical* systems that require a high level of performance and dependability for a wide range of (near-)collision situations. Unnecessary deployment of safety restraints could be very dangerous for the vehicle occupants. False alarms should therefore be minimized to the levels defined in Chapter 2. Unfortunately, with field trials it may take a very long time to encounter a representative set of scenarios and to obtain a reliable estimate of the dependability of the system. Testing a PCS for missed alarms is equally impractical, since an actual collision would be necessary to reproduce the operating conditions. Because of the problems involved with testing PCSs on the road, relevant pre-crash scenarios are often reproduced using crash tests with foam dummy vehicles [199]. However, these tests are not repeatable and lack accurate ground truth knowledge on the position and velocity of the vehicles involved in the test [35]. As an alternative, crash tests can be performed in a crash lab, but these are limited, due to the testing effort and prototype costs. Furthermore, similar to the Euro-NCAP tests for passive safety, there is a need for guidelines on assessing the benefit of PCSs. The objective of this chapter is therefore to adapt the tools and methods presented in Chapters 4 and 5 for the development of PCSs, as illustrated by Figure 8.2.

8.1.1 Definition of the validation objectives

The development process starts with the definition of user requirements, which will be used for the validation of the final system. The main requirement for a PCS is that it provides a benefit in terms of injury mitigation. On functional level this means a timely, accurate, and

appropriate deployment of pre-crash safety restraints. Since PCSs are inherently safety-critical systems, the dependability requirements are even more stringent than for driver warning and longitudinal control systems. The system must exhibit robust behavior, such that these requirements are met in a variety of operating conditions, weather conditions, and (near-)crash scenarios. We should therefore define the parameter set.

8.1.2 Definition of the parameter set

An overview of relevant crash scenarios can be obtained by analysis of accident statistics. Relevant scenarios take into account the severity, as well as the frequency, of the accident scenarios. Crash parameters, such as impact speed, impact angle, and offset, vary through a wide range. Furthermore, the pre-crash position of the driver in his seat, as well as any belt slack, are disturbances that have an important effect on the performance of the PCS.

8.1.3 Pre-crash system design

The first step in the design phase is to specify safety restraints that are supposed to provide a significant benefit for the selected scenarios. Specification of optimum timing settings of the restraint activation is crucial in this phase. However, an important disadvantage of reversible actions, such as belt pretensioners, is the relatively long activation time of the electric motors (as opposed to conventional pyrotechnic devices). Therefore, reversible safety measures should be activated *before* the crash, based on accurate information on the pre-crash scenario.

8.1.4 System modeling

Simulation plays an important role in the specification and design of a PCS. The software package MADYMO (acronym for Mathematical Dynamic Models) [157] is used for efficient design and optimization of the crash safety performance of vehicles. A main application area is the development and integration of restraint systems, such as seat belts and airbags. In the design phase a MADYMO trend study can provide an initial estimate for the required trigger time of the belt pretensioner. These initial trigger values are then used in the pre-crash controller that estimates the collision probability and controls the safety actions when the probability is higher than a certain threshold.

The estimation of the collision probability places stringent requirements on the accuracy and the update rate of the sensorial platform. In addition, specifications for the sensor post-processing algorithms are drawn up, regarding obstacle detection rate, obstacle classification rate, tracking accuracy, path prediction, and threat assessment capability. An important design tool for these design aspects is the simulation environment PreScan, which is useful for optimization of the type, positioning, and specifications of pre-crash environment sensors.

In parallel to the specification of the actuators and sensors, the basic operation of the pre-crash decision algorithm must be defined. Its task is to activate a trigger when the sensor monitors an object within a certain predefined range. PreScan simulations are very useful for algorithm development, since complex and safety-critical traffic scenarios can be simulated, with no limitations regarding the number of traffic participants, closing speed, and vehicle dynamics. A trend study in PreScan is performed in this phase to test the

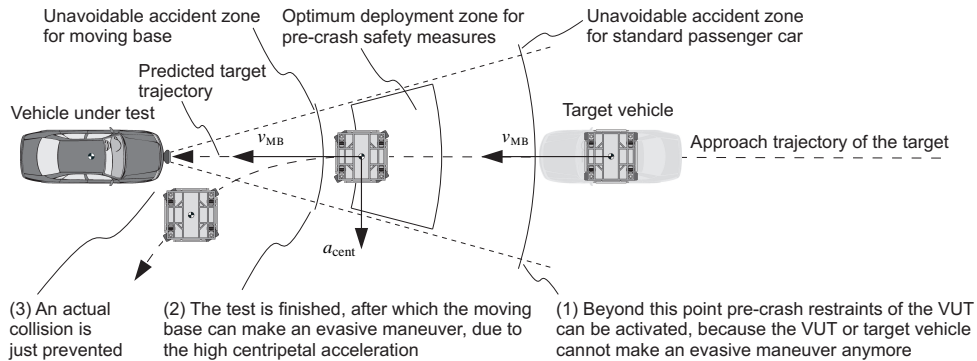


Figure 8.3: Pre-crash test sequence for a head-on collision scenario in VeHIL.

functional performance for various conditions and investigate the effect of crash parameters and disturbances.

8.1.5 System construction, integration, and verification

After the hardware components are selected, a first concept of the integrated system can be designed. The performance of the individual components should be verified to check their conformance to the specification. For example, the pre-crash decision algorithm requires verification of the braking and steering capabilities of the vehicle in order to estimate the crash probability. Furthermore, the environment sensor should be verified to provide an accurate and reliable measurement of the relative motion between sensor and obstacle sensor (*i.e.*, range, range rate, relative acceleration, and azimuth angle). Finally, MADYMO simulations and sled tests can be used in an iterative process for the verification and redesign of safety restraints. After the verification of these system components, the integrated PCS should be verified for electromechanical and software compatibility.

8.1.6 Challenges in system validation

In an iterative process the functional performance and system benefit of the PCS must then be validated, as illustrated in the right-hand side of Figure 8.2. A PreScan simulation study can be performed using the adaptive importance sampling technique, from which the performance and dependability of the system can be validated. In order to evaluate occupant injury levels, the vehicle dynamics data obtained from PreScan simulations are used as input in MADYMO simulations [138]. The technique to link these two simulations will be investigated in Section 8.8. Similar to Steps 6 to 9 in Algorithm 5.7, the PreScan-MADYMO simulation results should be validated with the real hardware. Obviously, it is desired to test the system in representative, but non-destructive, operating conditions.

8.2 VeHIL testing of PCSs

To overcome the difficulties of testing an integrated PCS on a test track or in a crash lab, VeHIL simulations provide an alternative approach. During the pre-crash experiment the

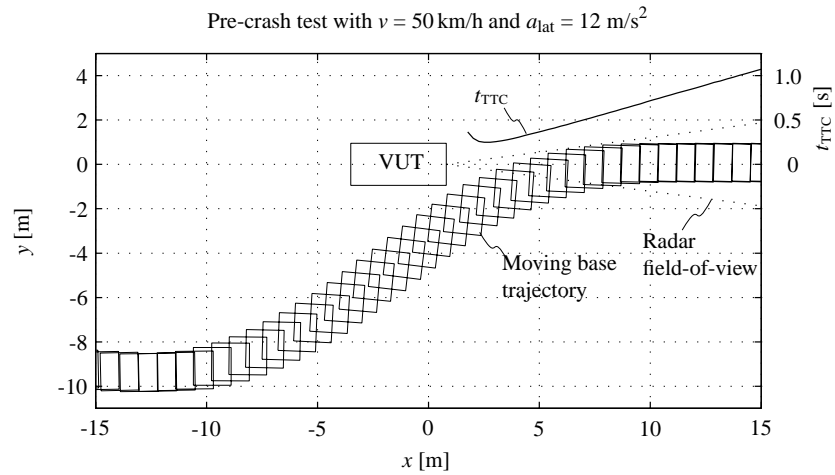


Figure 8.4: Overview of moving base trajectory and TTC during a head-on collision scenario.

moving base follows a crash trajectory, such that the sensor of the vehicle under test (VUT) recognizes it as a potential target. When the tracking algorithm estimates that a collision is imminent and unavoidable (considering conventional vehicle dynamics), it activates pre-crash restraints. However, an actual collision is avoided, because the moving base can achieve a higher centripetal acceleration a_{cent} than a normal passenger car. It can therefore make an evasive maneuver at the last moment, while still triggering activation of the PCS. After activation of the pre-crash restraints, the moving base starts the evasive maneuver and the test is finished. The vehicle and obstacle states at this time instant are used as initial conditions for the MADYMO simulation study, which will be further investigated in Section 8.8. The corresponding test sequence is illustrated in Figure 8.3.

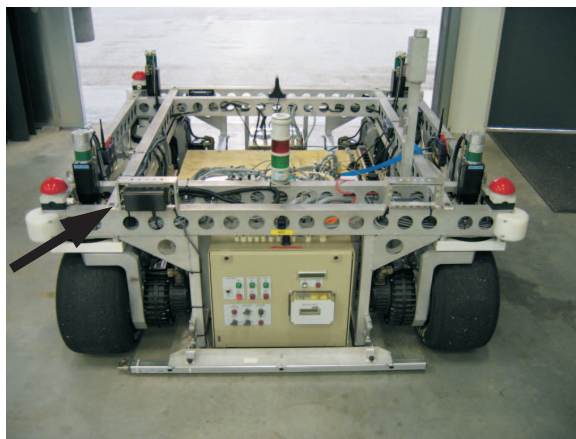


Figure 8.5: Moving base with radar sensor mounted on the frame (see arrow).

In case the pre-crash algorithm activates an autonomous braking system, the VUT would decelerate to reduce the crash velocity. In the VeHIL laboratory, the simulator calculates the corresponding relative motion ${}^L_2\mathbf{x}_1$ and sends position commands ${}^L_2\mathbf{x}_{\text{ref},1}$ to the moving base in a closed-loop setup. However, for pre-crash testing an *open-loop* test is usually sufficient, since the test is finished as soon as the VUT reacts. In that case the moving base motion is programmed to follow a pre-defined relative collision trajectory. In this way the vehicle state ${}^G\mathbf{x}_{2,\text{VUT}}$ (as measured by the chassis dyno) is not fed back through the simulator, such that the motion of the moving base is unrelated to the motion of the VUT.

These safety-critical experiments can be performed with a relative velocity up to 50 km/h. At this velocity the centripetal acceleration of 12 m/s^2 allows the moving base to approach the VUT very close and start the evasive maneuver at a time-to-collision (TTC) of only $t_{\text{TTC}} \approx 600 \text{ ms}$. Due to the time duration of the evasive maneuver itself and the delay in the sensor post-processing, the moving base appears as a critical target within the sensor field-of-view up to $t_{\text{TTC}} \approx 200 \text{ ms}$. Figure 8.4 shows the position and TTC during a pre-crash test, where the moving base just misses the VUT by 50 cm. In this way, it is possible to evaluate a PCS in a realistic, but non-destructive way.

Alternatively, environment sensors and inertial sensors can be installed on a moving base (see Figure 8.5) that executes a traffic scenario as if it were a standard road vehicle, while another moving base represents a target vehicle. This means reproducing the scenario with absolute velocities, as was shown in Figure 4.4(c) on page 82, such that a relative velocity of up to 100 km/h can be obtained. As discussed in the next section, this closing velocity covers about 95 % of all collision scenarios.

8.3 Accident study and parameter set definition

To identify the relevant scenarios for PreScan simulations and VeHIL testing, an accident study is performed. The focus of this study is on accidents involving two vehicles, since the PCS under consideration is designed for collisions between two vehicles. The goal of this study is to obtain a full set of scenarios capable of both aiding the initial system specification of a PCS, as well as facilitating a randomized simulation study on the basis of accident statistics.

8.3.1 Categorization of relevant scenarios

The selection of relevant scenarios is based on a statistical analysis of vehicle accidents reported in the 2000 National Automotive Sampling System (NASS) Crashworthiness Data System (CDS) [174]. The NASS-CDS is an in-depth database of a nationally representative sample of US police reported motor vehicle accidents. The analysis is based on 43 000 accident cases collected in the years 1993–2003. Although the NASS-CDS database has a bias towards moderate to severe accidents, the selected cases are representative of the 3 million two-vehicle accidents occurring in the US every year [173].

The NASS-CDS classifies accidents in over 100 types that indicate the accident cause, the pre-crash motion, and the impact location. For systematic testing these accident types are grouped into a limited set of generic scenarios, based on the similarity of the pre-crash motion from the sensor point of view. For example, the two scenarios ‘head-on’ collision and ‘sideswipe angle’ are treated as one category, since the pre-crash motion is similar (the target vehicle makes an inadvertent lane-change), but only the impact angle is different.

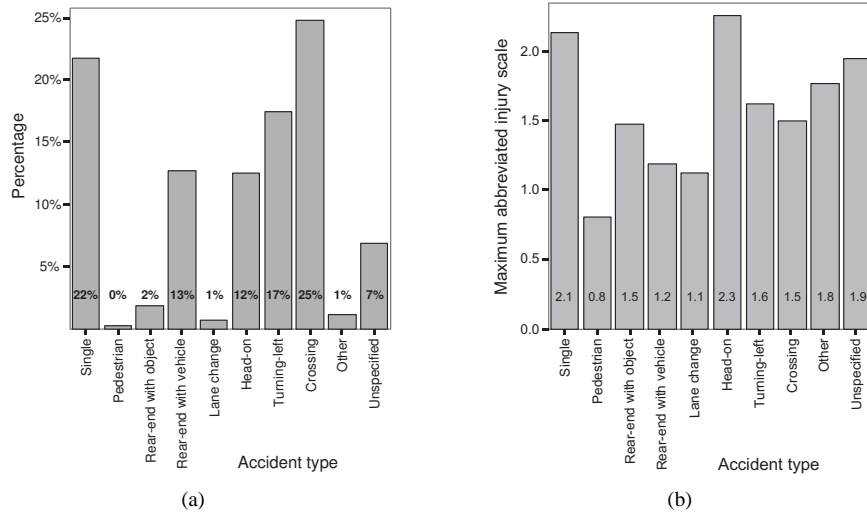


Figure 8.6: Accident type with (a) occurrence rate; and (b) average maximum abbreviated injury scale in the vehicle.

Categorization of the database results in eight generic scenarios plus a small number of ‘other’ and ‘unspecified’ scenarios, see Figure 8.6(a). Note the small percentage of pedestrian impacts, due to the fact that the NASS-CDS only considers collisions with injury to car occupants. However, for two-vehicle crashes the figures are representative. When only two-vehicle crashes with an occurrence rate above 5% are considered, the four remaining categories are ‘rear-end with vehicle’, ‘head-on’, ‘turning-left’, and ‘crossing-scenario’, as illustrated in Figure 8.7. These scenarios cover 72% of all accidents reported in the NASS-CDS and 99% of all vehicle-to-vehicle collisions.

Since PCSs are meant to increase occupant safety, it is important to take into account the severity of the accident type. Occupant injuries are commonly expressed in seven levels on the abbreviated injury scale, ranging from zero (no injury) to six (certain fatality). Figure 8.6(b) shows the maximum abbreviated injury scale, averaged over all cases. It can be seen that various accident types cause different injury levels. Except for the rear-end collisions, the selected generic scenarios have relatively high values on the abbreviated injury scale.

8.3.2 Crash parameters

An important factor in the severity of an accident is the impact velocity. This velocity is the *relative* motion at the time of impact, resulting from the *absolute* velocity vectors \mathbf{v}_1 and \mathbf{v}_2 of both vehicles. Crash parameters are therefore defined from a relative point of view, *i.e.*, from the point of view of the pre-crash sensor. As illustrated in Figure 8.7(d), this relative motion can be expressed in the polar coordinate system $\{P\}$ as an impact velocity v_{coll} and impact angle ϕ . In this analysis lateral motion (sliding of the vehicle) and the impact location (*i.e.*, offset) are not considered.

Figure 8.8(a) shows a polar plot of v_{coll} as a function of ϕ for all four selected scenarios together. The impact velocity is expressed in the 5th percentile, the 50th percentile (the

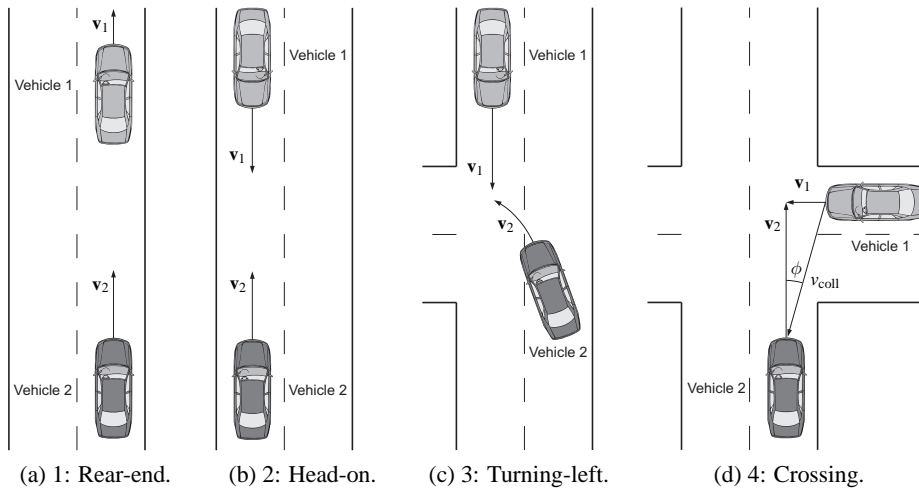


Figure 8.7: Generic impact scenarios identified from the NASS-CDS database.

median), and the 95th percentile. Collisions occur over almost 180°, with the impact angle divided in fractions of 10°. In addition, Figure 8.8(b) shows the maximum level on the abbreviated injury scale as a function of v_{coll} and ϕ . This graph shows that the most severe accidents happen at a smaller angle and higher velocity. This implies that head-on collisions would especially benefit from a pre-crash sensor installed in the front of the car. Information on v_{coll} and ϕ for the four scenarios separately is provided in Figure 8.9. For the head-on and turning-left scenarios, results are provided for both the striking and the struck vehicle, since in these cases both cars may benefit from a forward-looking PCS.

Further analysis of the scenarios shows that the rear-end impact scenario mostly occurs in longitudinal direction with an average impact speed of 50 km/h. On the other hand, head-on collisions have a significantly higher impact velocity with an average value of about 100 km/h, and a wider spread in impact angle. Both Figure 8.9(b) and (c) show similar crash parameters: a median value around 100 km/h, a 95th percentile around 200 km/h, and a 5th percentile around 60 km/h.

In crossing scenarios (with the PCS-equipped vehicle hitting another vehicle in the side) the impact angle varies from -70° to 80° . The average impact speed is around 60 km/h. The turning-left scenario shows different crash parameters for the two cars involved. The vehicle that drives straight ahead is mostly hit on the left front side between 0° and 40° , whereas the vehicle turning left is generally hit on the right front side, from -90° to -10° . The average velocity difference is 80 km/h.

An overview of these *crash parameters* is given in Table 8.1. In addition, a PCS should be able to recognize an imminent collision for a wide variety of pre-crash *scenario parameters*, such as curve radius, velocity profiles, and steering behavior. From this table it follows that passenger safety is significantly improved by designing a PCS with a forward-looking sensor installed in the front of the vehicle, although there are systems under development that focus on side collisions [162].

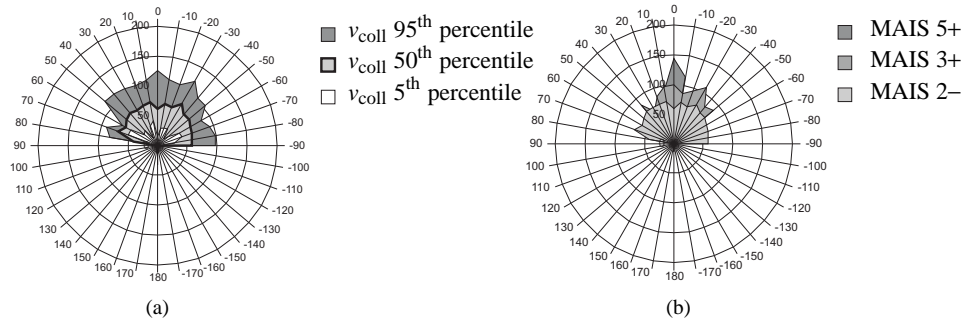


Figure 8.8: Impact velocity v_{coll} [km/h] as a function of impact angle ϕ [°] for all collisions: (a) distribution; (b) injury severity in terms of the maximum abbreviated injury scale (MAIS).

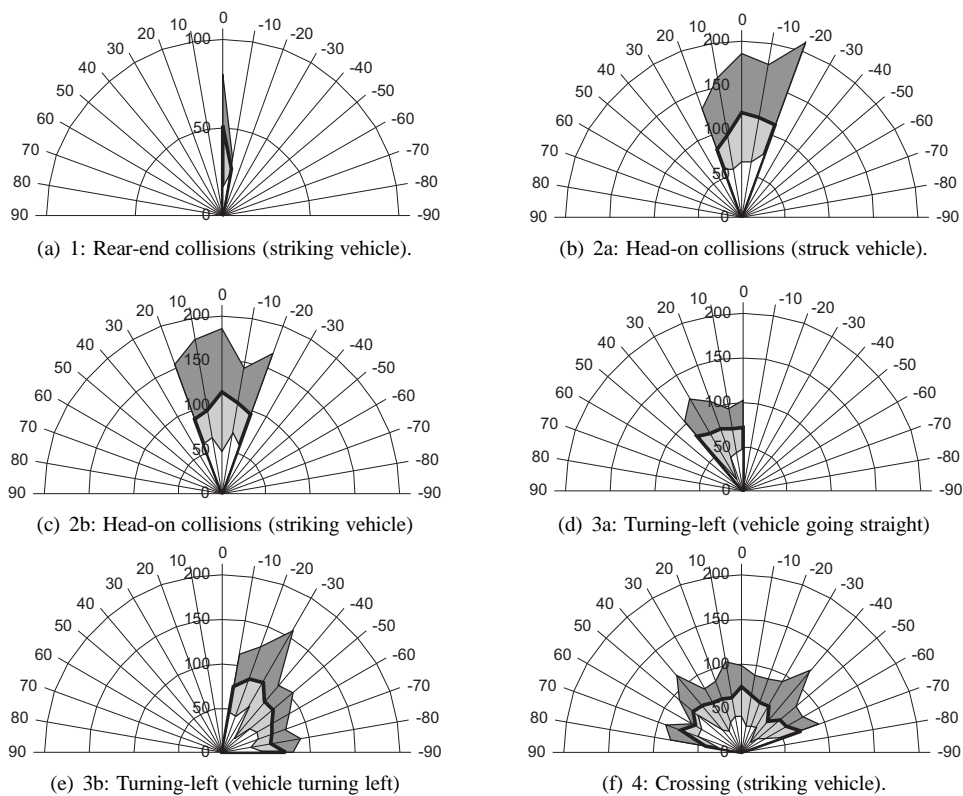


Figure 8.9: Polar plots of the distribution of v_{coll} [km/h] and ϕ [°] for the relevant scenarios: 5th percentile, median value, and 95th percentile. Note the smaller scale for v_{coll} for the rear-end collision.

Table 8.1: Overview of crash parameters.

No.	Scenario type	Impact speed [km/h]			Angle [°]		Number of cases [-]	
		5 th	50 th	95 th	Min	Max	Absolute	Weighted
1	Rear-end, striking vehicle	20	50	90	-10	10	573	684 000
2a	Head-on, struck vehicle	64	104	173	-20	20	330	107 000
2b	Head-on, striking vehicle	60	99	165	-20	20	315	122 000
3a	Turning-left, vehicle going straight	32	69	98	0	40	694	521 000
3b	Turning-left, vehicle turning	42	73	103	-90	-10	672	495 000
4	Crossing scenario, striking vehicle	36	62	90	-70	80	947	777 000

8.4 Specification and design of the PCS

The prototype system under consideration is representative of the first-generation PCSs that have recently been offered as an option on series production vehicles [240]. These systems typically use an existing forward-looking radar or lidar sensor that is installed for ACC. The particular radar used in this case study is the most recent version of the Forewarn[®] Smart Cruise Control system that Delphi has had in production since 1999 [40]. The radar converts real-world objects in front of the vehicle into radar targets and tracks those targets over time, including information on the range r , range rate \dot{r} , azimuth angle ϕ , and other target attributes. Some relevant specifications are noted in Table 8.2. The corresponding system configuration is shown in Figure 8.10.

As the radar tracks objects within its zone-of-coverage, real-time target data is transmitted over the CAN bus to a laptop computer for data collection, threat assessment, and display purposes. The pre-crash algorithm determines the threat level posed by each target and decides if and when a collision is imminent. Figure 8.11 illustrates the different steps of the operation:

1. Screen out non-closing and stationary targets that the host vehicle cannot hit.
2. Determine a set of measure ratings that reflect the probabilities for the motion of the host and target vehicles.

Table 8.2: Specifications of the forward-looking radar system.

Parameter	ACC-2 specification
Range coverage	1 to 150 m (for 10 m ² radar cross-section)
Range resolution	2 m (0.80 m range bin)
Range accuracy	± 2 m
Range rate coverage	-64 to +32 m/s
Range rate accuracy	±0.5 m/s
Azimuth coverage	15°
Azimuth accuracy	±0.3°
Number of tracked targets	15
Acquisition time	< 0.3 s
Cycle time	100 ms
Sensor size	140 × 70 × 100 mm
Operating frequency	76 GHz

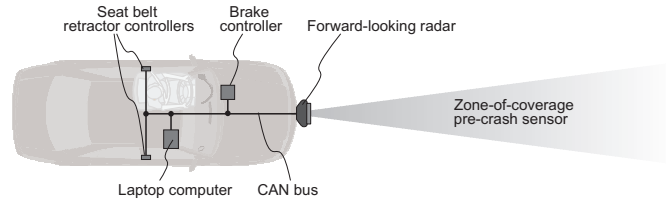


Figure 8.10: Pre-crash prototype system configuration.

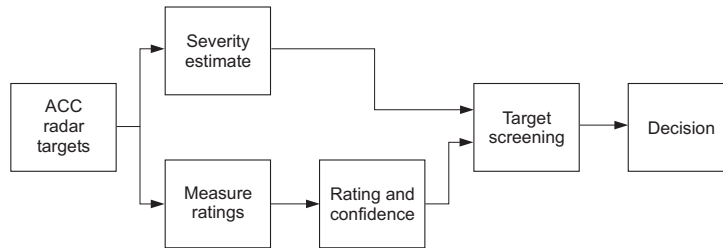


Figure 8.11: Pre-crash decision algorithm.

3. Combine the measure ratings and determine the confidence of the decision to deploy pre-crash restraints.
4. Determine the severity, represented by the predicted impact velocity.
5. Combine the measure ratings, confidence, and severity into the final decision to deploy.

In the current setup, the driver is warned below a TTC of 5 s by vibration of the seat belt. This warning provides the driver the opportunity for collision avoidance. The system also activates a brake assist system to help the driver reduce the crash velocity before an impact. In addition, the vehicle is equipped with reversible pretensioning seat belt retractors to remove slack from the seat belt, and to bring the occupants to an optimal position in case of an unavoidable crash [39]. This improves the effectiveness of the airbag and seat belt restraint systems during collision. The activation time for these reversible belt retractors is set between $t_{TTC} = 600$ ms and $t_{TTC} = 200$ ms. Autonomous activation prior to that time would be more comfortable, but also increases the probability of a false alarm in the event that the driver of either the host or target vehicle could still avoid the collision.

8.5 Generation of a PreScan model of the PCS

In order to validate the pre-crash algorithm, the software code is simulated in PreScan, where the rest of the vehicle, the radar sensor, and the environment are simulated. The virtual sensors in PreScan are modeled in such a way that an online interaction with the virtual world is permitted. The existing radar model from the PreScan library was adjusted to emulate the hardware radar [238], and sensor data processing algorithms were implemented. In



Figure 8.12: Virtual VEHIL laboratory in PreScan.

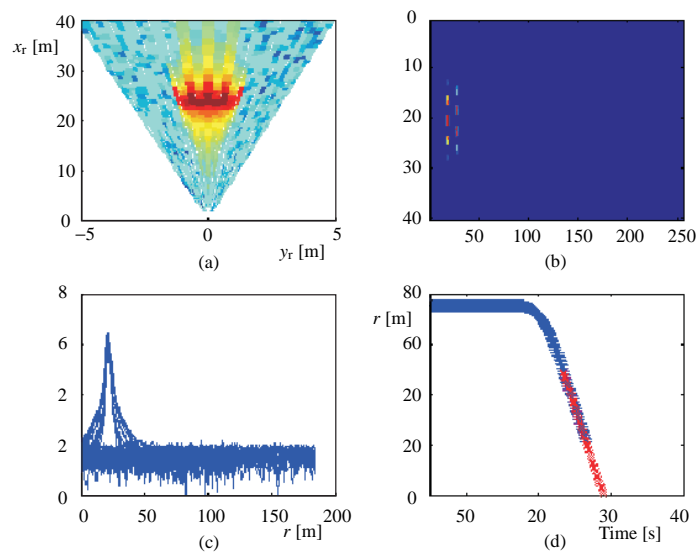


Figure 8.13: PreScan simulation results: (a) the reflection of the moving base; (b) The complex demodulated incoming radio frequency signal of the virtual radar (also known as IQ-data, where IQ stands for in-phase and quadrature-phase, reflecting the fact that the signal is complex with a real and imaginary part); (c) the reflection intensity in function of range; and (d) the range in function of time (blue line represents the simulated results and red are the experimental results).

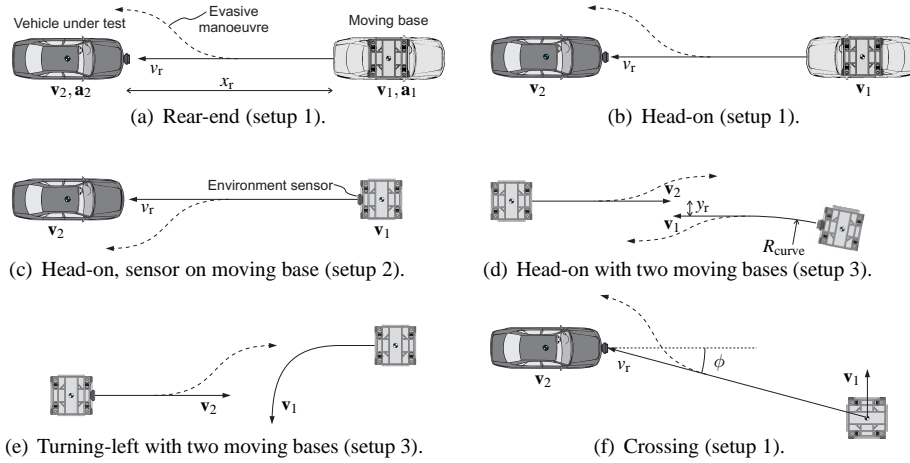


Figure 8.14: VeHIL pre-crash scenarios. The transparent car body in (a) and (b) depicts the moving base orientation.

addition, a virtual representation of the VeHIL laboratory was created in PreScan to facilitate in the model validation, see Figure 8.12

Figure 8.13(a) provides a simulation result of the reflection of the moving base during an approach scenario. The intensity of reflected signals is plotted for the different beams of the radar as function of the range r . Variations in intensity occur due to the fact that adjacent beams are in a different phase of the frequency modulation. The simulated radar output is processed using the implemented data processing algorithms as shown in plots 8.13(b) and 8.13(c).

Validation of the sensor model in PreScan is performed by comparing simulation results with the VeHIL test results (to be discussed later on) for the same scenario. Figure 8.13(d) compares the resulting range estimations for a given test scenario with the actual VeHIL data. Note that experimental data are available only for ranges below 50 m. It can be seen that simulated data correlate quite well with experimental data. For further details on the validation of the PreScan radar model, the reader is referred to the paper by Lemmen *et al.* [145].

8.6 Test schedule selection for pre-crash testing

According to Step 6 of Algorithm 5.7, a limited number of critical scenarios is used in a test schedule for VeHIL testing. Figures 8.14(a), (b), (e), and (f) successively show the corresponding test setup in VeHIL for the selected generic pre-crash scenarios: rear-end, head-on, turning-left, and crossing accidents. To recreate these four scenarios, the following experimental setups are used:

1. The target vehicle (the moving base) drives towards the sensor-equipped host vehicle (the VUT). This setup allows for the evaluation of the integrated PCS. The maximum relative velocity v_r is 50 km/h. See the head-on collision scenario in Figure 8.14(b).

2. The moving base is equipped with the sensor, as shown in Figure 8.5, and drives towards the VUT. Here the performance of the sensor is evaluated on a moving vehicle. Again the maximum v_r is 50 km/h. See the head-on collision scenario in Figure 8.14(c).
3. The sensor-equipped moving base drives towards another moving base. Both moving bases can drive at a velocity of up to 50 km/h, resulting in a maximum v_r of 100 km/h. See the head-on collision scenario in Figure 8.14(d).

To cover the relevant scenarios and conditions identified from the accident study, a wide range of scenario parameters should be used to test the system with a reliable outcome. The selection process is based on the parameter set \mathcal{Q} , defined by Table 8.1 and the sensor specifications of Table 8.2. The resulting set of scenario parameters is reduced to a representative

Table 8.3: Test schedule with initial conditions for pre-crash tests.

Scenario parameter	Symbol ^a	Unit ^b	Range ^c		Number of tests	
			Min	Max	Deploy	Non-deploy
<i>Scenario 1: Rear-end</i>					6	8
Host vehicle velocity	\mathbf{v}_2	km/h	32.2			
Target vehicle velocity	\mathbf{v}_1	km/h	0			
Relative velocity	v_r	km/h	-48.3	-10		
Acceleration profile host vehicle	a_2	m/s ²	-7	0		
Acceleration profile target vehicle	a_1	m/s ²	-7	0		
Distance to target	x_r	m	24	80		
Lateral offset	y_r	m	-2	2		
<i>Scenario 2: Head-on</i>					34	22
Host vehicle velocity	\mathbf{v}_2	km/h	20	48.3		
Target vehicle velocity	\mathbf{v}_1	km/h	0	48.3		
Relative velocity	v_r	km/h	-96.6	-32.2		
Approach angle	ϕ	°	-10	10		
Distance to target	x_r	m	80	128		
Lateral offset	y_r	m	-3.25	2		
Curve radius	R_{curve}	m	300	1000		
<i>Scenario 3: Turning-left</i>					2	0
Host vehicle velocity	\mathbf{v}_2	km/h	48.3			
Target vehicle velocity	\mathbf{v}_1	km/h	10			
Relative velocity	v_r	km/h	-58.3			
Distance to target	x_r	m	138			
<i>Scenario 4: Crossing</i>					1	2
Host vehicle velocity	\mathbf{v}_2	km/h	46.8	48.2		
Target vehicle velocity	\mathbf{v}_1	km/h	1.8	3.9		
Relative velocity	v_r	km/h	-48.7	-48.2		
Approach angle	ϕ	°	4.7	14		
Distance to target	x_r	m	70	80		
Lateral offset	y_r	m	-14.3	-8.5		

^aSee Figure 8.14 for an illustration of the parameters.

^bFor consistence with the accidentology database the unit for velocity is [km/h] instead of the usual SI unit [m/s].

^cWhen no maximum value is given, only the minimum value has been tested.

Table 8.4: Initial conditions for occupant position.

Posture	Angle [°] ^a	Slack shoulder belt [mm]
Leaning forward (position 1)	15	100
Leaning forward (position 1)	15	0
Leaning forward (position 2)	30	100
Leaning forward (position 2)	30	0

^aApproximate angle; see Figure 8.15 for an indication of the positions.

set of 75 tests, as shown in Table 8.3. The focus is on head-on collisions, since these are the most critical scenarios that the PCS should handle. In addition, near-miss scenarios are executed, in order to test the PCS for its capability to distinguish between a crash (deploy) and a miss (non-deploy) situation. Some tests are performed twice to check the repeatability of the decision by the PCS. Comparison of Tables 8.1 and 8.3 shows that the test schedule covers a wide range of the relevant conditions.

Although VeHIL testing is mainly aimed at the validation of the sensor and decision algorithms, testing also includes the reversible belt pretensioner. Currently, no anthropomorphic test device is available that represents the acceleration levels imposed by the belt pretensioner on the body realistically. Therefore, the operation of the pretensioner is evaluated using a crash test dummy and a human driver, as shown in Figure 8.15. This figure also shows the VUT on the chassis dyno and the approach of the moving base. The dummy is a TNO-10 dummy with an adjusted hip joint to exhibit realistic motion of the upper body during belt pretensioning. It has to be noted that no detailed assessment of this device was made. Instead, the dummy response is compared with the motion of the human driver to validate its representativeness. In order to check whether the PCS is able to pull the occupants to a position outside the airbag's firing region before the crash (to prevent airbag-inflicted injuries during the firing phase of the airbag), the conditions in Table 8.4 are considered.

8.7 Sensitivity analysis with VeHIL experiments

The results of the VeHIL pre-crash test schedule will first be illustrated by two examples. Then, the performance and dependability for all tests will be discussed.

8.7.1 Head-on collision test

The analysis starts with the head-on collision scenario of Figure 8.14(b), where the host vehicle drives with $v_2 = 20$ km/h and the target vehicle $v_1 = 28.3$ km/h. The resulting relative velocity of the moving base is -48.3 km/h. Figure 8.16 shows the resulting path of the target vehicle, along with the path measured by the radar sensor. At $t_{TTC} = 596$ ms the PCS algorithm estimates that a collision cannot be avoided anymore, after which the belt pretensioner is activated. Subsequently, the test is finished and the moving base makes an evasive maneuver with a centripetal acceleration a_{cent} of 12 m/s². Figure 8.16 also shows that the radar track continues towards the VUT after the deployment. This is due to the fact that the tracking algorithm predicts the target position for some time after it has left the sensor field-of-view. Figure 8.17 shows successive video stills of the pre-crash test.



Figure 8.15: Experimental setup of a pre-crash test. Clockwise: dummy position with postures indicated; occupant view; moving base approach; side view.

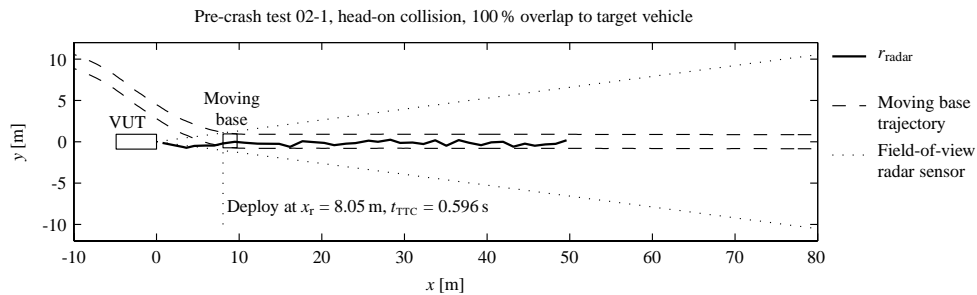


Figure 8.16: Overview of the head-on collision scenario of Figure 8.14(b).

The results in Figure 8.19 also show that during the pre-crash maneuver the moving base has a maximum error ε of 0.10 m between desired and measured position, and a repeatability within 0.03 m between consecutive test runs. The velocity error was verified to be smaller than 0.1 m/s. This accuracy is within the measurement noise of any automotive environment sensor. This ground truth data allows to check typical sensor characteristics, such as detection range and the field-of-view. From a detailed comparison of radar measurements and moving base trajectory, it was verified that the radar meets the specifications of Table 8.2. The plot also shows the increase of the threat level during the pre-crash maneuver. At $t = 15.3$ s the moving base is recognized as potential target. Subsequently, a warning is issued at $t = 17.0$ s. Because the driver does not react, the PCS deploys at $t = 18.4$ s, when the t_{TTC} crosses the threshold value of 600 ms. Finally, the moving base makes an evasive maneuver, resulting in the rapidly changing angle ϕ . In this way, the performance of the decision algorithm was validated with respect to the timeliness of the driver warning and restraint activation.

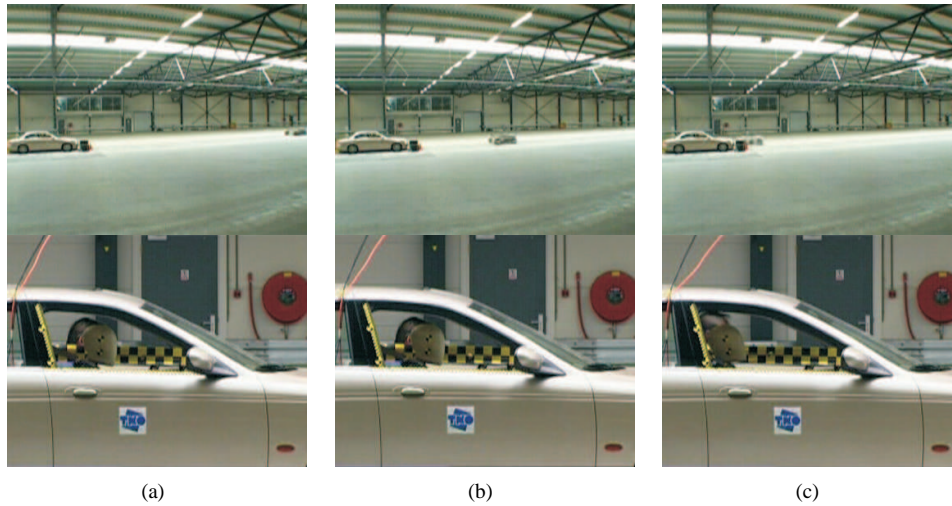


Figure 8.17: Successive video stills of the head-on collision test of Figure 8.19: (a) warning at $t_{TTC} = 2$ s; (b) deployment of belt pretensioner at $t_{TTC} = 0.596$ ms plus start of evasive maneuver; and (c) predicted impact at $t_{TTC} = 0$ s. The moving base trajectory is similar to that in Figure 8.4.

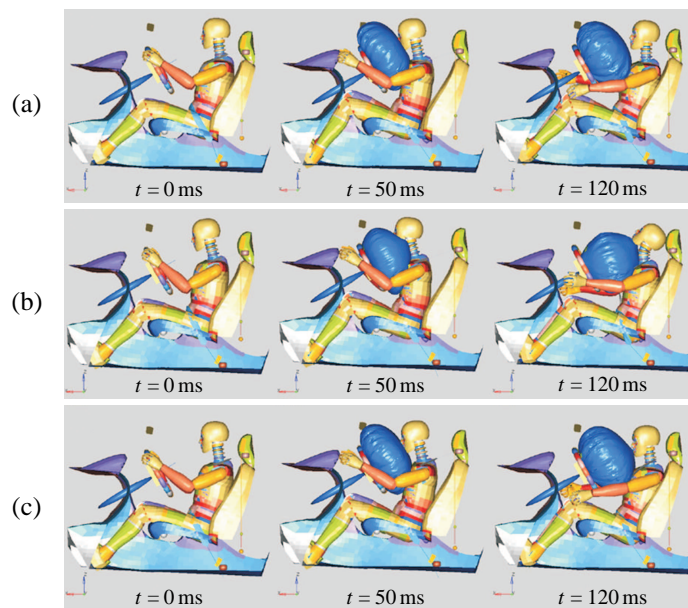


Figure 8.18: Animation of the MADYMO simulation for the rear-end collision scenario at impact ($t = 0$), at 50 ms, and at 120 ms after the start of the collision for three different configurations: (a) without pre-crash safety systems (50 km/h crash); (b) brake assist without pretensioning results in occupant out-of-position situation (35 km/h crash); (c) brake assist with pretensioner activated 500 ms before crash (35 km/h crash).

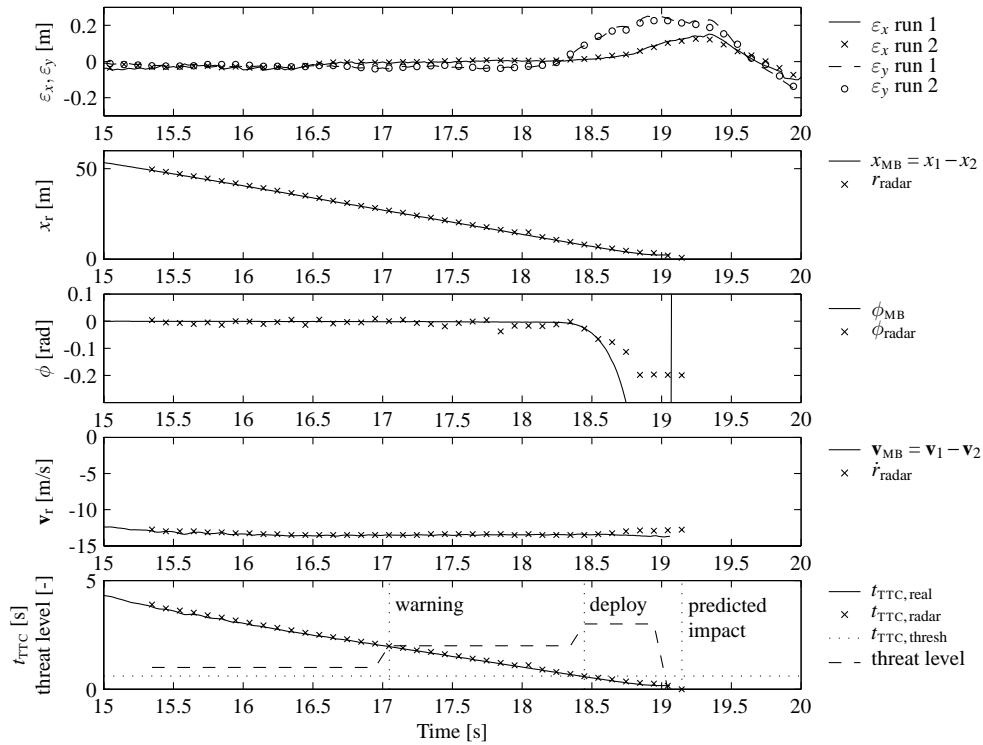


Figure 8.19: Test results for the head-on collision scenario of Figure 8.14(b). From top to bottom: position error of the moving base $\varepsilon_{x,y}$, range r , angle ϕ , range rate \dot{r} , time-to-collision t_{TTC} , and threat level.

8.7.2 Rear-end collision test

Figure 8.20 shows the test results for the rear-end collision scenario from Figure 8.14(a), where the host vehicle approaches a standing target with $v_2 = 48.3$ km/h. The resulting v_r is -48.3 km/h. The driver, alerted by the PCS warning, brakes 2 s before impact, which can be seen from the increase in v_r from $t = 18.5$ s onwards. However, the decision algorithm predicts that a collision cannot be avoided, since the relative velocity is too high. At $t_{TTC} = 529$ ms the PCS therefore activates the belt pretensioner. Comparison of the results with Table 8.2 show that the radar meets its specifications with a dynamic range accuracy of ± 1.5 m and range rate accuracy of ± 0.5 m/s. The lidar system (which is not used in the PCS algorithm, but available for verification purposes) has a slightly worse dynamic performance. A detailed comparison of Delphi's lidar-based and radar-based sensor systems is presented by Widmann *et al.* [262].

8.7.3 Validation of the system dependability

Similarly, tests were carried out for other scenario parameters, and for the turning-left and crossing scenarios, corresponding to the test schedule in Table 8.3. The test results were then used for validation of the PreScan simulation model, as discussed in Section 8.5. Fur-

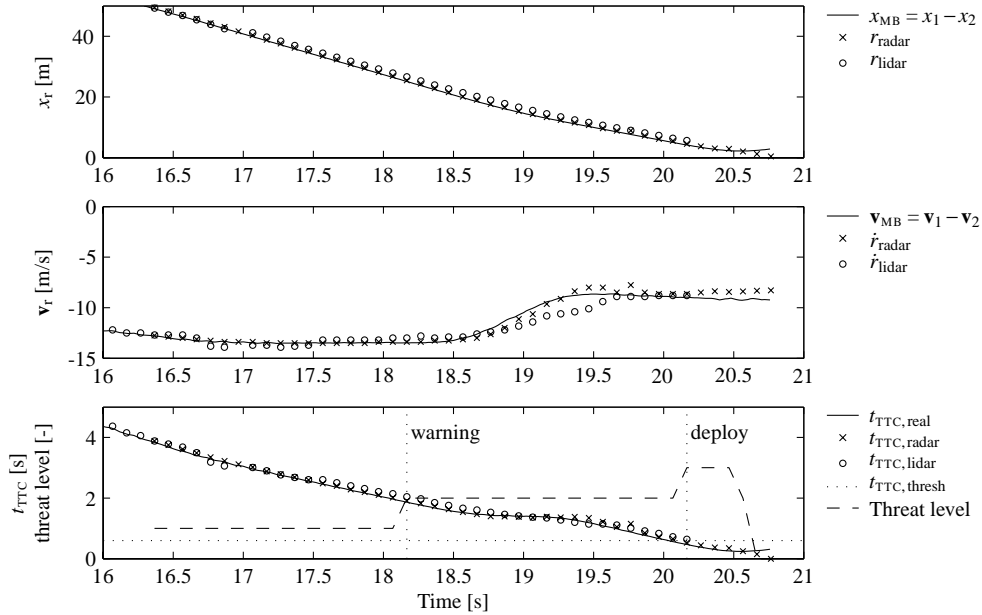


Figure 8.20: Test results for the rear-end collision scenario of Figure 8.14(a).

thermore, a preliminary validation of the PCS algorithm for a wide range of scenarios is performed, and potential problem areas can be identified. The algorithm passed 65 out of the 75 tests, but failed in 8 tests, *e.g.*, the system did not deploy when it should have (8 missed alarms out of 43 deploy tests), or deployed when it should not have (2 false alarms out of 32 non-deploy tests). This leads to a correct alarm rate \hat{p}_{CP} of 0.81. The timeliness of the activation of the belt pretensioner is depicted in Figure 8.21, which shows that some deployment actions occur a little late. After a set of initial experiments the tracking and decision algorithms were therefore fine-tuned. In the final test schedule the system then passed all tests, activating the belt pretensioner only when necessary and at an appropriate time.

8.8 Benefit assessment of pre-crash sensing

Apart from validation of the functional performance and dependability of the system, the safety benefit to the driver must be validated, since the main objective of a PCS is to mitigate crash injuries. Analysis of video recordings verified that with t_{TTC} between 400 and 600 ms the belt retractor effectively positions occupants outside the airbag's firing region before the imminent crash, such that the driver does not hit the airbag during, but only after inflation. In all tests the back of the occupant was in contact with or close to the seat rest upon the calculated moment of impact, as depicted in Figure 8.17 on page 195. This was the case for the TNO-10 dummy and the human driver in both the 15° and the 30° positions, including those with belt slack. As such, it is expected that the belt retractor increases the occupant's survival space, which significantly benefits occupant safety [65, 178, 202].

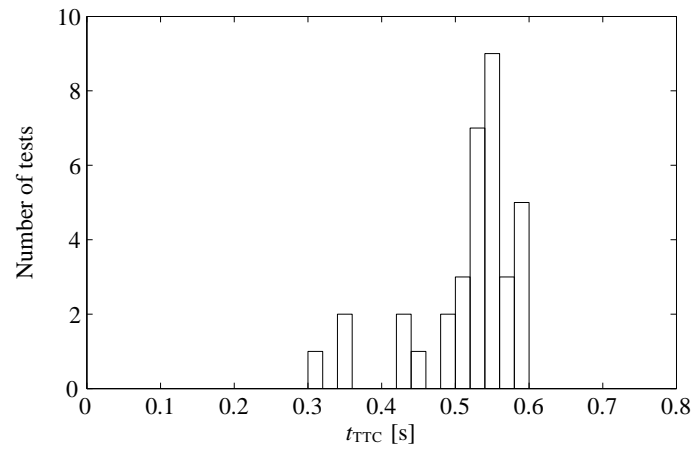


Figure 8.21: Timeliness of the activation of the belt pretensioner.

In addition, the test results indicated that with a pre-crash brake assist function applied 2 s before the collision, a reduction in crash velocity of around 15 km/h can be reached. At 50 km/h this velocity reduction corresponds to a 50 % reduction in the kinetic energy that has to be dissipated during this particular crash scenario. The PCS can therefore obtain an even further reduction in injury values [146]. It has to be noted that the occupants do not experience any deceleration during this pre-crash braking, due to the static setup of the VUT in VeHIL. In reality, this may increase the required time for full retraction and affect the position of the occupant. Volunteer tests that also take into account the variance in occupants, would therefore be required to make a more sound assessment of the actuator performance, since no anthropomorphic test devices are available yet. For more detailed benefit assessment we therefore present an alternative approach.

On the basis of PreScan simulations and VeHIL experiments, relevant crash scenarios are defined. To evaluate the effectiveness of the pre-crash restraints on passive safety, the states $\mathbf{x}_{2,VUT}$ of the VUT and $\mathbf{x}_{1,MB}$ of the moving base in VeHIL at $t_{TTC} = 0$ s are used as input for an off-line MADYMO simulation. When simulations are performed *with* and *without* these pre-crash restraints, the benefit of these actions with respect to occupant safety can be evaluated. Figure 8.22 compares several standard injury criteria for the reference rear-end crash scenario of Figure 8.14(a) at 50 km/h; a system with brake assist (where the crash speed is reduced to 35 km/h); and a system where the brake assist is combined with reversible belt pretensioning (as demonstrated in Figure 8.20).

Figure 8.18(a) on page 195 shows the animated results for the reference rear-end collision scenario. In Figure 8.18(b), the occupant's position is moved forward due to the deceleration of the vehicle, resulting in a disadvantageous position relative to the airbag, called 'out-of-position'. At the time instance the airbag is fired, the occupant is already too close to the steering wheel. When inflating, the airbag hits the occupant under the chin, evoking a negative bending moment of the neck that exceeds the neck injury tolerance limit. Serious injury would therefore be inflicted to the neck, as indicated by the FNIC criterion in Figure 8.22.

By pretensioning the seat belt directly upon the activation of the brake assist, the forward motion of the occupant can be reduced significantly, as indicated in Figure 8.18(c). The

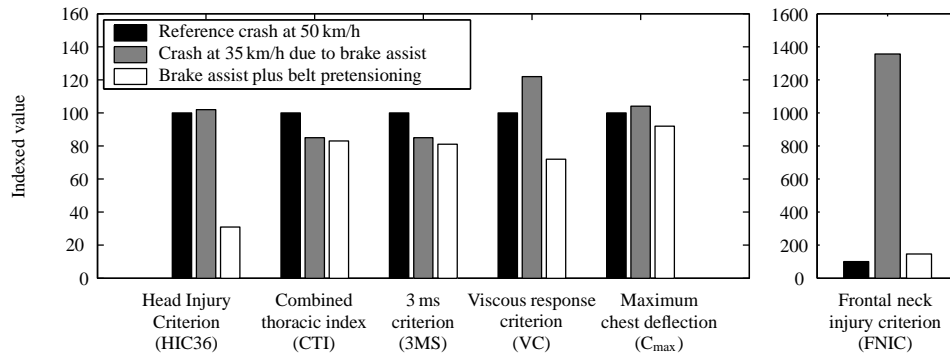


Figure 8.22: Injury criteria for a simulated rear-end collision scenario with and without pre-crash actions.

reversible belt pretensioner causes a maximum retracted seating position that is even further backward than the initial seating position. The occupant no longer gets out-of-position and the injury criteria are reduced correspondingly, as shown by the white bar in Figure 8.22. We therefore conclude that any PCS that activates a brake assist or autonomous emergency braking system, should also activate a belt pretensioner before the crash to prevent out-of-position of the occupant and subsequent airbag-inflicted injuries.

8.9 Summary

This chapter has presented the use of the tools PreScan, VeHIL, and MADYMO for the design and evaluation of PCSs, and integrated their application in the methodological framework that was presented earlier in Chapter 5. After the definition of the validation objectives and the parameter set using accidentology analysis, a wide range of accident scenarios can be simulated in PreScan and MADYMO. Subsequently, simulation results can be used to define and perform VeHIL experiments. In VeHIL, PCSs can be accurately and efficiently tested for critical scenarios in a non-destructive way.

This design and validation methodology has been demonstrated using a prototype PCS, consisting of a radar system to detect an imminent collision, and motorized belt retractors and a brake assist system to reduce the injury severity. A test schedule has been developed, based on accident data from the NASS-CDS database. The scenarios included rear-end, head-on, turning-left, and crossing scenarios, accounting for 99 % of all vehicle-to-vehicle collisions.

The repeatability and accuracy of VeHIL is within the noise level of automotive sensors. The moving base trajectory can therefore be regarded as ground truth, which allowed to verify that the performance of the radar meets its specification. The performance of the decision algorithm and activation timing was validated for a wide variety of scenario parameters, which resulted in a small amount of false and missed alarms. Furthermore, it has been verified for all tests that activation at a time-to-collision of approximately 600 ms enables to fully retract the driver from its leaning forward position to a position outside the airbag's firing region. The sensor data logged during VeHIL tests can be used for further

optimization of the robustness of the obstacle detection algorithms and decision algorithm.

MADYMO simulations have been used to evaluate the effectiveness of PCSs on injury reduction. It was shown that brake assist in combination with belt pretensioning can considerably reduce the impact severity. For further evaluation of the PCS, test drives are required to evaluate the system's dependability in everyday traffic conditions. Furthermore, human body simulations are necessary to further investigate the system's effect on injury mitigation. However, the next steps in the validation (Steps 7, 8, and 9 in Algorithm 5.7) can be performed with a much higher confidence and less risk, since considerable insight in the system performance has already been gained with the trend studies in PreScan, MADYMO, and VeHIL.

Chapter 9

Conclusions and recommendations

This thesis has extended and integrated several fields of research into a methodological framework for the design and validation of advanced driver assistance systems (ADASs). The added value of this methodology has been demonstrated with case studies involving a driver warning system, a cooperative adaptive cruise control system, and a pre-crash system. The main conclusions of this research and the contributions to the state-of-the-art are summarized in Section 9.1. A critical review of these conclusions is provided in Section 9.2, which highlights the limitations of the methodology. Subsequently, Section 9.3 discusses ongoing and future research activities to address these issues.

9.1 Conclusions and contributions of this thesis

This section summarizes the conclusions and contributions in the context of the thesis objectives and research issues that were formulated in Chapter 1 and that are briefly repeated here. First, with the increasing complexity of the system and its environment, the ADAS must satisfy increasingly stringent performance and dependability requirements for a wide range of operating conditions. In the second place, during the design phase of an ADAS, fault-tolerant control methods are necessary to maintain dependable performance throughout the parameter set, which is the combined set of scenario parameters, disturbances, and failure modes. Furthermore, an important aspect of the validation phase is to accurately reproduce the conditions under which the control system operates. This emphasizes the need for new tools in the development process of ADASs. Finally, this process, and more specifically the validation phase, must be made more efficient, by covering a representative subset of the parameter set with a minimum sufficient number of simulations and experiments. The main result of this thesis is therefore that we have developed an efficient methodological framework and associated tools for model-based design and validation of ADASs. This methodology allows to provide quantitative measures for the performance and dependability of an ADAS with higher accuracy and more confidence than is possible with existing methods. The main conclusions are discussed in more detail below.

Definition of quantitative requirements and parameter set

The first contribution of this thesis is the development of quantitative requirements for performance measures, such as stability, driver comfort, safety, and warning timeliness. Furthermore, the often vague definitions for dependability requirements, such as reliability, safety, and fault tolerance, have been defined and quantified.

A microscopic traffic model, based on field data from inter-vehicle motion, has been constructed to provide a representative parameter set for the ADAS-equipped vehicle. In addition, for the purpose of pre-crash system validation, existing accidentology data has been extended to include pre-crash maneuvers. This parameter set is based on field data obtained from the NASS-CDS database, and accounts for 99% of all vehicle-to-vehicle collisions. Other elements in the parameter set, such as driver input, disturbances, and failure modes, have been defined to support the design and validation process of ADASs.

Fault-tolerant state estimation

To maintain a fault-tolerant operation in the presence of sensor faults, Chapter 3 has presented a new method for fault-tolerant state estimation. Extended Kalman filters are used to fuse measurements from vehicle state sensors, DGPS, environment sensors, and vehicle-to-vehicle communication, in order to obtain reliable state estimation of the host vehicle state and inter-vehicle motion. In addition, a generalized observer scheme has been designed to manage faults that may occur in any of these sensors, and to provide an analytically redundant estimate.

Tools for model-based controller design and validation

Fortunately, faults are rare events, but this also makes it hard to validate the fault management system, which is a prerequisite for demonstrating the dependability of the ADAS. Chapter 4 has therefore presented the unique vehicle hardware-in-the-loop (VeHIL) concept for testing ADASs, where a real intelligent vehicle is operated in a hardware-in-the-loop environment. VeHIL resolves most of the difficulties associated with full-scale prototype tests, while still having a high level of representativeness, due to the use of real hardware. VeHIL experiments are performed in an accurate, repeatable, and controllable manner to create a representative test environment. It was also demonstrated that VeHIL has an added value in several phases of the development process of an ADAS, such as sensor verification, rapid control prototyping, model validation, functional performance validation, fault injection, and homologation.

The simulated components of VeHIL are based on the multi-agent simulation environment PreScan (Pre-crash Scenario analyzer). PreScan enables reliable simulation of ADASs in a microscopic traffic simulation, using validated physical sensor models in a virtual environment.

Due to this model-based control design process, VeHIL facilitates the transition from simulations to outdoor test drives, which are used to evaluate the system behavior in the real traffic environment. However, these tests can be performed with a much higher confidence and less risk, when the ADAS has already been thoroughly tested in VeHIL. VeHIL is therefore *not* meant to replace simulations and test drives, but to form an efficient link between them. Consequently, the number of iteration loops in the development process is reduced, saving time and costs.

Adaptive importance sampling for efficient validation

For further optimization of the use of PreScan and VeHIL, a methodological framework for probabilistic validation has been developed, based on a new randomized algorithm for adaptive importance sampling. This algorithm provides an upper bound on a sufficient sample complexity to obtain guaranteed estimates for the ADAS performance and dependability, given a desired accuracy and confidence level. This probabilistic approach cannot *prove* that the system has adequate performance and dependability. However, when we accept a (small) risk of failure, this probabilistic approach *is* able to obtain an efficient estimate of the performance and dependability of the system. Especially compared to conventional grid-based validation and Monte Carlo simulation, our approach is significantly more efficient in terms of the number of experiments that is required to obtain a desired level of accuracy and confidence. Furthermore, use can be made of *a priori* information on the system, thereby emphasizing interesting samples and speeding up the validation process. Especially the design of experiments theory can be efficiently used to investigate the sensitivity of the performance measure to particular parameters or combinations thereof. Critical parameter combinations are then subsequently emphasized in the validation process.

A major advantage of the methodology is that the sufficient sample complexity for PreScan simulations and VeHIL experiments can be predicted *a priori*. This allows to allocate the appropriate time and resources that are required for the validation of a control system. This is an improvement over the trial-and-error methods that are currently used in practice, and for which the required time and resources can only be approximated beforehand, based on engineering judgement. In addition, based on the outcome of the simulation study, a suitable VeHIL test schedule can be derived.

Added value of the methodology to ADAS control system development

The current automotive validation process is largely based on trial-and-error methods, which involves thousands of scenarios to be tested, consuming large amounts of man power, test track capacity, and available budget. The presented methodological framework and associated tools for design and validation of ADASs can therefore support car manufacturers and suppliers of ADASs in the following phases of the validation process:

- *Test preparation*: The scenarios to be tested can be selected in a more efficient way.
- *Test execution*: Each test run can be executed in a fast, accurate, flexible, and reliable manner, due to the use of dedicated test tools.
- *Test analysis*: More useful test results are available, because of the higher accuracy, availability of ground truth information, and more insight in the test operation.

This added value of the methodology has been demonstrated by application to three case studies. Although none of the case studies demonstrate *all* steps of the methodology, every step is illustrated by at least *one* of the cases, as shown by Table 5.3 on page 142. The use of the methodology has allowed to improve the ADAS control system design and validate the performance and dependability of these demonstrators.

Chapter 6 has provided the validation of a driver information and warning system for safe speed and safe distance (SASPENCE). Results of the VeHIL experiments show that the reference maneuver module and the warning and intervention strategies need to be fine-tuned to further improve the dependability of the system.

Chapter 7 has presented a control algorithm for cooperative adaptive cruise control that can achieve a string-stable following behavior by using environment sensing and vehicle-to-vehicle communication. It is further shown that feedback of the acceleration of preceding vehicles enhances the comfort, stability, and safety of the longitudinal control function. Furthermore, the fault management system provides the capability to achieve this performance and dependability, even in the presence of safety-critical failure modes.

In Chapter 8 the methodological framework has been adapted for validation of pre-crash systems in non-destructive and representative pre-crash scenarios. For this purpose the VeHIL concept has been modified, such that a vehicle equipped with a pre-crash system can be approached up to a time-to-collision of approximately 200 ms. MADYMO simulations and VeHIL experiments have demonstrated the safety benefit of a pre-crash system, consisting of both a belt pretensioner and a brake assist system.

9.2 Critical discussion of the results

The above-mentioned results are subject to several limitations, as discussed in this section.

Careful definition of performance measures

First of all, the definition of evaluation criteria is subject to discussion and depends on the type of ADAS and the validation objectives. The definition of dependability measures can easily be misinterpreted, and care should be taken to select the appropriate criteria that are relevant for the subsystem under consideration. Furthermore, it is still up to the validation engineer to judge whether the outcome of the validation procedure is acceptable. Although there is a limited possibility for testing against subjective criteria, the psychological elements of driving behavior can better be evaluated in a driving simulator or in a field-operational test.

Limited fault tolerance in case of multiple failures

Unfortunately, fault management systems make the ADAS control system even more complex, and thus more difficult to test. Furthermore, although the fault tolerance has increased, the current system is still not able to detect multiple faults. Furthermore, it remains difficult to test the system for *all* possible failure modes, since unexpected faults may remain unnoticed in the development process and only appear during operation.

Limited suitability of the validation tools

If simulators are to be used to test critical systems then clearly our confidence in the test results will be affected by the validity of the simulations used. This raises the issue to what extent the simulation environment is representative and repeatable, since we can only assign a level of dependability to a controller, when the level of dependability of the test tool is at least as high.

The validity of the PreScan simulation results strongly depends on the model uncertainty. The outcome of the adaptive importance sampling algorithm may therefore differ from the 'real' outcome. By using real hardware, VeHIL tests can reduce this model uncertainty and provide a validation of the simulation model. In an iterative approach, the

performance and dependability of the ADAS can then be estimated. However, care must still be exercised when interpreting the simulation results, and the test engineer should always verify that the simulation results and the subsequent VeHIL test schedule make sense.

As was shown in Chapter 4, VeHIL forms a representative test environment, but there are limitations for certain scenarios and systems:

- The VeHIL facility might disturb environment sensors, such that the sensor does not receive a representative input, *e.g.*, ‘ghost’ reflections for radar, or the absence of a moving environment for camera vision.
- Testing the effect of certain weather conditions (snow, rain, and ice) is currently not possible.
- Inertial motion is absent, such that accelerometers and yaw rate sensors do not give a representative signal. In addition, active braking systems will not cause a realistic deceleration that results in a forward displacement of the occupants, as will be relevant for pre-crash testing.
- There is a limitation to the type of scenarios that can be tested in VeHIL. Specifically, head-on collisions (due to high speed) and side collisions (due to limited floor space), are difficult to emulate in VeHIL. Furthermore, VeHIL cannot be used to test systems in scenarios with a time-to-collision below 200 ms.
- Scenarios are currently limited to cars only, since slowly moving pedestrians cannot be emulated by the moving bases.

Careful interpretation of validation results

Although the methodology provides a VeHIL test schedule that can be considered sufficient from a theoretical viewpoint, in practice this selection may not be entirely appropriate. Some VeHIL tests may be impossible to perform. On the other hand, extra tests may have to be performed based on engineering judgement; to obtain a better insight into the system operation; to analyze the dependence of performance criteria on the parameter set; or for homologation and benchmarking purposes.

Although a confidence interval can be a sufficient assurance for the control designer concerning the validation of control system performance and dependability, it can never be *proven* that the estimated values for dependability will be representative of the real-world experience. The representativeness of the probabilistic sampling approach directly depends on the level to which the probabilistic definition of the underlying parameter set will correspond to the real world. Sometimes, probabilistic information for a parameter $q_i \in \mathcal{Q}$ may be unreliable or not available at all, which limits the added value of the methodology for that particular parameter, and possibly for the entire validation process.

In practice, the parameter set for a particular driver in a particular scenario will always be different from the parameter set definition that is used for the sampling process. The outcome of the simulation process must therefore never be interpreted as the *real* value that the system will achieve in practice for that particular situation. Nevertheless, provided this limitation is taken into account, the methodology can give a fairly good interpretation of whether or not the ADAS operates appropriately for the *average* user. Moreover, since the most important requirements are defined in probabilistic terms (*e.g.*, reliability, safety), a probabilistic outcome is more suitable than a deterministic solution to the problem.

9.3 Recommendations for future research

This section summarizes several remaining research issues to complement the methodological framework. In addition, this thesis has revealed some new directions for future research that are critical for a successful market penetration of ADASs.

Carry out system validation and improvement of tools and methods

Within the scope of this project it has not been possible to apply all elements of the validation methodology to a single case study. Instead, three case studies were carried out, each of which demonstrated some of the phases in the validation process. A topic for future research is to apply the methodological framework to the entire development cycle of an ADAS. This experience will be useful to demonstrate the added value and implement any necessary modifications to the methodology.

In addition, the case studies in this thesis have emphasized the validation process with respect to investigation of performance measures. Comparatively little results on the volume of the cleared and non-cleared parameter subsets have been presented. Future validation projects should therefore investigate the ability of the methodology to estimate the volume of these subsets under the influence of model uncertainty and other disturbances and faults.

Furthermore, we have assumed in this thesis more or less correct knowledge of the underlying distribution of the parameter set. However, it would also be relevant to consider the effect of uncertainty in the multivariate probability density function on the performance measures.

The case studies in this thesis have been limited to longitudinal support systems. Extension of VeHIL with respect to the capability for steering and the resulting lateral dynamics should be investigated. This would allow to test ADASs for lateral assistance, such as lane departure warning systems, and side collision warning and avoidance systems. In addition, the capability of VeHIL should be extended to accommodate a higher relative velocity and smaller time-to-collision for the purpose of testing pre-crash scenarios.

Increasing knowledge of driver behavior

Appropriate validation criteria must be developed for subjective requirements, such as for testing driver warning systems. In addition, the scenario database should be extended to multi-lane scenarios, and the corresponding probability density functions modified depending on the application, location, and driving style. It should be investigated how to link microscopic vehicle and driver models with macroscopic traffic simulators. Furthermore, knowledge of driver behavior is useful for tuning of future ADASs, in order to achieve a more naturalistic driving behavior. An approach for obtaining the necessary data is to set up a field-operational test. In this way the representativeness of the parameter set can be increased in terms of scenario modeling and the probabilistic distribution of scenario parameters. An alternative research approach is to investigate a real-time connection between the VeHIL laboratory and a driving simulator, which would enable to run high-fidelity dynamical behavioral studies.

Robustness and fault-tolerant control for complex hybrid systems

For future safety-critical applications, or even autonomous driving, a more comprehensive fault management system would be necessary. This system should be able to handle multiple simultaneous faults, not only in sensors, but also in actuators and vehicle components. This requires extra residual generation modules, but also a more sophisticated fault diagnosis. A first step would be to implement nonlinear systems for fault detection and fault-tolerant control. Furthermore, the use of a nonlinear state estimation system requires to investigate the observability and reachability for nonlinear stochastic processes.

A challenge in this respect is the complexity of future ADAS control systems, which typically are hybrid switched systems. These controllers are connected to other embedded controllers and vehicle subsystems through an in-vehicle network. The dependability of these systems cannot be guaranteed with current methods for robust and fault-tolerant control design. Considerable research effort is therefore required to develop methods that are able to increase the dependability of these networked systems. With regard to validation of these systems, randomized algorithms will be a promising approach that needs to be investigated.

Probabilistic design of ADAS control systems

In this thesis we have emphasized the *validation* of ADAS control systems by randomized algorithms, whereas the *design* of the control configuration was considered a given fact. In order to integrate the design and validation phase of future controllers, probabilistic methods for control synthesis should be investigated, where the system uncertainty is taken into account by randomized algorithms, as demonstrated by Tempo *et al.* [236].

Standardization of homologation procedures

Since there is a need for homologation procedures for ADASs, a joint industry-wide effort should be started to develop standardized procedures. An important element is a homologation procedure for pre-crash systems, complementary to existing Euro NCAP procedures. The safety benefit assessment of pre-crash systems necessitate the development of human body simulation models and crash test procedures. The results of this thesis provide useful input to these future procedures regarding the definition of evaluation criteria and the pre-crash scenario database.

Methods for increasing the market penetration of ADASs

Considering the huge safety potential of ADASs, a combined effort by the automotive industry, governmental institutions, insurance companies, and consumer organizations should be set up to increase the market penetration of ADASs. Methods are tax incentives, mandatory regulations, and increasing the familiarity of the public with these systems. An investigation into the costs and benefits of these methods should support the decision-making on government level and among automotive manufacturers.

Bibliography

- [1] R. Abou-Jaoude. ACC radar sensor technology, test requirements, and test solutions. *IEEE Transactions on Intelligent Transportation Systems*, 4(3):115–122, September 2003.
- [2] A. Agogino, S. Chao, K. Goebel, S. Alag, B. Cammon, and J. Wang. Intelligent diagnosis based on validated and fused data for reliability and safety enhancement of automated vehicles in an IVHS. UCB-ITS-PRR-98-17, California PATH, Berkeley, CA, USA, 1998. Available at: <http://www.path.berkeley.edu/PATH/Publications/PDF/PRR/98/PRR-98-17.pdf>
- [3] A. Agogino, K. Goebel, and S. Alag. Intelligent sensor validation and sensor fusion for reliability and safety enhancement in vehicle control. UCB-ITS-PRR-95-40, California PATH, Berkeley, CA, USA, 1995. Available at: <http://www.path.berkeley.edu/PATH/Publications/PDF/PRR/95/PRR-95-40.pdf>
- [4] G. Alessandretti. Collision mitigation and road users protection: First test results. In *Proceedings of the 13th World Congress on Intelligent Transport Systems and Services (ITS)*, London, UK, October 8–12, 2006.
- [5] G. Alessandretti, P. Baraud, C. Domsch, and G. Sala. A European activity on pre-crash application: The CHAMELEON project. In *Proceedings of the e-Safety Conference*, Lyon, France, September 2002.
- [6] M. Alonso, P. Garayo, and L. Herrán. Functional requirements. Deliverable D20.33, European Commission, IP PReVENT, Subproject SASPENCE, Brussels, Belgium, November 30, 2004. Available at: http://www.prevent-ip.org/en/public_documents/deliverables/d2033_functional_requirements.htm
- [7] S. Amberkar, J.G. D’Ambrosio, B.T. Murray, J. Wysocki, and B.J. Czerny. A system-safety process for by-wire automotive systems. *SAE Technical Paper Series*, 2000-01-1056:69–74, 2000.
- [8] B. van Arem, J.K.H. Carlier, O.J. Gietelink, C.J. van Waveren, F. Lilli, C. Lanfranco, and G. Ghigo. ADAS critical applications and related critical scenario. Deliverable D1, Version 1.4, European Commission – GALileo for safety of Life Application of driver assistaNce in road Transport (GALLANT), Brussels, Belgium, October 23, 2002. Available at: <http://www.regione.piemonte.it/trasporti/prss/biblioteca/dwd/progetti/veicolo/gallant.pdf>

- [9] B. van Arem, C.J.G. van Driel, and R. Visser. The impact of cooperative adaptive cruise control on traffic-flow characteristics. *Proceedings of the IEEE Transactions on Intelligent Transportation Systems*, 7(4):429–436, December 2006.
- [10] J. Arlat, J. Boué, and Y. Crouzet. Validation-based development of dependable systems. *IEEE MICRO*, 19(4):66–79, July 1999.
- [11] K. Åström, P. Albertos, M. Blanke, A. Isidori, W. Schaufelberger, and R. Sanz. *Control of Complex Systems*. Springer-Verlag, 2001. ISBN: 1-85233-324-3.
- [12] K. Athanasas, C. Bonnet, H. Fritz, C. Scheidler, and G. Volk. VALSE – validation of safety-related driver assistance systems. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 610–615, Columbus, OH, USA, June 9–11, 2003.
- [13] R. Bakker, J. Hogema, W. Huiskamp, and Z. Papp. IRVIN – Intelligent Road and Vehicle test Infrastructure. In *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pages 947–952, Vienna, Austria, September 13–15, 2005.
- [14] Z. Bareket, P.S. Fancher, H. Peng, K. Lee, and C.A. Assaf. Methodology for assessing adaptive cruise control behavior. *IEEE Transactions on Intelligent Transportation Systems*, 4(3):123 – 131, September 2003.
- [15] M. Basseville and I.V. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. Prentice Hall, Upper Saddle River, NJ, USA, 1993. ISBN: 0-13-126780-9.
- [16] S. Becker, editor. ADAS: Market introduction scenarios and proper realisation. Deliverable D1, RESPONSE 2 – Advanced Driver Assistance Systems: From Introduction Scenarios towards a Code of Practice for Development and Testing, Cologne, Germany, January 21, 2004. Available at: <http://response.adase2.net/>
- [17] F. Biral, M. Da Lio, and E. Bertolazzi. Combining safety margins and user preferences into a driving criterion for optimal control-based computation of reference maneuvers for an ADAS of the next generation. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 36–41, Las Vegas, NV, USA, June 6–8 2005.
- [18] R. Bishop. *Intelligent Vehicle Technology and Trends*. Artech House, Norwood, MA, USA, 2005. ISBN: 1-58053-911-4.
- [19] M. Blanke, R. Izadi-Zamanabadi, S.A. Bøgh, and C.P. Lunau. Fault-tolerant control systems – a holistic view. *Control Engineering Practice*, 5(5):693–702, 1997.
- [20] M. Blanke, M. Staroswiecki, and N. E. Wu. Concepts and methods in fault-tolerant control. In *Proceedings of the 2001 American Control Conference (ACC)*, pages 2606–2620, Arlington, VA, USA, June 25–27, 2001.
- [21] R. Boot, J. Richert, H. Schütte, and A. Rügauer. Automated test of ECUs in a hardware-in-the-loop simulation environment. In *Proceedings of the 1999 IEEE International Symposium on Computer Aided Control System Design*, pages 587–594, Kohala Coast, HI, USA, August 22–27, 1999.

- [22] A. Bose and P. Ioannou. Analysis of traffic flow with mixed manual and intelligent cruise control (ICC) vehicles: Theory and experiments. UCB-ITS-PRR-2001-13, California PATH, Berkeley, CA, USA, April 1, 2001. Available at: <http://www.path.berkeley.edu/PATH/Publications/PDF/PRR/2001/PRR-2001-13.pdf>
- [23] A.W. Bowman and A. Azzalini. *Applied Smoothing Techniques for Data Analysis*. Oxford University Press, Oxford, UK, 1997. ISBN: 978-0-198-52396-3.
- [24] M. Brackstone and M. McDonald. Car-following: a historical review. *Transportation Research Part F: Psychology and Behaviour*, 2(4):181–196, December 1999.
- [25] E. Brockfeld, R.D. Kühne, and P. Wagner. Calibration and validation of microscopic models of traffic flow. *Transportation Research Record: Journal of the Transportation Research Board*, 1934:179–187, 2005.
- [26] J. Broughton and C. Baughan. The effectiveness of antilock braking systems in reducing accidents in Great Britain. *Accident Analysis and Prevention*, 34(3):347–355, May 2002.
- [27] L. Bruel, J.-P. Colinot, E. Adell, A. Várhelyi, M. Dalla Fontana, and M. D’Alessandro. HMI tests in simulator. Deliverable D20.54, European Commission, IP PReVENT, Subproject SASPENCE, Brussels, Belgium, July 4, 2006.
- [28] S.J. Brunson, E.M. Kyle, N.C. Phamdo, and G.R. Preziotti. Alert algorithm development program NHTSA rear-end collision alert algorithm, Final report. DOT HS 809 526, DOT/NHTSA, Washington, DC, USA, 2002. Available at: http://www.nrd.nhtsa.dot.gov/departments/nrd-12/pubs_rev.html
- [29] J.A. Bucklew. *Introduction to Rare Event Simulation*. Springer-Verlag, New York, NY, USA, 2004. ISBN: 978-0-387-20078-1.
- [30] O.M.J. Carsten and F.N. Tate. Intelligent speed adaptation: accident savings and cost-benefit analysis. *Accident Analysis & Prevention*, 37(3):407–416, May 2005.
- [31] C.-Y. Chan, W.-B. Zhang, E.-M. El Koursi, and E. Lemaire. Safety assessment of advanced vehicle control and safety systems (AVCSS): A case study. UCB-ITS-PRR-2001-30, California PATH, Berkeley, CA, USA, 2001. Available at: <http://www.path.berkeley.edu/PATH/Publications/PDF/PRR/2001/PRR-2001-30.pdf>
- [32] J. Chen and R.J. Patton. *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishers, Norwell, MA, USA, 1999. ISBN: 0-7923-8411-3.
- [33] R.H. Chen, H.K. Ng, J.L. Speyer, and D.L. Mingori. Integration of fault detection and identification into a fault tolerant automated highway system: Final report. UCB-ITS-PRR-2002-36, California PATH, Berkeley, CA, USA, 2002. Available at: <http://www.path.berkeley.edu/PATH/Publications/PDF/PRR/2002/PRR-2002-36.pdf>
- [34] H. Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Annals of Mathematical Statistics*, 23:493–507, 1952.

- [35] S. Citelli, G. Alessandretti, and C. Domsch. ISO guidelines recommendations for pre-crash application. Deliverable D03.4, European Commission, CHAMELEON Project, Brussels, Belgium, October 31, 2001. Available at: <http://www.crfproject-eu.org/>
- [36] D. Cody, S. Tan, and A. Garcia. Human driver model development. Final Report UCB-ITS-PRR-2005-21, California PATH, Berkeley, CA, USA, June 2005. Available at: <http://www.path.berkeley.edu/PATH/Publications/PDF/PRR/2005/PRR-2005-21.pdf>
- [37] J. Craig. *Introduction to Robotics, Mechanics and Control*. Addison-Wesley, Boston, MA, USA, 3rd edition, September 2004. ISBN: 0-20110-326-5.
- [38] F. Dabbene, B.T. Polyak, and R. Tempo. On the complete instability of interval polynomials. *Systems & Control Letters*, 56:431–438, 2007.
- [39] Delphi Seat Belt Systems. Product information sheet DD05E0368, Delphi Corporation, Troy, MI, USA, 2005. Available at: <https://ppd.delphi.com/pdf/ppd/safesec/seatbelts.pdf>
- [40] Delphi Smart Cruise Control with Headway Alert & Stop-and-Go. Product information sheet DD05E0021, Delphi Corporation, Troy, MI, USA, 2005. Available at: <https://ppd.delphi.com/pdf/ppd/safesec/stopgo.pdf>
- [41] A. Doi, T. Butsuen, T. Niibe, T. Yakagi, Y. Yamamoto, and H. Seni. Development of a rear-end collision avoidance system with automatic braking control. *JSAE Review*, 15(4):335–340, October 1994.
- [42] C. Dora and M. Phillips. Transport, environment and health. WHO regional publications. European series; No. 89, World Health Organization, Copenhagen, Denmark, 2000. ISBN: 92-890-1356-7. Available at: <http://www.euro.who.int/document/e72015.pdf>
- [43] T. Duong. *Bandwidth Matrices for Multivariate Kernel Density Estimation*. PhD thesis, University of Western Australia, Crawley, Australia, October 1, 2004.
- [44] H.-M. Duringhof, J. Ploeg, and O. Gietelink. VeHIL, hardware-in-the-loop testing of advanced driver assistance systems. In *Proceedings of the 3rd International Conference on Automotive Technologies (ICAT)*, Istanbul, Turkey, November 17, 2006.
- [45] J. Eelkema, W. Vink, and E. van den Tillaart. ADVANCE, a modular vehicle simulation environment in MATLAB/SIMULINK. In *Proceedings of the MathWorks 3rd International Automotive Conference (IAC)*, Stuttgart, Germany, June 2002.
- [46] D. Ehmanns and H. Spannheimer. Roadmap. Deliverable D2D, Roadmap Development, Version 1.0, European Commission – Advanced Driver Assistance Systems in Europe (ADASE-II), Brussels, Belgium, July 1, 2004. Available at: <http://www.adase2.net/>
- [47] eSafety Support. eSafety Compendium. Brussels, Belgium, May 2006. Available at: http://ec.europa.eu/information_society/activities/esafety/doc/esafety_library/esafety_compendium_final.pdf

- [48] P.R. Escobal. *Methods of Orbit Determination*. John Wiley & Sons, Inc., New York, NY, USA, 1965.
- [49] European Commission. White Paper – European transport policy for 2010: time to decide. Office for Official Publications of the European Communities, Luxembourg, September 21, 2001. ISBN: 92-894-0341-1. Available at: http://ec.europa.eu/transport/white_paper/documents/doc/lb_texte_complet_en.pdf
- [50] European Commission. European road safety action programme: Halving the number of road accident victims in the European Union by 2010: A shared responsibility. COM(2003) 311 final, Brussels, Belgium, June 2, 2003. Available at: <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=COM:2003:0311:FIN:EN:PDF>
- [51] European Commission. On the harmonisation of the 24 GHz range radio spectrum band for the time-limited use by automotive short-range radar equipment in the community. Document number C(2005)34, Brussels, Belgium, January 25, 2005. Available at: http://eur-lex.europa.eu/LexUriServ/site/en/oj/2005/l_021/l_02120050125en00150020.pdf
- [52] European Commission. Cost-benefit assessment and prioritisation of vehicle safety technologies. Final report TREN-ECON2-002, Brussels, Belgium, January 19, 2006. Available at: http://ec.europa.eu/transport/roadsafety_library/publications/vehicle_safety_technologies_final_report.pdf
- [53] European Commission. Regulation (EC) No 715/2007 of the European Parliament and of the Council of 20 June 2007 on type approval of motor vehicles with respect to emissions from light passenger and commercial vehicles (Euro 5 and Euro 6) and on access to vehicle repair and maintenance information. Brussels, Belgium, June 29, 2007. Available at: <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2007:171:0001:0016:EN:PDF>
- [54] European Commission. Results of the review of the Community Strategy to reduce CO₂ emissions from passenger cars and light-commercial vehicles. Communication from the Commission to the Council and the European Parliament, COM(2007) 19 final, Brussels, Belgium, February 7, 2007. Available at: http://ec.europa.eu/environment/co2/pdf/com_2007_19_en.pdf
- [55] J. Eyre, D. Yanakiev, and I. Kanellakopoulos. A simplified framework for string stability analysis of automated vehicles. *International Journal of Vehicle System Dynamics*, 30(5):375–405, 1998.
- [56] D.B. Fambro, R.J. Koppa, D.L. Picha, and K. Fitzpatrick. Driver perception-brake response in stopping sight distance situations. *Transportation Research Record: Journal of the Transportation Research Board*, 1628:1–7, 1998.
- [57] P. Fancher, Z. Bareket, and R. Ervin. Human-centered design of an ACC-with-braking and forward-crash-warning system. *International Journal of Vehicle System Dynamics*, 36(2/3):203–223, 2001.
- [58] P. Fancher, R. Ervin, J. Sayer, M. Hagan, S. Bogard, Z. Bareket, M. Mefford, and J. Haugen. Intelligent cruise control field operational test. Final report DOT HS

- 808 849, DOT/NHTSA, Washington, DC, USA, May 1998. Available at: <http://www-nrd.nhtsa.dot.gov/pdf/nrd-12/icc1998.pdf>
- [59] P. Fancher, H. Peng, Z. Bareket, C. Assaf, and R. Ervin. Evaluating the influences of adaptive cruise control systems on the longitudinal dynamics of strings of highway vehicles. In *Proceedings of 17th IAVSD Symposium: Dynamics of Vehicles on Roads and Tracks, Supplement to the International Journal of Vehicle System Dynamics*, volume 37, pages 125–136, Copenhagen, Denmark, August 20–24, 2001.
- [60] J. Farrell and M. Barth. Integration of GPS-aided INS into AVCSS. UCB-ITS-PRR-2000-22, California PATH, Berkeley, CA, USA, 2000. Available at: <http://www.path.berkeley.edu/PATH/Publications/PDF/PRR/2000/PRR-2000-22.pdf>
- [61] C. Fielding, A. Vargas, S. Bennani, and M. Selier, editors. *Advanced Techniques for Clearance of Flight Control Laws*. Springer-Verlag, Berlin, Germany, 2002. ISBN: 3-540-44054-2.
- [62] M. Flament. Integrated passive and active safety solutions. In *Proceedings of the 5th European MADYMO User Meeting*, Cambridge, UK, September 27, 2005.
- [63] N. Floudas, M. Tsogas, A. Amditis, and H. Weigel. Positioning and path prediction for scenario assessment of safe speed system. In *Proceedings of the 14th World Congress on Intelligent Transport Systems and Services (ITS)*, Beijing, P.R. China, October 9–13, 2007.
- [64] H. Fölster, F. Rohling. Data association and tracking for automotive radar networks. *IEEE Transactions on Intelligent Transportation Systems*, 6(4):370–377, December 2005.
- [65] K. Fujita, H. Fujinami, K. Moriizumi, T. Enomoto, R. Kachu, and H. Kato. Development of pre-crash safety system. In *Proceedings of the 18th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, Nagoya, Japan, May 19–22, 2003. Paper number: 544.
- [66] K. Fürstenberg and R. Schulz. Laserscanner für Fahrerassistenzsysteme. *Automobil-technische Zeitschrift*, 107(9):718–727, September 2005. In German.
- [67] K.C. Fürstenberg, P. Baraud, G. Caporaletti, S. Citelli, Z. Eitan, U. Lages, and C. Lavergne. Development of a pre-crash sensorial system – the CHAMELEON project. In *Proceedings of Joint VDI/VW Congress “Vehicle Concepts for the 2nd Century of Automotive Technology”*, Wolfsburg, Germany, November 21–23, 2001.
- [68] T. Galla, A. Schedl, T. Thurner, and J. Spohr. X-by-wire, safety related fault-tolerant systems in vehicles. Final report, Document number: XByWire-DB-6/6-24, European Commission – Brite-EuRam III, Brussels, Belgium, 1998. Available at: <http://www.vmars.tuwien.ac.at/projects/xbywire/docs/final.doc>
- [69] K. Gerlach. New results in importance sampling. *IEEE Transactions on Aerospace and Electronic Systems*, 35(3):917–925, July 1999.
- [70] J.J. Gertler. *Fault Detection and Diagnosis in Engineering Systems*. Marcel Dekker, New York, NY, USA, 1998. ISBN: 0-8247-9427-3.

- [71] O. Gietelink, B. De Schutter, and M. Verhaegen. Probabilistic validation of advanced driver assistance systems. In *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, July 3–8, 2005. Paper number: 4254.
- [72] O. Gietelink, M. Lammers, and S. Jansen. SASPENCE validation results. Deliverable D20.70, European Commission, IP PReVENT, Subproject SASPENCE, Brussels, Belgium, 2007. To be published.
- [73] O. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen. Testing advanced driver assistance systems for fault management with the VEHIL test facility. In *Proceedings of the 7th International Symposium on Advanced Vehicle Control (AVEC)*, Arnhem, The Netherlands, August 23–27, 2004.
- [74] O. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen. Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations. *International Journal of Vehicle System Dynamics*, 44(7):569–590, July 2006.
- [75] O.J. Gietelink, B. De Schutter, and M. Verhaegen. A probabilistic approach for validation of advanced driver assistance systems. In *Proceedings of the 8th TRAIL Congress*, pages 1–17, Rotterdam, The Netherlands, November 23, 2004.
- [76] O.J. Gietelink, B. De Schutter, and M. Verhaegen. Probabilistic approach for validation of advanced driver assistance systems. *Transportation Research Record: Journal of the Transportation Research Board*, (1910):20–28, 2005.
- [77] O.J. Gietelink, B. De Schutter, and M. Verhaegen. Adaptive importance sampling for probabilistic validation of advanced driver assistance systems. In *Proceedings of the 2006 American Control Conference (ACC)*, pages 4002–4007, Mineapolis, MN, USA, June 14–16, 2006. Paper number: ThC14.5.
- [78] O.J. Gietelink, B. De Schutter, and M. Verhaegen. Adaptive importance sampling for validation of advanced driver assistance systems. 2007. Submitted to a journal.
- [79] O.J. Gietelink, R. Hallouzi, B. De Schutter, and M. Verhaegen. Fault management for automated longitudinal vehicle control. 2007. Submitted to a journal.
- [80] O.J. Gietelink, R. Hallouzi, J. Ploeg, B. De Schutter, and M. Verhaegen. Cooperative driving using environment sensors and vehicle-to-vehicle communication. 2007. Submitted to a journal.
- [81] O.J. Gietelink, P. Lemmen, F. Leneman, M. Shah, W. Kosiak, and R. Cashler. Validation of pre-crash systems with vehicle hardware-in-the-loop experiments. *International Journal of Vehicle Safety*, 2(1/2):185–205, 2007.
- [82] O.J. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen. VEHIL: Test facility for fault management testing of advanced driver assistance systems. In *Proceedings of the 10th World Congress on Intelligent Transport Systems and Services (ITS)*, Madrid, Spain, November 16–20, 2003. Paper number: 2639.
- [83] O.J. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen. VEHIL: a test facility for validation of fault management systems for advanced driver assistance systems. In *Proceedings of the First IFAC Symposium on Advances in Automotive Control*, pages 410–415, Salerno, Italy, April 19–23, 2004.

- [84] O.J. Gietelink, B. De Schutter, and M. Verhaegen. A probabilistic approach for validation of advanced driver assistance systems. In *Proceedings of the 84th Annual Meeting of the Transportation Research Board*, Washington, DC, USA, January 9–13, 2005. Paper number: 05-1524.
- [85] O.J. Gietelink, D.J. Verburg, K. Labibes, and A.F. Oostendorp. Pre-crash system validation with PRESCAN and VEHIL. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 913–918, Parma, Italy, June 14–17, 2004.
- [86] P.G. Gipps. Behavioural car-following model for computer simulation. *Transportation Research B*, 15B(2):105–111, 1981.
- [87] A.R. Girard, J. Borges de Sousa, J.A. Misener, and J.K. Hedrick. A control architecture for integrated cooperative cruise control and collision warning systems. In *Proceedings of the IEEE Conference on Decision and Control*, Orlando, FL, USA, December 2001.
- [88] D.N. Godbole, R. Sengupta, J. Misener, N. Kourjanskaia, and J.B. Michael. Benefit evaluation of crash avoidance systems. *Transportation Research Record: Journal of the Transportation Research Board*, (1621):1–9, 1998.
- [89] J. Golias, G. Yannis, and C. Antoniou. Classification of driver-assistance systems according to their impact on road safety and traffic efficiency. *Transport Reviews*, 22(2):179–196, April–June 2002.
- [90] H.P. Groll and J. Detlefsen. History of automotive anticollision radars and final experimental results of a MM-wave car radar developed by the Technical University of Munich. *IEEE Aerospace and Electronic Systems Magazine*, 12(8):15–19, August 1997.
- [91] R.C. Hammett and P.S. Babcock. Achieving 10^{-9} dependability with drive-by-wire systems. *SAE Technical Paper Series*, 2003-01-1290, 2003.
- [92] W. Härdle, M. Müller, S. Sperlich, and A. Werwatz. *Nonparametric and Semiparametric Models*. Springer-Verlag, Berlin, Germany, May 2004. ISBN: 978-3-540-20722-1.
- [93] A. Hart. European Telematics & ITS – Growth strategies to unlock the safety benefits of ADAS. SBD/TEL/1340, Secured By Design Ltd, Cosgrove, UK, May 2007. Available at: http://www.english.ptv.de/cgi-bin/mobility/mob_report.pl
- [94] K. Hartman and J. Strasser. Saving lives through advanced vehicle safety technology: Intelligent vehicle initiative. Final Report FHWA-JPO-05-057, Federal Highway Administration, Washington, DC, USA, September 2005. Available at: http://www.itsdocs.fhwa.dot.gov/JPODOCS/REPTS_PR/14153_files/ivi.pdf
- [95] W. Helly. Simulation of bottlenecks in single lane traffic flow. In *Proceedings of the Symposium on Theory of Traffic Flow*, pages 207–238, New York, NY, USA, 1959.
- [96] R. Hermann and A. Krener. Nonlinear controllability and observability. *IEEE Transactions on Automatic Control*, 22(5):728–740, October 1977.

- [97] Honda Motor Company, Ltd. Honda develops world's first 'collision mitigation brake system' (CMS) for predicting rear-end collisions and controlling brake operations, May 20, 2003. Press release. Available at: <http://world.honda.com/news/2003/4030520.html>
- [98] S.P. Hoogendoorn. *Multiclass Continuum Modelling of Multilane Traffic Flow*. TRAIL Thesis Series nr. T1999/5, The Netherlands TRAIL Research School, Delft University of Technology, Delft, The Netherlands, September 20, 1999. ISBN: 90-407-1931-4.
- [99] S.P. Hoogendoorn, S. Ossen, and M. Schreuder. Empirics of multianticipative car-following behavior. *Transportation Research Record: Journal of the Transportation Research Board*, (1965):112–120, 2006.
- [100] C. Hôte. Abstract interpretation techniques for software testing. *Business Briefing: Global Automotive Manufacturing & Technology*, pages 1–7, 2002.
- [101] A.S. Howell. *Nonlinear Observer Design and Fault Diagnostics for Automated Longitudinal Vehicle Control*. PhD thesis, University of California at Berkeley, Berkeley, CA, USA, 2002.
- [102] Ibeo Automobile Sensor GmbH. Bad weather performance – multi-target capability facilitates operation in poor weather, 2007. Available at: http://www.ibeo-as.com/english/technology_d_badweather.asp
- [103] M. Ichinose, A. Yokoyama, T. Nishigaito, H. Saito, and N. Ueki. Development of hardware-in-the-loop simulator for adaptive cruise control system. In *Proceedings of the 6th International Symposium on Advanced Vehicle Control (AVEC)*, Hiroshima, Japan, September 9–13, 2002. Paper number: 98.
- [104] R.L. Iman, J.E. Campbell, and J.C. Helton. An approach to sensitivity analysis of computer models. Part I – Introduction, input, variable selection and preliminary variable assessment. *Journal of Quality Technology*, 13:174–183, July 1981.
- [105] Integrated Project PReVENT – Preventive and Active Safety Applications. Available at: <http://www.prevent-ip.org/>
- [106] International Organization for Standardization (ISO). Transport information and control systems — adaptive cruise control systems — performance requirements and test procedures. ISO Standard 15622, First edition 2002-10-15, Geneva, Switzerland, 2002. Available at: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=27834
- [107] International Organization for Standardization (ISO). Transport information and control systems — forward vehicle collision warning systems — performance requirements and test procedures. ISO Standard 15623, First edition 2002-10-01, Geneva, Switzerland, 2002. Available at: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=27835
- [108] International Road Traffic Database (IRTAD). Available at: <http://www.irtad.com/>

- [109] P.A. Ioannou and M. Stefanovic. Evaluation of ACC vehicles in mixed traffic: lane change effects and sensitivity analysis. *IEEE Transactions on Intelligent Transportation Systems*, 6(1):79–89, March 2005.
- [110] R. Isermann. Fault diagnosis of machines via parameter estimation and knowledge processing – tutorial paper. *Automatica*, 29(4):815–835, 1993.
- [111] R. Isermann. Diagnosis methods for electronic controlled vehicles. *International Journal of Vehicle System Dynamics*, 36(2–3):77–117, 2001.
- [112] R. Isermann. *Mechatronic Systems*. Springer-Verlag, London, 2003. ISBN: 1-85233-693-5.
- [113] R. Isermann and P. Ballé. Trends in the application of model-based fault detection and diagnosis of technical processes. *Control Engineering Practice*, 5(5):709–719, May 1997.
- [114] R. Isermann, J. Schaffnit, and S. Sinsel. Hardware-in-the-loop simulation for the design and testing of engine-control systems. *Control Engineering Practice*, 7(5):643–653, May 1999.
- [115] R. Isermann, R. Schwarz, and S. Stölzl. Fault-tolerant drive-by-wire systems. *IEEE Control Systems Magazine*, 22(5):64–81, October 2002.
- [116] K. Isomoto, T. Niibe, T. Suetomi, and T. Butsuen. Development of a lane-keeping system for lane departure avoidance. In *Proceedings of the 2nd World Congress on Intelligent Transport Systems and Services (ITS)*, pages 1266–1271, Yokohama, Japan, November 1995.
- [117] R. Izadi-Zamanabadi. *Fault-tolerant Supervisory Control – System Analysis and Logic Design*. Thesis number: 0908-1208, Aalborg University, Aalborg, Denmark, September 1999.
- [118] H.M. Jagtman, V.A.W.J. Marchau, and T. Heijer. Current knowledge on safety impacts of Collision Avoidance Systems (CAS). In P.M. Herder and W.A.H. Thissen, editors, *Proceedings of the 5th International Conference on Technology, Policy and Innovation*, Delft, June 26–29, 2001. Paper number: 1152.
- [119] J. Jansson, J. Johansson, and F. Gustafsson. Decision making for collision avoidance systems. *SAE Technical Paper Series*, 2002-01-0403, March 2002.
- [120] P.H. Jesty, J.F. Gaillet, J. Giezen, G. Franco, I. Leighton, and H.J. Schultz. Guidelines for the development and assessment of intelligent transport system architectures. Project TR 1101, Deliverable DSA2.3, Version 1.0, European Commission, CONVERGE project, May 18, 1998. Available at: ftp://ftp.cordis.europa.eu/pub/telematics/docs/tap_transport/converge_dsa2.3.pdf
- [121] P.H. Jesty, K.M. Hobley, R. Evans, and I. Kendall. Safety analysis of vehicle-based systems. In *Aspects of Safety Management: Proceedings of the Ninth Safety Critical Systems Symposium*, pages 90–110, 2000.

- [122] W.-D. Jonner, H. Winner, L. Dreilich, and E. Schunck. Electrohydraulic brake system – the first approach to brake-by-wire technology. *SAE Technical Paper Series*, 960991:221–228, 1999.
- [123] S. Juliana, Q. Chu, J. Mulder, and T. van Baten. Flight envelope clearance of atmospheric re-entry module with flight control. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, RI, USA, August 16–19, 2004. Paper number: AIAA-2004-5170.
- [124] R.K. Jurgen, editor. *Object Detection, Collision Warning and Avoidance Systems*, volume 1. SAE International, Warrendale, PA, USA, May 1998. ISBN: 0-7680-0226-3.
- [125] R.K. Jurgen, editor. *Adaptive Cruise Control*. Number PT-132. SAE International, Warrendale, PA, USA, 2006. ISBN: 0-7680-1792-0.
- [126] R.K. Jurgen, editor. *Object Detection, Collision Warning, and Avoidance Systems*, volume 2. SAE International, Warrendale, PA, USA, 2006. ISBN: 0-7680-1792-0.
- [127] P. Kågeson. Cycle-beating and the EU test cycle for cars. T&E 98/3, European Federation for Transport and Environment, Brussels, Belgium, November 1998. Available at: <http://www.transportenvironment.org/Downloads-req-getit-lid-17.html>
- [128] K. Kamiji and H. Akaba. Research of an advanced seat belt system. In *Proceedings of the 18th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, Nagoya, Japan, May 19–22, 2003. Paper number: 480.
- [129] B. van Kampen. Rear end or chain accidents: case study assessing the European road safety problem: an exploitation study of the CARE database ASTERYX. Contract No P TREN/E3/SI2.273205, SWOV Institute for Road Safety Research, Leidschendam, The Netherlands, October 2003. Available at: http://ec.europa.eu/transport/roadsafety_library/publications/asteryx/cs3_report.pdf
- [130] S. Kanev. *Robust Fault-tolerant Control*. PhD thesis, University of Twente, Enschede, The Netherlands, March 12, 2004. ISBN: 90-9017903-8.
- [131] R. Karlsson, J. Jansson, and F. Gustafsson. Model-based statistical tracking and decision making for collision avoidance application. LiTH-ISY-R-2599, Department of Electrical Engineering, Linköping University, Linköping, Sweden, March 2004. Available at: <http://www.control.isy.liu.se/research/reports/2004/2599.pdf>
- [132] R.J. Kiefer, M.T. Cassar, C.A. Flannagan, D.J. LeBlanc, M.D. Palmer, R.K. Deering, and M.A. Shulman. Forward Collision Warning Requirements Project Task 1. Final Report DOT HS 809 574, DOT/NHTSA, Washington, DC, USA, January 2003. Available at: http://www-nrd.nhtsa.dot.gov/departments/nrd-12/pubs_rev.html
- [133] R.J. Kiefer, D.J. LeBlanc, M.D. Palmer, J. Salinger, R.K. Deering, and M.A. Shulman. Development and validation of functional definitions and evaluation procedures for collision warning/avoidance systems. Final Report DOT HS 808 964, DOT/NHTSA, Washington, DC, USA, August 1999. Available at: http://www-nrd.nhtsa.dot.gov/pdf/nrd-12/acas/HS808964_Report-1999-08.pdf

- [134] K. Kodaka, M. Otabe, Y. Urai, and H. Koike. Rear-end collision avoidance assist system. In *Proceedings of the 18th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, Nagoya, Japan, May 19–22, 2003. Paper number: 405.
- [135] H. Kopetz. *Real-Time Systems*. Kluwer Academic Press, Boston, MA, USA, 1997. ISBN: 0-7923-9894-7.
- [136] L.J.J. Kusters, O.J. Gietelink, J. van Hoof, and P.P.M. Lemmen. Evaluation of advanced driver assistance systems with the VEHIL test facility – experiences and future developments at TNO Automotive. In *Proceedings of the 21st International VDI/VW Conference “Integrated safety and driver assistance systems”*, Wolfsburg, Germany, October 27–29, 2004.
- [137] L.J.J. Kusters, R.J.A. Kleuskens, D.J. Verburg, and A.C.M. van der Knaap. System for performing tests on intelligent road vehicles. US Patent US 2003/0183023 A1, October 2, 2003.
- [138] K. Labibes, Z. Papp, A.H.C. Thean, P.P.M. Lemmen, M. Dorrepaal, and F.J.W. Lene-man. An Integrated Design and Validation Environment for Intelligent Vehicle Safety Systems (IVSS). In *Proceedings of the 10th World Congress on Intelligent Transport Systems and Services (ITS)*, Madrid, Spain, November 16–20, 2003. Paper 2731.
- [139] R. LaGuerra. Automotive radar/lidar systems – a component-level market analysis of radar, lidar, ultrasonic, and optics-based automotive safety systems. ABI Research, Oyster Bay, NY, USA, 2004. Available at: http://www.abiresearch.com/products/market_research/Automotive_Obstacle_Detection_Systems
- [140] J. Langheim, A. Buchanan, U. Lages, and M. Wahl. CARSENSE – new environment sensing for advanced driver assistance systems. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, Tokyo, Japan, May 14–17 2001.
- [141] J.C. Laprie, editor. *Dependability: Basic Concepts and Terminology*. Springer-Verlag, Vienna, Austria, 1992. ISBN: 3-211-82296-8.
- [142] G.D. Lee and S.W. Kim. A longitudinal control system for a platoon of vehicles using a fuzzy-sliding mode algorithm. *Mechatronics*, 12:97–118, 2002.
- [143] K. Lee and H. Peng. Identification and verification of a longitudinal human driving model for collision warning and avoidance systems. *International Journal of Vehicle Autonomous Systems*, 2(1/2):3–17, 2004.
- [144] K. Lee and H. Peng. Evaluation of automotive forward collision warning and collision avoidance algorithms. *International Journal of Vehicle System Dynamics*, 43(10):735–751, October 2005.
- [145] P. Lemmen, O.J. Gietelink, M. Shah, C. Parenteau, W. Kosiak, and R. Cashler. Development of a pre-crash system using the VEHIL test facility. In *Proceedings of the 19th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, Washington, DC, USA, June 6–9, 2005. Paper number: 05-0322.

- [146] P.P.M. Lemmen, O.J. Gietelink, and R.J.G. Haan. Pre-crash system design and evaluation using PRESCAN, VEHIL and MADYMO. In *Proceedings of the 11th World Congress on Intelligent Transport Systems and Services (ITS)*, Nagoya, Japan, October 18–22, 2004.
- [147] N.G. Leveson. *Safeware: System Safety and Computers*. Addison-Wesley, Boston, MA, USA, 1995. ISBN: 978-0-201-11972-5.
- [148] W. Levine and M. Athans. On the optimal error regulation of a string of moving vehicles. *IEEE Transactions on Automatic Control*, 11(3):355–361, July 1966.
- [149] J. Li, F. Yu, J.-W. Zhang, J.-Z. Feng, and H.-P. Zhao. The rapid development of a vehicle electronic control system and its application to an antilock braking system based on hardware-in-the-loop simulation. *Proceedings of the Institution of Mechanical Engineers – Part D: Journal of Automobile Engineering*, 216:95–105, 2002.
- [150] C.-Y. Liang and H. Peng. Optimal adaptive cruise control with guaranteed string stability. *International Journal of Vehicle System Dynamics*, 31(4–5):313–330, November 1999.
- [151] A. Lie, C. Tingvall, M. Krafft, and A. Kullgren. The effectiveness of ESP (Electronic Stability Program) in reducing real life accidents. *Traffic Injury Prevention*, 5(1):37–41, March 2004.
- [152] X.-Y. Lu and J.K. Hedrick. A panoramic view of fault management for longitudinal control of automated vehicle platooning. In *Proceedings of the ASME International Mechanical Engineering Congress and Expositions (IMECE)*, number IMECE2002-DSC-32106, New Orleans, LA, USA, November 17–22, 2002.
- [153] X.-Y. Lu and J.K. Hedrick. Practical string stability for longitudinal control of automated vehicles. In *Proceedings of 18th IAVSD Symposium: Dynamics of Vehicles on Roads and Tracks, Supplement to the International Journal of Vehicle System Dynamics*, volume 41, pages 577–586, Atsugi, Kanagawa, Japan, August 24–30, 2003.
- [154] X.-Y. Lu, J.K. Hedrick, and M. Drew. ACC/CACC – control design, stability and robust performance. In *Proceedings of the American Control Conference*, pages 4327–4332, Anchorage, AK, USA, May 8–10, 2002.
- [155] G. Luderer. Greenhouse gas emission trends and projections in Europe 2006. EEA Report No 9/2006, European Environment Agency, Copenhagen, Denmark, 2006. ISBN: 92-9167-885-6. Available at: http://reports.eea.europa.eu/eea_report_2006_9/en/eea_report_9_2006.pdf
- [156] N. Madras. *Lectures on Monte Carlo Methods*. American Mathematical Society, Providence, RI, USA, 2002. ISBN: 0-8218-2978-5.
- [157] MADYMO website. Available at: <http://www.madymo.com/>
- [158] T. Maki, T. Asai, and J. Kajzer. Development of future pedestrian protection technologies. In *Proceedings of the 18th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, Nagoya, Japan, May 19–22, 2003. Paper number: 165.

- [159] V. Marchau. *Technology Assessment of Automated Vehicle Guidance – Prospects for automated driving implementation*. TRAIL Thesis Series nr. 2000/1, The Netherlands TRAIL Research School, Delft University of Technology, Delft, The Netherlands, 2000. ISBN: 90-407-2015-0.
- [160] W. van der Mark. *Stereo and Colour Vision Techniques for Autonomous Vehicle Guidance*. ASCI dissertation series number 145, University of Amsterdam, Amsterdam, The Netherlands, 2007. ISBN: 978-90-5986-242-5.
- [161] G. Marsden, M. McDonald, and M. Brackstone. Towards an understanding of adaptive cruise control. *Transportation Research Part C: Emerging Technologies*, 9(1):33–51, February 2001.
- [162] M.-M. Meinecke, R. Holze, M. Gonter, T. Wohllebe, R. Mende, and R. Petelka. Side-pre-crash sensing system for automatic vehicle height level adaptation. In *Proceedings of the 3rd International Workshop on Intelligent Transportation (WIT)*, Hamburg, Germany, March 14–15, 2006.
- [163] J.B. Michael, A.C. Segal, and S. Patwardhan. Validation of software testing results for real-time vehicle control software. In *SAE Future Transportation Technology Conference*, Costa Mesa, California, USA, August 7–10, 1995.
- [164] T. Miyazaki. Promotion of the program of advanced safety vehicles for the 21st century. In *Proceedings of the 2nd World Congress on Intelligent Transport Systems and Services (ITS)*, pages 2612–2617, Yokohama, Japan, November 9–11, 1995.
- [165] D.C. Montgomery. *Design and Analysis of Experiments*. John Wiley & Sons, Inc., New York, NY, USA, 5th edition, 2001. ISBN: 0-471-31649-0.
- [166] R. Moritz. Pre-crash sensing – functional evolution based on short range radar sensor platform. *SAE Technical Paper Series*, 00IBECD-11, 2000.
- [167] P.L.J. Morsink. CarTALK2000: Development of a co-operative ADAS based on vehicle-to-vehicle communication. In *Proceedings of the 10th World Congress on Intelligent Transport Systems and Services (ITS)*, Madrid, Spain, 2003. Paper number: 2575.
- [168] P.L.J. Morsink and O.J. Gietelink. Preliminary design of an application for CBLC in the CarTALK2000 project: Safe, comfortable and efficient driving based upon inter-vehicle communication. In *Proceedings of the e-Safety Conference*, Lyon, France, September 16–18, 2002.
- [169] R. Motwani. *Randomized Algorithms*. Cambridge University Press, New York, NY, USA, 1995. ISBN: 0-521-47465-5.
- [170] W.G. Najm and D.L. Smith. Modeling driver response to lead vehicle decelerating. *SAE Technical Paper Series*, 2004-01-0171:1–10, 2004.
- [171] W.G. Najm, D.L. Smith, and A.H. Lam. Modeling car-following performance for vehicle safety applications. In *Proceedings of the 82nd Annual Meeting of the Transportation Research Board*, number TRB 03-4382, Washington, DC, USA, January 12–16, 2003.

- [172] W.G. Najm, M.D. Stearns, H. Howarth, J. Koopmann, and J. Hitz. Evaluation of an automotive rear-end collision avoidance system. DOT/NHTSA DOT HS 810 569, Washington, DC, USA, March 2006. Available at: <http://www-nrd.nhtsa.dot.gov/pdf/nrd-12/HS910569.pdf>
- [173] National Automotive Sampling System (NASS) – Crashworthiness Data System, Analytical User’s Manual. 2003 File, Washington, DC, USA, 2003. Available at: <http://www-nrd.nhtsa.dot.gov/pdf/nrd-30/NCSA/Manuals/CDSAUM03.pdf>
- [174] National Automotive Sampling System (NASS) Crashworthiness Data System (CDS), 2004. National Highway Traffic Safety Administration. Available at: <ftp://ftp.nhtsa.dot.gov/NASS>
- [175] Netherlands Organisation for Applied Scientific Research (TNO). Available at: <http://www.tno.nl/>
- [176] NHTSA. Automotive collision avoidance system field operational test (ACAS FOT). Final Program Report DOT HS 809 866, DOT/NHTSA, Washington, DC, USA, May 2005. Available at: http://www-nrd.nhtsa.dot.gov/departments/nrd-12/pubs_rev.html
- [177] H.B. Pacejka. *Tyre and Vehicle Dynamics*. Butterworth-Heinemann, Oxford, UK, 2002. ISBN: 978-0-750-65141-7.
- [178] R. Pack, J. Koopmann, H. Yu, and W.G. Najm. Pre-crash sensing countermeasures and benefits. In *Proceedings of the 19th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, Washington, DC, USA, June 6–9, 2005. Paper number: 05-0202.
- [179] Z. Papp, M. Dorrepaal, and D.J. Verburg. Distributed hardware-in-the-loop simulator for autonomous continuous dynamical systems with spatially constrained interactions. In *Proceedings of the 11th IEEE/ACM International Workshop on Parallel and Distributed Real-Time Systems*, Nice, France, April 22–26, 2003.
- [180] Z. Papp, K. Labibes, A.H.C. Thean, and M.G. van Elk. Multi-agent based HIL simulator with high fidelity virtual sensors. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 213–219, Columbus, OH, USA, June 9–11, 2003.
- [181] Z. Parseghian, A.C. Stein, and D. Ziedman. Field evaluation of a Nissan laser collision avoidance system. DOT HS 808 375, DOT/NHTSA, Washington, DC, USA, January 1989.
- [182] R.J. Patton. Fault detection and diagnosis in aerospace systems using analytical redundancy. *IEE Computing & Control Engineering Journal*, 2(3):127–136, 1991.
- [183] R.J. Patton. Fault-tolerant control systems: The 1997 situation. In *Proceedings of the 3rd IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (SAFEPROCESS’97)*, pages 1029–1051, Hull, UK, August 26–28, 1997.
- [184] R.J. Patton and J. Chen. Review of parity space approaches to fault diagnosis for aerospace systems. *Journal of Guidance, Control and Dynamics*, 17(2):278–285, March–April 1994.

- [185] R.J. Patton and J. Chen. Observer-based fault detection and isolation: Robustness and applications. *Control Engineering Practice*, 5(5):671–682, 1997.
- [186] R.J. Patton, P.M. Frank, and R.N. Clark. *Issues of Fault Diagnosis for Dynamic Systems*. Springer-Verlag, London, UK, 2000. ISBN: 3-54019968-3.
- [187] M. Peden, R. Scurfield, D. Sleet, D. Mohan, A.A. Hyder, E. Jarawan, and C. Mathers. World report on road traffic injury prevention. World Health Organization, Geneva, Switzerland, 2004. ISBN: 92-4-156260-9. Available at: http://www.who.int/violence_injury_prevention/publications/road_traffic/world_report/en/
- [188] M. Persson, F. Botling, E. Hesslow, and R. Johansson. Stop and go controller for adaptive cruise control. In *Proceedings of the 1999 IEEE International Conference on Control Applications*, volume 2, pages 1692–1697, Kohala Coast, HI, USA, August 22–27, 1999.
- [189] J.R. Pimentel, editor. *Safety-Critical Automotive Systems*. Number PT-103. SAE International, Warrendale, PA, USA, August 2006. ISBN: 0-7680-1243-0.
- [190] J. Ploeg, O.J. Gietelink, and D.J. Verburg. Experimental evaluation of a communication based cooperative driving algorithm. In *Proceedings of the 13th World Congress on Intelligent Transport Systems and Services (ITS)*, London, UK, October 8–12, 2006.
- [191] J. Ploeg, Z. Papp, R.C. van de Pijpekamp, S.E. Skolnik, and E.A.C. van den Eijnden. ControlCIT – a control design and implementation toolbox for automatic vehicle guidance. In *Proceedings of the 7th International Symposium on Advanced Vehicle Control (AVEC)*, Arnhem, The Netherlands, August 23–27, 2004.
- [192] J. Ploeg, A.C.M. van der Knaap, and D.J. Verburg. ATS/AGV, design, implementation and evaluation of a high performance AGV. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, volume 1, pages 127–134, Versailles, France, June 18–20, 2002.
- [193] J. Ploeg, J.P.M. Vissers, and H. Nijmeijer. Control design for an overactuated wheeled mobile robot. In *Proceedings of the 4th IFAC Symposium on Mechatronic Systems (MECHATRONICS 2006)*, Heidelberg, Germany, September 12–14, 2006.
- [194] W. Prestl. The BMW active cruise control ACC. *SAE Technical Paper Series*, 2000-01-0344, 2000.
- [195] A. Pretschner, M. Broy, I.H. Kruger, and T. Stauner. Software engineering for automotive systems: A roadmap. In *Proceedings of the 29th ACM/IEEE International Conference on Software Engineering (ICSE)*, pages 55–71, Minneapolis, MN, USA, May 19–27, 2007.
- [196] PREVENT-SASPENCE Consortium. SASPENCE project description – appendix of PREVENT Technical Annex. European Commission, IP PREVENT, Subproject SASPENCE, Brussels, Belgium, 2003. Available at: www.prevent-saspence.org
- [197] R. Rajamani. *Vehicle Dynamics and Control*. Mechanical Engineering Series. Springer-Verlag, New York, NY, USA, 2006. ISBN: 0-387-26396-9.

- [198] G. Reymond, A. Heidet, M. Canry, and A. Kemeny. Validation of Renault's dynamic simulator for adaptive cruise control experiments. In *Proceedings of the Driving Simulation Conference (DSC2000)*, pages 181–192, Paris, France, September 6–8, 2000.
- [199] G. Sala. Safety, legal issues, standards. Deliverable D5.01, European Commission, CHAMELEON Project, Brussels, Belgium, January 28, 2003. Available at: <http://www.crfproject-eu.org/>
- [200] A.J. van der Schaft and H. Schumacher. *An Introduction to Hybrid Dynamical Systems*. Lecture Notes in Control and Information Sciences, 251. Springer-Verlag, London, UK, 2000. ISBN: 978-1-852-33233-4.
- [201] B. Schick, R. Büttner, K. Baltruschat, G. Meier, and H. Jakob. Bewertung der Funktion und Güte von Fahrerassistenzsystemen bei aktivem Bremseneingriff. *Automobil-technische Zeitschrift*, 109(5):414–425, May 4, 2007. In German.
- [202] R. Schöneburg, K.-H. Baumann, and R. Justen. Pre-safe – the next step in the enhancement of vehicle safety. In *Proceedings of the 18th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, Nagoya, Japan, May 19–22, 2003. Paper number: 410.
- [203] A. Schrage. Traffic congestion and accidents. *Regensburger Diskussionsbeiträge zur Wirtschaftswissenschaft*, 419:1–25, November 9, 2006.
- [204] R. Schulz and K. Fürstenberg. Laserscanner for multiple applications in passenger cars and trucks. In *Proceedings of the Conference on Advanced Microsystems for Automotive Applications (AMAA)*, Berlin, Germany, April 25–27 2006.
- [205] J. Schwarz, editor. Annexes to the Code of Practice for the design and evaluation of ADAS. RESPONSE 3 – PREVENT subproject, Brussels, Belgium, October 31, 2006. Available at: http://prevent-ip.org/en/public_documents/deliverables/d112_code_of_practice_for_the_design_and_evaluation_of_adas.htm
- [206] J. Schwarz, editor. Code of Practice for the design and evaluation of ADAS. RESPONSE 3 – PREVENT subproject, Brussels, Belgium, October 31, 2006. Available at: http://prevent-ip.org/en/public_documents/deliverables/d112_code_of_practice_for_the_design_and_evaluation_of_adas.htm
- [207] D.W. Scott. *Multivariate Density Estimation: Theory, Practice and Visualization*. John Wiley & Sons, Inc., New York, NY, USA, 1992. ISBN: 978-0-471-54770-9.
- [208] P. Seiler, B. Song, and J.K. Hedrick. Development of a collision avoidance system. *SAE Technical Paper Series*, 98PC-417, 1998.
- [209] H. Selzle, editor. Mit Abstand Vorne. *Automobil Produktion*, Sonderausgabe Mercedes Benz S-Klasse, November 1998. In German.
- [210] S. Shaheen, D. Heffernan, and G. Leen. A comparison of emerging time-triggered protocols for automotive x-by-wire control networks. *Proceedings of the Institution of Mechanical Engineers – Part D: Journal of Automobile Engineering*, 217:13–22, 2003.

- [211] D. Shefer. Congestion, air pollution, and road fatalities in urban areas. *Accident Analysis & Prevention*, 26(4):501–509, August 1994.
- [212] S. Shladover, J. VanderWerf, M.A. Miller, and N. Kourjanskaia. Development and performance evaluation of AVCSS deployment sequences to advance from today's driving environment to full automation. Final Report for MOU 366 UCB-ITS-PRR-2001-18, California PATH, Berkeley, CA, USA, August 20, 2001. Available at: <http://www.path.berkeley.edu/PATH/Publications/PDF/PRR/2001/PRR-2001-18.pdf>
- [213] S.E. Shladover. Potential contributions of intelligent vehicle/highway systems (IVHS) to reducing transportation's greenhouse gas production. PATH Technical Memorandum 91-04, California PATH, Berkeley, CA, USA, August 1, 1991. Available at: http://www.path.berkeley.edu/PATH/Publications/PDF/TECHMEMOS/TECH_MEMO-91-04.pdf
- [214] S.E. Shladover. The California PATH Program of IVHS research and its approach to vehicle-highway automation. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 347–352, Detroit, MI, USA, June 29 – July 1, 1992.
- [215] S.E. Shladover. Highway electrification and automation. UCB-ITS-PRR-92-17, California PATH, Berkeley, CA, USA, 1992. Available at: <http://www.path.berkeley.edu/PATH/Publications/PDF/PRR/92/PRR-92-17.pdf>
- [216] S.E. Shladover. Review of the state of development of advanced vehicle control systems. *International Journal of Vehicle System Dynamics*, 24(6–7):551–595, 1995.
- [217] S.E. Shladover. Progressive deployment steps leading toward an automated highway system. *Transportation Research Record: Journal of the Transportation Research Board*, (1727):154–161, 2000.
- [218] S.E. Shladover. Modelling and control issues for automated highway systems. *Proceedings of the Institution of Mechanical Engineers – Part I: Journal of Systems & Control Engineering*, 215(4):335–343, August 19, 2001.
- [219] S.E. Shladover. Automated vehicles for highway operations (automated highway systems). *Proceedings of the Institution of Mechanical Engineers – Part I: Journal of Systems & Control Engineering*, 219(11):53–75, February 2005.
- [220] S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control – Analysis and Design*. John Wiley & Sons, Inc., Chichester, UK, 1996. ISBN: 0-471-94330-4.
- [221] S. Slater, editor. Multifunctional automotive radar network (RadarNet). Final Report, Deliverable D40, Version 0.5, European Commission, Brussels, Belgium, November 25, 2004. Available at: http://www.radarnet.org/publications/zip/radarnet_final_report_public.pdf
- [222] Speed management. Report number 75 2006 02 1 P, OECD Transport Research Centre, Paris, France, October 2006. ISBN: 92-821-0377-3. Available at: <http://cemt.org/pub/pubresearch.htm>
- [223] J.S. Stadler. Adaptive importance sampling. *IEEE Journal on Selected Areas in Communications*, 11(3):309–316, April 1993.

- [224] Statistics Netherlands – Centraal Bureau voor de Statistiek (CBS). Index figures traffic density, 2007. Available at: <http://statline.cbs.nl/StatWeb/Start.asp?lp=Search/Search&LA=EN&DM=SLEN>
- [225] R.F. Stengel and L.R. Ray. Stochastic robustness of linear time-invariant control systems. *IEEE Transactions on Automatic Control*, 36(1):82–87, January 1991.
- [226] N. Storey. *Safety-Critical Computer Systems*. Addison-Wesley Longman Ltd, Essex, UK, 1996. ISBN: 0-201-42787-7.
- [227] T. Strobel, A. Serval, C. Coue, and T. Tatschke. Compendium on sensors – state-of-the-art of sensors and sensor data fusion for automotive preventive safety applications. Version 1.0, European Commission, IP PREVENT, Subproject ProFusion, Brussels, Belgium, July 19, 2004. Available at: http://prevent-ip.org/en/public_documents/deliverables/d112_code_of_practice_for_the_design_and_evaluation_of_adas.htm
- [228] SupplierBusiness – AutoBusiness Reports & Databases. Driver assistance systems report. January 8, Stamford, UK, 2007. Available at: http://www.supplierbusiness.com/reports_endpoint.asp?id=53
- [229] S. Suryanarayanan and M. Tomizuka. Fault tolerant lateral control of automated vehicles based on simultaneous stabilization. In *Proceedings of the 1st IFAC Conference on Mechatronic Systems*, pages 899–923, Darmstadt, Germany, September 18–20, 2000.
- [230] D. Swaroop. *String Stability of Interconnected Systems: An Application to Platooning in Automated Highway Systems*. PhD thesis, University of California at Berkeley, Berkeley, CA, USA, 1994.
- [231] D. Swaroop, J.K. Hedrick, C.C. Chien, and P. Ioannou. A comparison of spacing and headway control laws for automatically controlled vehicles. *International Journal of Vehicle System Dynamics*, 8(8):597–625, 1994.
- [232] C.M.J. Tampère. *Human-Kinetic Multiclass Traffic Flow Theory and Modelling – With Application to Advanced Driver Assistance Systems in Congestion*. TRAIL Thesis Series nr. T2004/11, The Netherlands TRAIL Research School, Delft University of Technology, Delft, The Netherlands, December 17, 2004. ISBN: 90-5584-060-2.
- [233] F. Tango, A. Oyaide, and O.J. Gietelink. The SASPENCE (safe-speed and safe-distance) system as a new approach to support drivers: technical evaluation phase. In *Proceedings of the 6th European Congress and Exhibition on Intelligent Transport Systems and Services (ITS)*, Aalborg, Denmark, June 18–20, 2007.
- [234] F. Tango and A. Saroldi. System specifications. Deliverable D20.34, European Commission, IP PREVENT, Subproject SASPENCE, Brussels, Belgium, June 9, 2005. Available at: http://www.prevent-ip.org/en/public_documents/deliverables/d2034_system_specifications.htm
- [235] F. Tango and A. Saroldi. Towards a new approach in supporting drivers function: specifications of the SASPENCE system. In *Proceedings of the 5th European*

- Congress and Exhibition on Intelligent Transport Systems and Services (ITS)*, pages 614–620, Hannover, Germany, September 13–15, 2005.
- [236] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems*. Springer-Verlag, London, UK, 2005. ISBN: 1-85233-524-6.
- [237] R. Tempo and H. Ishii. Monte Carlo and Las Vegas randomized algorithms for systems and control: An introduction. *European Journal of Control*, 13:189–203, 2007.
- [238] A. Thean, M. van Elk, C. Lievers, K. Labibes, W. van der Mark, and J. Kleijweg. Design and validation of a virtual FMCW radar for automotive applications. In *Proceedings of the 5th EUROSIM Congress on Modelling and Simulation*, Paris, France, September 6–10, 2004.
- [239] I. Theis and J. Güldner. Reliability prediction of fault tolerant automotive systems. *SAE Paper Series*, (2000-01-1049):9–15, 2000.
- [240] S. Tokoro, K. Moriizumi, T. Kawasaki, T. Nagao, K. Abe, and K. Fujita. Sensor fusion system for pre-crash safety system. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 945–950, Parma, Italy, June 14–17, 2004.
- [241] R. van Tongeren, O.J. Gietelink, B. De Schutter, and M. Verhaegen. Traffic modelling validation of advanced driver assistance systems. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 1246–1251, Istanbul, Turkey, June 13–15, 2007.
- [242] A. Touran, M.A. Brackstone, and M. McDonald. A collision model for safety evaluation of autonomous intelligent cruise control. *Accident Analysis and Prevention*, 31(5):567–578, September 1999.
- [243] J.R. Treat. A study of precrash factors involved in traffic accidents. *HSRI Research Review*, 10(6):1–35, May – June 1980.
- [244] S. Tsugawa, M. Aoki, A. Hosaka, and K. Seki. A survey of present IVHS activities in Japan. *Control Engineering Practice*, 5(11):1591–1597, November 1997.
- [245] J. Tung. Enhanced test and verification capabilities using model-based design. *SAE Technical Paper Series*, 2006-01-1445, 2006.
- [246] A. Vahidi and A. Eskandarian. Research advances in intelligent collision avoidance and adaptive cruise control. *IEEE Transactions on Intelligent Transportation Systems*, 4(3):143–153, September 2003.
- [247] J. VanderWerf, S.E. Shladover, M.A. Miller, and N. Kourjanskaia. Effects of adaptive cruise control systems on highway traffic flow capacity. *Transportation Research Record: Journal of the Transportation Research Board*, 1800:78–84, 2002.
- [248] P. Venhovens, K. Naab, and B. Adiprasito. Stop and go cruise control. *International Journal of Automotive Technology*, 1(2):61–69, 2000.

- [249] P.J.T. Venhovens, J.H. Bernasch, J.P. Löwenau, H.G. Rieker, and M. Schraut. The application of advanced vehicle navigation in BMW driver assistance systems. *SAE Technical Paper Series*, 1999-01-0490, 1999.
- [250] D.J. Verburg. PreScan, physics in control – a simulation and verification environment for intelligent vehicle systems. Product information sheet, TNO Science and Industry, Helmond, The Netherlands, 2007. Available at: <http://www.prescan-tno.com/fileadmin/files/PreScanBrochure.pdf>
- [251] D.J. Verburg, A.C.M. van der Knaap, and J. Ploeg. VEHIL, developing and testing intelligent vehicles. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, volume 2, pages 537–544, Versailles, France, June 17–21, 2002.
- [252] M. Verhaegen and V. Verdult. *Filtering and System Identification: A Least Squares Approach*. Cambridge University Press, New York, NY, USA, May 2007. ISBN: 978-0521875127.
- [253] L. Verhoeff, D.J. Verburg, H.A. Lupker, and L.J.J. Kusters. VEHIL: A full-scale test methodology for intelligent transport systems and vehicles and subsystems. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 369–375, Detroit, MI, USA, October 2000.
- [254] M. Vidyasagar. *A Theory of Learning and Generalization*. Springer-Verlag, London, UK, 1997. ISBN: 3-540-76120-9.
- [255] M. Vidyasagar. Statistical learning theory and randomized algorithms for control. *IEEE Control Systems Magazine*, 18(6):69–85, December 1998.
- [256] N. Virtanen, A. Schirokoff, J. Luoma, and R. Kulmala. Impacts of an automatic emergency call system on accident consequences. Ministry of Transport and Communications Finland, Finnish R&D Programme on Real-Time Transport Information AINO, Helsinki, Finland, January 6, 2006. Available at: http://ec.europa.eu/information_society/activities/esafety/doc/esafety_forum/ecall/ecall_safety_effects_finland_summary.pdf
- [257] L. Vlacic, M. Parent, and F. Harashima, editors. *Intelligent Vehicle Technologies – Theory and Applications*. Butterworth-Heinemann, Oxford, UK, 2001. ISBN: 0-7506-5093-1.
- [258] Volvo Car Corporation. New collision warning with auto brake helps prevent rear-end collisions, August 28, 2007. Press release. Available at: <http://www.volvocars.us/footer/about/NewsAndEvents/News/default.htm?item=CACBEE20-779F-4CE7-8026-308B4E699E73>
- [259] T. Watanabe, N. Kishimoto, K. Hayafune, K. Yamada, and N. Maede. Development of an intelligent cruise control system. In *Proceedings of the 2nd World Congress on Intelligent Transport Systems and Services (ITS)*, volume III, pages 1229–1235, Yokohama, Japan, November 9–11, 1995.
- [260] K. van Wees and K. Brookhuis. Product liability for ADAS; legal and human factors perspectives. *European Journal of Transport and Infrastructure Research*, 5(4):357–372, 2005.

- [261] H. Weigel, H. Cramer, G. Wanielik, A. Polychronopoulos, and A. Saroldi. Accurate road geometry estimation for a safe speed application. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 516–521, Tokyo, Japan, June 13–15, 2006.
- [262] G. Widmann, M. Daniels, L. Hamilton, L. Humm, B. Riley, J. Schiffmann, D. Schnelker, and W. Wishon. Comparison of lidar-based and radar-based adaptive cruise control systems. *SAE Technical Paper Series*, 2000-01-0345, 2000.
- [263] M. Williams. PROMETHEUS – the European research programme for optimising the road transport system in Europe. In *IEE Colloquium on Driver Information*, pages 1–9, London, UK, December 1, 1988.
- [264] B.H. Wilson. How soon to brake and how hard to brake: Unified analysis of the envelope of opportunity for rear-end collision warnings. In *Proceedings of the 17th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, Amsterdam, The Netherlands, June 4–6, 2001.
- [265] H. Winner, S. Witte, W. Uhler, and B. Lichtenberg. Adaptive cruise control system aspects and development trends. *SAE Technical Paper Series*, 961010, February 1996.
- [266] J.D. Woll. VORAD collision warning radar. In *Record of the IEEE 1995 International Radar Conference*, pages 369–372, Alexandria, VA, USA, May 8–11, 1995.
- [267] Y. Yamamura, Y. Seto, H. Nishira, and T. Kawabe. An ACC design method for achieving both string stability and ride comfort. In *Proceedings of the 6th International Symposium on Advanced Vehicle Control (AVEC)*, Hiroshima, Japan, September 9–13, 2002. Paper number: 41.
- [268] Y. Yamamura, M. Tabi, M. Kanehira, and T. Murakami. Development of an adaptive cruise control system with stop-and-go capability. *SAE Technical Paper Series*, 2001-01-0798, 2001.
- [269] D. Yanakiev and I. Kanellakopoulos. Nonlinear spacing policies for automated heavy-duty vehicles. *IEEE Transactions on Vehicular Technology*, 47(4):1365–1377, November 1998.
- [270] K. Yi, I. Moon, S. Min, H.J. Yoon, N. Ryu, and K. Huh. Vehicle tests of longitudinal control algorithm for stop and go cruise control. In *Proceedings of the 6th International Symposium on Advanced Vehicle Control (AVEC)*, number 11, Hiroshima, Japan, September 9–13, 2002.
- [271] K. Yi and I.-K. Moon. A driver-adaptive stop-and-go cruise control strategy. In *IEEE International Conference on Networking, Sensing and Control*, volume 1, pages 601–606, March 21–23, 2004.
- [272] K. Yi and Y. Park. An investigation into a string-stable vehicle following control strategy for stop-and-go cruise control. *Proceedings of the Institution of Mechanical Engineers – Part D: Journal of Automobile Engineering*, 216:947–956, 2002.

- [273] K. Yi, M. Woo, S. Kim, and S. Lee. Study on a road-adaptive CW/CA algorithm for automobiles using HiL simulations. *JSME International Journal Series C – Mechanical Systems Machine Elements and Manufacturing*, 42(1):163–170, March 1999.
- [274] P.L. Zador, S.A. Krawchuk, and R.B. Voas. Automotive collision avoidance system (ACAS) program. Final Report DOT HS 809 080, DOT/NHTSA, Washington, DC, USA, August 2000. Available at: http://www-nrd.nhtsa.dot.gov/departments/nrd-12/pubs_rev.html
- [275] D.H. Zhou and P.M. Frank. Fault diagnosis and fault tolerant control. *IEEE Transactions on Aerospace and Electronic Systems*, 34(2):420–427, 1998.
- [276] K. Zhou, J.C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice Hall, Upper Saddle River, NJ, USA, 1996. ISBN: 0-134-56567-3.
- [277] X. Zhu, Y. Huang, and J. Doyle. Soft vs. hard bounds in probabilistic robustness analysis. In *Proceedings of the 35th IEEE Conference on Decision and Control*, volume 3, pages 3412–3417, Kobe, Japan, December 11–13, 1996.
- [278] J. Zuurbier and P. Bremmer. State estimation for integrated vehicle dynamics control. In *Proceedings of the 6th International Symposium on Advanced Vehicle Control (AVEC)*, Hiroshima, Japan, September 9–13, 2002.
- [279] P. Zwaneveld and B. van Arem. Traffic effects of automated vehicle guidance systems. In *Proceedings of the 5th World Congress on Intelligent Transport Systems (ITS)*, Seoul, Korea, October 12–16, 1998.

Glossary

Notation and symbols

This section defines the notation conventions and the symbols that are used throughout this thesis. Unless indicated otherwise, the notation is consistent with SI notation and with standard notation in the fields of control engineering, vehicle dynamics, and randomized algorithms.

Convention of notation

x	Scalar
\mathbf{x}	Vector
x_i	i -th element of the vector \mathbf{x}
\mathbf{x}_j	j -th realization of the vector \mathbf{x}
$(\mathbf{x}_j)_i$	i -th element of the j -th realization of the vector \mathbf{x}
X	Variate
X_i	i -th variate
\mathbf{X}	Matrix
$X_{i,j}$	Matrix element on the i -th row and j -th column
\hat{x}	Estimated value of x
\tilde{x}	<i>A priori</i> estimate of x
x^*	Worst-case value of x
\bar{x}	Average value of x
\dot{x}	Time derivative of x
\ddot{x}	Second time derivative of x
$x(0)$	Initial condition of x
$x(t)$	Variable x as a function of time t
$x(k)$	Variable x at time step k
\mathbf{x}'	Augmented vector of \mathbf{x}
\mathbf{x}^T	Transposed vector of \mathbf{x}
${}^G\mathbf{x}$	Vector \mathbf{x} represented in the coordinate system $\{G\}$
\mathbf{q}°	Parameter vector with negative (cleared) outcome: $J(\mathbf{q}) = 0$
\mathbf{q}^\bullet	Parameter vector with positive (non-cleared) outcome: $J(\mathbf{q}) = 1$
\mathcal{Q}	Set of vectors $\mathbf{q} \in \mathcal{Q}$
\mathcal{Q}°	Set of cleared vectors $\mathbf{q}^\circ \in \mathcal{Q}^\circ \subseteq \mathcal{Q}$
\mathcal{Q}^\bullet	Set of non-cleared vectors $\mathbf{q}^\bullet \in \mathcal{Q}^\bullet \subseteq \mathcal{Q}$
\mathcal{Q}°	Set of vectors $\mathbf{q}^\circ \in \mathcal{Q}^\circ \subseteq \mathcal{Q}$ with unknown outcome

N°	Number of samples with negative (cleared) outcome: $J(\mathbf{q}) = 0$
N^{\bullet}	Number of samples with positive (non-cleared) outcome: $J(\mathbf{q}) = 1$
N^{-}	Minimum number of samples for the sample complexity N
N^{+}	Maximum number of samples for the sample complexity N
$\xi_{\mathcal{Q},2}$	Importance sampling PDF for the second sequence
$N_{\text{Is},2}$	Sample complexity for the second sequence with importance sampling
\mathbb{N}	Set of natural numbers
\mathbb{R}	Set of real numbers
$f(x)$	Function f of x
$F(s)$	Laplace transform of the function $f(t)$
$f * g$	Convolution of functions f and g
$ x $	Absolute value of a signal x
$ H(j\omega) $	Magnitude of the transfer function $H(s)$
$\ x\ _2$	ℓ_2 norm of a vector or signal x
$\ x\ _{\infty}$	\mathcal{H}_{∞} norm of a vector or signal x
$\ x(t)\ _{\text{pow}}$	Weighted ℓ_2 norm or root mean square of a signal $x(t)$
$f^{\mathcal{N}}$	Gaussian probability density function (PDF) f
\hat{f}	Estimated PDF
$\mathbf{q} \sim f_{\mathcal{Q}}$	Vector \mathbf{q} is distributed according to the PDF $f_{\mathcal{Q}}$
$\mathbf{q} \in [\dots]$	Vector \mathbf{q} is within an interval
$\mathbf{q} \in \{\dots\}$	Vector \mathbf{q} is within a set
$\binom{n}{k}$	Binomial coefficient $\frac{n!}{k!(n-k)!}$
$E\{x\}$	Expectation of the value of x
$\text{Pr}\{x\}$	Probability of the event x
$\partial f / \partial x$	Partial derivative of a function f with respect to the variable x

Operators and functions

\arctan_2	Four-quadrant arc tangent operator
\det	Determinant
diag	Diagonal matrix
j	Imaginary constant
\ln	Natural logarithm (base e)
\log	Logarithm (base 10)
mod	Modulo
std	Standard deviation
var	Variance
$\xi_{\mathcal{Q}}$	Importance sampling PDF on the set \mathcal{Q}
$f_{\mathcal{Q}}$	PDF on the set \mathcal{Q}
g	Difference function log-likelihood
\hat{h}	Error propagation function
J	Indicator function
κ	Kernel function
$\mathcal{N}(\mu, \sigma)$	Normal distribution with mean μ and standard deviation σ
$\log\text{-}\mathcal{N}(\mu, \sigma)$	Log-normal distribution with mean μ and standard deviation σ
s	Log-likelihood function

Variables

α	Lateral tire slip angle	rad
β	Body slip angle	rad
γ	Threshold	
Δ	Uncertainty matrix	
δ	Confidence parameter randomized algorithm	
ϵ	Accuracy parameter randomized algorithm	
ϵ_r	Relative accuracy parameter randomized algorithm	
ε	Position accuracy	m
η	GPS longitude	°
ζ	GPS latitude	°
ϑ	GPS heading	°
θ	Pitch angle	rad
\varkappa	Longitudinal tire slip angle	rad
κ	Sequential sampling optimization factor	
λ_{IS}	Importance sampling reduction factor	
λ_v	Gipps driver model velocity ratio	
μ	Mean value	
ρ	Performance measure	
ρ_{air}	Air density	km/m ³
σ	Standard deviation	
ς	Scaling parameter Laplace PDF	rad
τ	Time lag	s
ϕ	Azimuth angle	rad
Φ	Importance sampling parameter vector	
φ	Roll angle	rad
ν	Camber angle	rad
ψ	Yaw angle	rad
ω	Rotational wheel speed	rad/s
ω	Frequency	rad/s
Φ	Orientation vector	
\mathbf{A}	System matrix	
A_x	Frontal area	m ²
A	Actuator	
a	Acceleration	m/s ²
\mathbf{B}	Input matrix	
b	Bias	
\mathbf{C}	Measurement matrix	
C	Constant	
C_w	Aerodynamic drag coefficient	
\mathbf{c}	Change detection vector	
\mathbf{D}	Direct feedthrough matrix	
\mathcal{D}	Disturbance set	
\mathcal{D}	Model uncertainty set	
\mathbf{d}	Disturbance vector	
\mathbf{E}	Disturbance matrix	

E	Entity	
e	Error signal	
e_i	Spacing error vehicle i	m
e_v	Relative velocity error	m/s
e_x	Spacing error	m
\mathbf{F}	Fault matrix	
\mathcal{F}	Fault set	
F	Force	N
\mathbf{f}	Fault vector	
f_{roll}	Rolling resistance coefficient	
G	System	
g	Gravitational acceleration	m/s ²
\mathbf{H}	Kernel density bandwidth matrix	
h	Kernel density bandwidth	
h_i	Height of COG or axle i	m
\mathbf{I}_n	Identity matrix of dimension n	
I	Moment of inertia	kgm ²
\mathbf{K}	Feedback gain matrix	
K	Feedback gain	
k	Time instant	
L	Vehicle length	m
l	Wheelbase	m
\mathbf{M}	Generalized plant	
M	Number of simulation sets	
M_z	Self-aligning moment around z axis	Nm
m	Mass	kg
N	Sample complexity	
n_{eng}	Engine speed	rpm
n	Dimension	
\mathbf{O}	Observability matrix	
O	Object	
\mathbf{P}	Error covariance matrix	
P	Power	W
p	Probability	
Q	Parameter set	
\mathbf{q}	Parameter vector	
\mathbf{R}	Transformation matrix	
R	Radius	m
\mathbf{r}	Residual vector	
r	Range	m
\mathbf{S}	Covariance matrix of multivariate PDF	
S	Distance interval of the scenario	m
S	Sensor	
\mathbf{s}	Position vector	
s	Distance	m
s_j	Actuator position of actuator j	
s_i	Axis width of axle i	m

s_0	Extra safety distance	m
\mathbf{t}	Quaternion	
t	Time	s
t_{delay}	Time delay	s
t_h	Headway time	s
t_{reac}	Reaction time	s
t_{stop}	Stopping time	s
t_{TTC}	Time-to-collision	s
t_{unexp}	Reaction time for unexpected events	s
T	Torque	Nm
T_j	Duration of scenario j	s
\mathbf{u}	Input signal	
\mathbf{V}	Measurement noise covariance matrix	
\mathbf{v}	Measurement noise vector	
v	Velocity	m/s
\mathbf{W}	Process noise covariance matrix	
\mathbf{w}	Process noise vector	
w	Warning level	
\mathbf{x}	State vector	
x	x position	m
\mathbf{y}	Output vector	
y	y position	m
\mathbf{z}	Measurement vector	
z	z position	m

Subscript abbreviations

ACC	Adaptive cruise control
acc	Accelerometer
AIS	Adaptive importance sampling
bin	Binomial bound
br	Brake
CACC	Cooperative adaptive cruise control
CC	Cruise control
cent	Centripetal
Ch	Chernoff bound
chas	Chassis
coll	Collision
comm	Vehicle-to-vehicle communication
diff	Difference
DOE	Design of experiments
drum	Chassis dynamometer drum
eng	Engine
feas	Feasible subset
FN	False negative
FP	False positive
fusion	Estimate of the sensor fusion system

GPS	Global positioning system
grav	Gravitational
GS	Grid search
gyro	Gyroscope
infeas	Infeasible subset
IS	Importance sampling
L	Left
lat	Lateral direction
long	Longitudinal direction
m	Measurement
max	Maximum value
MB	Moving base
min	Minimum value
mult	Multiplicative bound
R	Right
r	Relative value
ref	Reference value
s	Sensor
SS	Simple sampling
st	Steer
sw	Software
th	Throttle
thresh	Threshold
time	Timeliness
TN	True negative
TP	True positive
TTC	Time-to-collision
v	Related to the velocity
VeHIL	Vehicle hardware-in-the-loop
VUT	Vehicle under test
warn	Warning
x	Related to the x direction
y	Related to the y direction














Right subscripts and superscripts

\mathcal{N}	Gaussian
T	Transpose
-1	Inverse
*	Worst-case value
+	Maximum value
0	Nominal value
-	Minimum value
•	'Positive' (non-cleared) outcome ($J = 1$)
◦	'Negative' (cleared) outcome ($J = 0$)
◊	<i>A priori</i> unknown outcome

Left subscripts and superscripts

L	Local Cartesian coordinate frame of the vehicle
G	Global Cartesian coordinate frame of the vehicle
G'	Global Cartesian coordinate frame of the vehicle, aligned with vehicle chassis
P	Polar coordinate frame of the environment sensor
W	Local Cartesian coordinate frame of the wheel

Other symbols

	ADAS-equipped host vehicle
	Target vehicle
	Vehicle center of gravity and origin of local coordinate frame $\{L\}$
	Environment sensor
	Vehicle-to-vehicle communication
	Brake light
	Positive sample
	Element q_i for which some $\mathbf{q}^* \in \mathcal{Q}$ are positive, but other $\mathbf{q}^* \in \mathcal{Q}$ are negative
	Negative sample
	Sample with unknown outcome
	Revolute (rotational) joint
	Prismatic (sliding) joint
	End of definition

Coordinate systems and sign conventions

Figure 9.1 illustrates the sign conventions and coordinate systems that are used in this thesis.

Since the number of vehicles in a scenario may vary, the ADAS-equipped host vehicle is always designated as vehicle i , with n preceding target vehicles subsequently denoted as vehicles $i-1, \dots, i-n$, where the order i depends on how close the target vehicle is to the host in longitudinal direction. In case of multiple preceding target vehicles, the front vehicle is called the lead vehicle in a string of $n+1$ vehicles. The usual situation with two vehicles results in the host being designated as vehicle 2, with the preceding target (and lead) vehicle called vehicle 1. In schematic drawings, vehicles always drive from left to right in the x direction of the global coordinate system, unless indicated otherwise. Furthermore, the host vehicle is always depicted as a dark grey color, whereas the target vehicles are light gray. The vehicle velocity is depicted in thick arrows, whereas vector components are placed in thin lines.

The global position information, such as latitude ζ [$^\circ$], longitude η [$^\circ$], and heading ϑ [$^\circ$] is provided by GPS in the World Geodetic System WGS84. For more convenient use in vehicle control systems, these parameters are transformed (using (3.23)-(3.25)) to a global coordinate frame $\{G\}$, which is a right-handed Cartesian coordinate system (x, y) with its axes x and y aligned with the longitude and latitude directions (pointing north and east respectively), and centered on a local datum (ζ_0, η_0) on the Earth. The height of the vehicle in the z direction is not considered. The orientation of the vehicle is indicated by

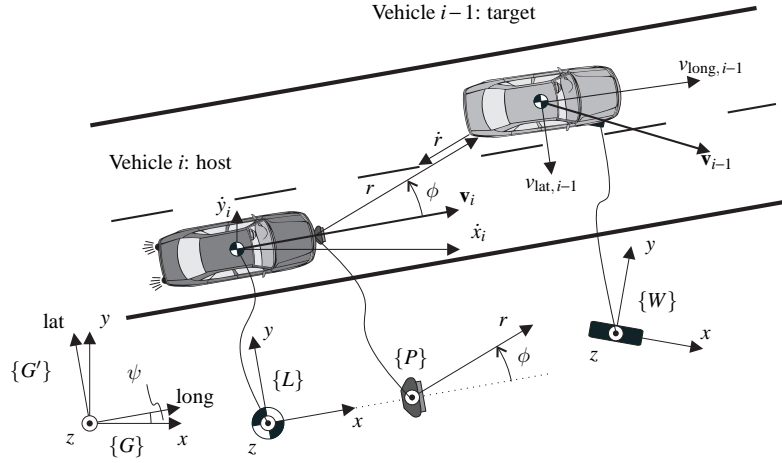


Figure 9.1: Overview of coordinate systems used in this thesis.

the yaw angle ψ . For example, the velocity \mathbf{v}_i of vehicle i is composed of the velocity components ${}^G\dot{x}_i$ and ${}^G\dot{y}_i$.

A more appropriate way to define vehicle velocity is by a global frame $\{G'\}$ with its origin fixed at the origin of frame $\{G\}$, but its direction vectors (denoted as 'lat' and 'long') aligned with the central principal axes of the vehicle chassis. Note that the direction vectors in the frame $\{G'\}$ refer to the *lateral* and *longitudinal* direction of the vehicle, and should not be confused with the terms *latitude* and *longitude* as used in the WGS84 frame. For example, the velocity \mathbf{v}_{i-1} of vehicle $i-1$ is composed of the longitudinal velocity ${}^{G'}v_{\text{long},i-1}$ and the lateral velocity ${}^{G'}v_{\text{lat},i-1}$. The frame $\{G'\}$ also defines the roll motion φ and the pitch motion θ around the longitudinal and lateral vehicle axes, respectively. For ease of notation, the longitudinal velocity ${}^{G'}v_{\text{long}}$ is often simply denoted as v .

The relative motion between two vehicles is measured by an environment sensor in a polar coordinate frame $\{P\}$, usually in the form of range r , range rate \dot{r} and azimuth angle ϕ . Furthermore, an ISO vehicle-fixed local coordinate frame $\{L_i\}$ is defined with its origin in the center of gravity of the vehicle i . This frame is useful for representing the relative motion between vehicles i and $i-1$ from the viewpoint of vehicle i . For example, the relative motion between vehicles 1 and 2 from the viewpoint of vehicle 2 is obtained by a coordinate transformation, where the state ${}^G\mathbf{x}_1$ of vehicle 1 is represented in the *local* coordinate frame $\{L_2\}$ of vehicle 2, *i.e.*, ${}^{L_2}\mathbf{x}_1$. To simplify notation, the relative motion \mathbf{x}_r between vehicles i and $i-1$ is always notated from the viewpoint of the following vehicle, *i.e.*, $\mathbf{x}_{r,i}$. For example, the range ${}^P r_i$ in the polar frame $\{P\}$ can be decomposed into its components ${}^{L_i}x_{r,i}$ and ${}^{L_i}y_{r,i}$ in frame $\{L_i\}$. Since there is no ambiguity in the case of two vehicles, the relative motion is then simply defined as x_r and y_r . Finally, the wheel motion and tire forces are defined in the wheel frame $\{W\}$, with the sign convention according to ISO [177].

The transformations between these frames are performed through pre-multiplication with a transformation matrix and multiplication of the Euler parameters:

$${}^{L_2}\mathbf{s}_1 = {}^{L_2}\mathbf{R}({}^G\mathbf{s}_1 - {}^G\mathbf{s}_2),$$

$${}^{L_2}\mathbf{t}_1 = {}^{L_2}{}^G\mathbf{t}_1,$$

where ${}^{L_2}\mathbf{R}$ is the rotation matrix from frame $\{G\}$ to $\{L_2\}$ and \mathbf{t} represents the orientation in Euler parameters (also known as quaternions).

List of abbreviations

The table below contains an explanation of frequently used abbreviations. They are introduced in every chapter, to keep the reading of a single chapter self-contained. Very common abbreviations, project acronyms, software packages, and names of organizations and companies are not explained. See Figure 1.2 for a list of advanced driver assistance systems not mentioned here.

ACC	Adaptive cruise control
ADAS	Advanced driver assistance system
AIS	Adaptive importance sampling
CACC	Cooperative adaptive cruise control
DPGS	Differential GPS
EKF	Extended Kalman filter
FCW	Forward collision warning
FDI	Fault detection and isolation
FMCW	Frequency-modulated continuous wave
FMECA	Failure modes, effects and criticality analysis
FTA	Fault tree analysis
FTC	Fault-tolerant control
GPS	Global positioning system
HIL	Hardware-in-the-loop
HMI	Human-machine interface
iid	Independent identically distributed
ITS	Intelligent transportation system
MIL	Model-in-the-loop
PCS	Pre-crash system
PDF	Probability density function
PreScan	Pre-crash Scenario analyzer
RMS	Root mean square
SIL	Software-in-the-loop
TTC	Time-to-collision
VeHIL	Vehicle hardware-in-the-loop
VUT	Vehicle under test
VVC	Vehicle-to-vehicle communication

Summary

Design and Validation of Advanced Driver Assistance Systems

Olaf Gietelink

The ever increasing societal cost of road traffic in terms of accessibility, safety, and sustainability, has stimulated the development of advanced driver assistance systems (ADASs). An ADAS is a control system that uses environment sensing to improve driving comfort and traffic safety, by assisting the driver in recognizing and reacting to potentially dangerous traffic situations. Examples of ADASs are adaptive cruise control for automatic car-following; forward collision warning systems that warn the driver of imminent rear-end collisions; and pre-crash systems that improve the effectiveness of safety restraints and subsequently minimize injury severity, by activating them before a collision occurs.

As discussed in Chapter 1, car manufacturers and suppliers of ADASs face several challenges in the development of these systems. First, with the increasing complexity of the system and its environment, the ADAS must satisfy increasingly stringent requirements for performance (*e.g.*, stability, car-following behavior, and comfort) and dependability (*e.g.*, safety, reliability, and fault tolerance). In the second place, during the design phase of an ADAS, fault-tolerant control methods are necessary to maintain dependable performance throughout the parameter set, *i.e.*, the combined set of traffic scenarios, disturbances, and failure modes. Furthermore, an important aspect of the validation phase is to accurately reproduce the conditions under which the control system operates. Finally, the validation phase must be made more efficient, by covering a representative subset of the parameter set with a minimum sufficient number of simulations and experiments. The main objective of this thesis is therefore to develop an efficient model-based methodological framework and associated tools for the design and validation of ADASs, such that the performance and dependability of these systems can be guaranteed.

Validation of the system's dependability, is particularly difficult, because of the increasing complexity of the ADAS, the vehicle, and its traffic environment. To characterize this environment, Chapter 2 presents a microscopic traffic model, based on test data from a field-operational test. In addition, the ADAS is affected by disturbances, such as driver influence and measurement noise. Another safety-critical factor is the occurrence of faults in environment sensors, satellite navigation, and vehicle-to-vehicle communication systems.

To manage these faults, Chapter 3 presents an approach for fault-tolerant state estimation of both the host vehicle state and the inter-vehicle motion. A generalized observer scheme is designed for detection and isolation of faults in vehicle state sensors. Another fault management system handles the fusion of redundant target information that is obtained from an environment sensor and received through vehicle-to-vehicle communication. Fortunately, faults are rare events, but this also makes it hard to validate the fault management system. It is very time-consuming to identify all potential failure modes and it is difficult to reproduce the test conditions and failure modes under which the control system operates.

Chapter 4 therefore presents dedicated tools for design and validation of ADASs. To support the initial design of the ADAS, the simulation environment PreScan is extended with the microscopic traffic model to enable representative simulation of traffic scenarios. In addition, a unique tool for design and validation of ADASs is presented and evaluated:

vehicle hardware-in-the-loop (VeHIL) simulation. The VeHIL laboratory allows an ADAS-equipped vehicle to be tested in an artificial environment, where surrounding traffic is emulated by robot vehicles. In that way, VeHIL can resolve most of the difficulties associated with human-in-the-loop test drives, while still having a relatively high level of reliability due to the connection with real hardware. VeHIL is not meant to replace simulations or test drives, but to form an efficient link between them. Consequently, the number of iteration loops in the development process can be reduced, saving time and costs. However, generation of a representative set of test vectors is crucial to an efficient and effective validation program using PreScan, VeHIL, and test drives.

Chapter 5 therefore combines the use of these three tools (PreScan, VeHIL, and test drives) in a methodological framework for design and validation of ADASs, based on randomized algorithms. This probabilistic approach cannot prove that the system has adequate performance and dependability. However, when we accept a (small) risk of failure, a randomized algorithm is able to obtain an efficient estimate of the performance and dependability of the system. An important characteristic of these algorithms is the number of simulations and experiments that is required to determine whether the ADAS meets its requirements. The conventional Chernoff bound on this sample complexity has been reduced to a significantly lower bound that provides a sufficient number of samples through a sequential approach to the validation problem. In addition, *a priori* information on the system behavior can be included in the test schedule, such that the most relevant subsets in the parameter set are investigated. Furthermore, an adaptive importance sampling algorithm is developed that enables to use a more efficient representation of this subset.

A major advantage of this methodology is that the number of required PreScan simulations and VeHIL experiments can already be predicted after a limited trend study. This enables the *a priori* allocation of the appropriate time and resources that are required for the validation of an ADAS. This makes the methodology considerably more efficient than conventional simulation techniques and the current practice of trial-and-error test drives. It results in a test schedule definition with a minimum number of simulations and test runs, such that the performance and dependability of an ADAS can be guaranteed, given a desired level of accuracy and confidence. Within this framework, the purpose of PreScan is to provide a preliminary validation of the ADAS using the adaptive importance sampling algorithm. The role of VeHIL is to validate the simulation model, as well as refine the simulation results. Finally, test drives are used to confirm the PreScan and VeHIL results. The added value of the methodology is demonstrated with three case studies.

Chapter 6 provides the validation of a driver information and warning system for safe speed and safe distance. VeHIL experiments show that the warning and intervention algorithms should be fine-tuned to further improve the dependability of the system. Chapter 7 has presented a control algorithm for cooperative adaptive cruise control that can achieve a string-stable following behavior by using environment sensing and vehicle-to-vehicle communication. It is further shown that feedback of the acceleration of preceding vehicles enhances the comfort, stability, and safety of the longitudinal control function. In Chapter 8 a pre-crash system is tested in non-destructive and representative pre-crash scenarios. For this purpose the VeHIL concept has been modified, such that a vehicle equipped with a pre-crash system can be approached by a target vehicle very closely. This allows to test the performance and dependability of reversible passive and active safety restraints. MADYMO simulations and VeHIL experiments have demonstrated the safety benefit of a pre-crash system, consisting of both a belt pretensioner and a brake assist system.

Samenvatting

Ontwerp en Validatie van Rijtaakondersteunende Systemen

Olaf Gietelink

De alsmaar toenemende verkeersproblematiek omtrent bereikbaarheid, veiligheid en duurzaamheid heeft de ontwikkeling gestimuleerd van rijtaakondersteunende systemen (een 'advanced driver assistance system' oftewel ADAS). Een dergelijk regelsysteem heeft als doel het rijcomfort en de verkeersveiligheid te verbeteren door middel van het herkennen van en reageren op potentieel gevaarlijke verkeerssituaties met behulp van omgevingssensoren. Voorbeelden zijn adaptieve snelheidsregeling voor automatisch afstand houden; systemen die de bestuurder waarschuwen bij een dreigende botsing, of passieve veiligheidssystemen die reeds geactiveerd worden vlak voordat een botsing plaats vindt.

Zoals besproken wordt in Hoofdstuk 1, staan autofabrikanten en leveranciers van ADAS voor een aantal uitdagingen in de ontwikkeling van deze systemen. Ten eerste moet een ADAS, vanwege de toenemende complexiteit van het systeem en zijn omgeving, voldoen aan strenge eisen op het gebied van stabiliteit, volgedrag, betrouwbaarheid, veiligheid en fout-tolerantie. Ten tweede, tijdens het ontwerp van een ADAS zijn fout-tolerante regeltechnieken nodig om betrouwbare prestaties te garanderen voor alle mogelijke combinaties van verkeersscenario's, verstoringen en faalmogelijkheden van het systeem (de parameter set). Bovendien is het belangrijk om tijdens de validatie zo natuurgetrouw mogelijk de omstandigheden te reproduceren waaronder het ADAS moet functioneren. Ten slotte moet het validatieproces efficiënter gemaakt worden door een representatieve subset van de parameter set af te dekken met een minimaal vereiste hoeveelheid simulaties en experimenten.

Validatie van de systeembetrouwbaarheid is vooral moeilijk vanwege de toegenomen complexiteit van het ADAS, het voertuig en de verkeersomgeving. Om deze omgeving te karakteriseren, beschrijft Hoofdstuk 2 een microscopisch verkeersmodel, gebaseerd op testdata van een praktijkproef. Verder is een ADAS onderhevig aan verstoringen, zoals bestuurdersinvloeden en meetruis. Een andere veiligheidskritische factor is het optreden van fouten in omgevingssensoren, satellietnavigatie, en voertuig-voertuig-communicatie.

Om deze fouten te beheersen, behandelt Hoofdstuk 3 een methode voor fout-tolerante toestandsschatting voor zowel de beweging van het ego-voertuig, als tussen voertuigen onderling. Een gegeneraliseerd stelsel van toestandsschatters wordt ontworpen dat fouten in voertuigsensoren kan detecteren en lokaliseren. Een tweede fout-tolerant systeem is verantwoordelijk voor de fusie van informatie die binnenkomt via de omgevingssensor en voertuig-voertuig-communicatie. Gelukkig komen fouten zelden voor, maar dit maakt het ook moeilijker om deze fout-tolerante systemen te valideren. Het kost namelijk veel tijd om alle mogelijkheden voor systeemfalen op te sporen en het is moeilijk om de exacte omstandigheden van de fouten te reproduceren waarmee de ADAS-regelaar te maken krijgt.

Hoofdstuk 4 introduceert daarom specifieke werktuigen voor het ontwerp en validatie van ADAS. Ten behoeve van waarheidsgetrouwe verkeerssimulatie van een ADAS wordt de simulatieomgeving PreScan uitgebreid met het microscopisch verkeersmodel. Bovendien wordt een uniek werktuig voor het ontwerp en validatie van ADAS geïntroduceerd en geëvalueerd: 'vehicle hardware-in-the-loop' (VeHIL) simulatie. Het VeHIL laboratorium voorziet in het testen van een met ADAS uitgerust voertuig in een kunstmatige omgeving, waarbij andere verkeersdeelnemers worden nagebootst door robotvoertuigen. Op die manier

slaagt VeHIL er in om de gebruikelijke problemen van testritten met menselijke bestuurders op te lossen, terwijl de betrouwbaarheid toch gegarandeerd wordt door het gebruik van een echt voertuig. VeHIL is daarom niet bedoeld ter vervanging van simulaties of testritten, maar juist om deze werktuigen efficiënt op elkaar aan te laten sluiten. Daardoor kunnen de doorlooptijd en de ontwikkelingskosten in het ontwikkelingsproces worden verminderd. Echter, het definiëren van een representatieve set van testvectoren is cruciaal voor een efficiënt en effectief testprogramma dat gebruik maakt van PreScan, VeHIL en testritten.

Hoofdstuk 5 combineert daarom deze drie werktuigen (PreScan, VeHIL en testritten) in een methodologisch raamwerk voor het ontwerp en validatie van ADAS, gebaseerd op toevals-algoritmen. Deze probabilistische aanpak kan niet bewijzen dat een systeem voldoet aan de eisen voor prestaties en betrouwbaarheid. Wanneer we echter een (kleine) faalkans accepteren, is een toevals-algoritme wel in staat om een efficiënte schatting van de prestaties en betrouwbaarheid van het systeem te bepalen. Een belangrijke eigenschap van deze algoritmes is het aantal simulaties en testen dat nodig is om te bepalen of het ADAS voldoet aan de gestelde eisen. Door middel van een sequentiële aanpak van het probleem wordt de conventionele Chernoff limiet van deze steekproefgrootte gereduceerd tot een significant lagere waarde. Daarnaast kan informatie over het systeemgedrag van te voren in het testprogramma worden opgenomen, zodat de meest relevante subsets uit de parameter set onderzocht worden. Bovendien wordt een algoritme voor een adaptief gewogen steekproef ontwikkeld dat een efficiëntere weergave van deze subset gebruikt.

Een belangrijk voordeel van deze methodologie is dat het aantal vereiste PreScan simulaties en VeHIL testen reeds voorspeld kan worden na een beperkte trendstudie. Dit maakt het mogelijk om van te voren de benodigde testtijd en middelen te bepalen die vereist zijn voor de validatie van een ADAS. Dit maakt de methodologie aanzienlijk efficiënter dan conventionele simulatietechnieken en proefondervindelijk testen. Dit resulteert in een testprogramma met een minimaal aantal simulaties en testen waarmee de prestatie en betrouwbaarheid kan worden aangetoond, gegeven een gewenste nauwkeurigheid en vertrouwen. Binnen dit raamwerk is het doel van PreScan om een voorlopige validatie te geven met behulp van de adaptief gewogen steekproef. De rol van VeHIL is om het simulatiemodel te valideren, evenals de simulatieresultaten te verfijnen. Ten slotte worden testritten gebruikt om de PreScan en VeHIL resultaten te bevestigen. De toegevoegde waarde van de methodiek wordt aanschouwelijk gemaakt met behulp van drie praktijkgevallen.

Hoofdstuk 6 behandelt de validatie van een waarschuwings- en informatiesysteem dat een bestuurder een veilige afstand en veilige snelheid adviseert. VeHIL experimenten tonen aan dat de waarschuwingalgoritmes van dit systeem moeten worden verfijnd om de betrouwbaarheid verder te verbeteren. In Hoofdstuk 7 wordt een regelalgoritme voor coöperatieve adaptieve snelheidsregeling ontwikkeld dat een stabiel volgedrag tot stand brengt met behulp van omgevingsdetectie en voertuig-voertuig communicatie. Er wordt verder aangetoond dat terugkoppeling van het acceleratiesignaal van voorliggers het comfort, de stabiliteit en de veiligheid van de longitudinale voertuigregeling verbetert. In Hoofdstuk 8 wordt een actief veiligheidssysteem getest dat reeds vóór een botsing reageert. Voor dit doel is het VeHIL concept aangepast, zodat een voertuig dat is uitgerust met een dergelijk systeem zeer dicht genaderd kan worden met een robotvoertuig in niet-destructieve maar wel representatieve ongevalsscenario's. Hierdoor kan de werking en betrouwbaarheid van omkeerbare passieve en actieve veiligheidssystemen worden getest. MADYMO simulaties en VeHIL experimenten hebben laten zien dat een systeem dat gebruik maakt van een gordelspanner en een actieve remhulp de veiligheid van inzittenden kan verhogen.

About the author

Olaf Gietelink was born on the 9th of August 1974 in Amsterdam, The Netherlands. He received his secondary education (VWO) at the Stedelijk Gymnasium in Haarlem in 1992. In the academic year 1992-1993 he studied at Oxford Brookes University, where he received his propaedeutic diploma in Economics and Computer Science. From 1993 to 2000 he studied Mechanical Engineering at Delft University of Technology, where he graduated cum laude at the departments of Transport Technology and Mechatronics. His graduation project was carried out at the Industrieanlagen-Betriebsgesellschaft in Lathen, Germany, where he completed his Masters thesis titled "Spectral Vibration Analysis and Ride Comfort Evaluation of the Magnetic Levitated Train Transrapid 08". After his graduation he joined the Advanced Transport Systems department of TNO Automotive in Delft, The Netherlands (currently the Integrated Safety department of TNO Science and Industry in Helmond, The Netherlands), where he worked as a Development Engineer in several projects on intelligent transportation systems and vehicle mechatronics. In 2003 he started a Ph.D. research project, titled "Validation Methodology for Fault-tolerant Advanced Driver Assistance Systems" in cooperation with the Delft Center for Systems and Control of Delft University of Technology. Within the framework of his Ph.D. research he stayed at the California PATH program based at the University of California at Berkeley, as a visiting scholar during the summer of 2004. His research interests are advanced driver assistance systems, control engineering, fault management, and other topics related to intelligent transportation systems. He has published over 20 conference and journal papers in these research fields.

TRAIL Thesis Series

A series of The Netherlands TRAIL Research School for theses on transport, infrastructure and logistics.

Nat, C.G.J.M. van der, *A Knowledge-based Concept Exploration Model for Submarine Design*, T99/1, March 1999, TRAIL Thesis Series, Delft University Press, The Netherlands

Westrenen, F.C. van, *The Maritime Pilot at Work: Evaluation and Use of a Time-to-boundary Model of Mental Workload in Human-machine Systems*, T99/2, May 1999, TRAIL Thesis Series, Eburon, The Netherlands

Veenstra, A.W., *Quantitative Analysis of Shipping Markets*, T99/3, April 1999, TRAIL Thesis Series, Delft University Press, The Netherlands

Minderhoud, M.M., *Supported Driving: Impacts on Motorway Traffic Flow*, T99/4, July 1999, TRAIL Thesis Series, Delft University Press, The Netherlands

Hoogendoorn, S.P., *Multiclass Continuum Modelling of Multilane Traffic Flow*, T99/5, September 1999, TRAIL Thesis Series, Delft University Press, The Netherlands

Hoedemaeker, M., *Driving with Intelligent Vehicles: Driving Behaviour with Adaptive Cruise Control and the Acceptance by Individual Drivers*, T99/6, November 1999, TRAIL Thesis Series, Delft University Press, The Netherlands

Marchau, V.A.W.J., *Technology Assessment of Automated Vehicle Guidance - Prospects for Automated Driving Implementation*, T2000/1, January 2000, TRAIL Thesis Series, Delft University Press, The Netherlands

Subiono, *On Classes of Min-max-plus Systems and their Applications*, T2000/2, June 2000, TRAIL Thesis Series, Delft University Press, The Netherlands

Meer, J.R. van, *Operational Control of Internal Transport*, T2000/5, September 2000, TRAIL Thesis Series, Delft University Press, The Netherlands

Bliemer, M.C.J., *Analytical Dynamic Traffic Assignment with Interacting User-Classes: Theoretical Advances and Applications using a Variational Inequality Approach*, T2001/1, January 2001, TRAIL Thesis Series, Delft University Press, The Netherlands

Muileman, G.J., *Time-based logistics: An analysis of the relevance, causes and impacts*, T2001/2, April 2001, TRAIL Thesis Series, Delft University Press, The Netherlands

Roodbergen, K.J., *Layout and Routing Methods for Warehouses*, T2001/3, May 2001, TRAIL Thesis Series, The Netherlands

Willems, J.K.C.A.S., *Bundeling van infrastructuur, theoretische en praktische waarde van een ruimtelijk inrichtingsconcept*, T2001/4, June 2001, TRAIL Thesis Series, Delft University Press, The Netherlands

Binsbergen, A.J. van, J.G.S.N. Visser, *Innovation Steps towards Efficient Goods Distribution Systems for Urban Areas*, T2001/5, May 2001, TRAIL Thesis Series, Delft University Press, The Netherlands

Rosmuller, N., *Safety analysis of Transport Corridors*, T2001/6, June 2001, TRAIL Thesis Series, Delft University Press, The Netherlands

Schaafsma, A., *Dynamisch Railverkeersmanagement, besturingsconcept voor railverkeer op basis van het Lagenmodel Verkeer en Vervoer*, T2001/7, October 2001, TRAIL Thesis Series, Delft University Press, The Netherlands

Bockstael-Blok, W., *Chains and Networks in Multimodal Passenger Transport. Exploring a design approach*, T2001/8, December 2001, TRAIL Thesis Series, Delft University Press, The Netherlands

Wolters, M.J.J., *The Business of Modularity and the Modularity of Business*, T2002/1, February 2002, TRAIL Thesis Series, The Netherlands

- Vis, F.A., *Planning and Control Concepts for Material Handling Systems*, T2002/2, May 2002, TRAIL Thesis Series, The Netherlands
- Koppius, O.R., *Information Architecture and Electronic Market Performance*, T2002/3, May 2002, TRAIL Thesis Series, The Netherlands
- Veeneman, W.W., *Mind the Gap; Bridging Theories and Practice for the Organisation of Metropolitan Public Transport*, T2002/4, June 2002, TRAIL Thesis Series, Delft University Press, The Netherlands
- Nes, R. van, *Design of multimodal transport networks, a hierarchical approach*, T2002/5, September 2002, TRAIL Thesis Series, Delft University Press, The Netherlands
- Pol, P.M.J., *A Renaissance of Stations, Railways and Cities, Economic Effects, Development Strategies and Organisational Issues of European High-Speed-Train Stations*, T2002/6, October 2002, TRAIL Thesis Series, Delft University Press, The Netherlands
- Runhaar, H., *Freight transport: at any price? Effects of transport costs on book and newspaper supply chains in the Netherlands*, T2002/7, December 2002, TRAIL Thesis Series, Delft University Press, The Netherlands
- Spek, S.C. van der, *Connectors. The Way beyond Transferring*, T2003/1, February 2003, TRAIL Thesis Series, Delft University Press, The Netherlands
- Lindeijer, D.G., *Controlling Automated Traffic Agents*, T2003/2, February 2003, TRAIL Thesis Series, Eburon, The Netherlands
- Riet, O.A.W.T. van de, *Policy Analysis in Multi-Actor Policy Settings. Navigating Between Negotiated Nonsense and Useless Knowledge*, T2003/3, March 2003, TRAIL Thesis Series, Eburon, The Netherlands
- Reeven, P.A. van, *Competition in Scheduled Transport*, T2003/4, April 2003, TRAIL Thesis Series, Eburon, The Netherlands
- Peeters, L.W.P., *Cyclic Railway Timetable Optimization*, T2003/5, June 2003, TRAIL Thesis Series, The Netherlands
- Soto y Koelemeijer, G., *On the behaviour of classes of min-max-plus systems*, T2003/6, September 2003, TRAIL Thesis Series, The Netherlands
- Lindveld, Ch.D.R., *Dynamic O-D matrix estimation: a behavioural approach*, T2003/7, September 2003, TRAIL Thesis Series, Eburon, The Netherlands
- Weerd, M.M. de, *Plan Merging in Multi-Agent Systems*, T2003/8, December 2003, TRAIL Thesis Series, The Netherlands
- Langen, P.W. de, *The Performance of Seaport Clusters*, T2004/1, January 2004, TRAIL Thesis Series, The Netherlands
- Hegy, A., *Model Predictive Control for Integrating Traffic Control Measures*, T2004/2, February 2004, TRAIL Thesis Series, The Netherlands
- Lint, J.W.C. van, *Reliable Travel Time Prediction for Freeways*, T2004/3, June 2004, TRAIL Thesis Series, The Netherlands
- Tabibi, M., *Design and Control of Automated Truck Traffic at Motorway Ramps*, T2004/4, July 2004, TRAIL Thesis Series, The Netherlands
- Verduijn, T. M., *Dynamism in Supply Networks: Actor switching in a turbulent business environment*, T2004/5, September 2004, TRAIL Thesis Series, The Netherlands
- Daamen, W., *Modelling Passenger Flows in Public Transport Facilities*, T2004/6, September 2004, TRAIL Thesis Series, The Netherlands
- Zoeteman, A., *Railway Design and Maintenance from a Life-Cycle Cost Perspective: A Decision-Support Approach*, T2004/7, November 2004, TRAIL Thesis Series, The Netherlands

- Bos, D.M., *Changing Seats: A Behavioural Analysis of P&R Use*, T2004/8, November 2004, TRAIL Thesis Series, The Netherlands
- Versteegt, C., *Holonic Control For Large Scale Automated Logistic Systems*, T2004/9, December 2004, TRAIL Thesis Series, The Netherlands
- Wees, K.A.P.C. van, *Intelligente voertuigen, veiligheidsregulering en aansprakelijkheid. Een onderzoek naar juridische aspecten van Advanced Driver Assistance Systems in het wegverkeer*, T2004/10, December 2004, TRAIL Thesis Series, The Netherlands
- Tampère, C.M.J., *Human-Kinetic Multiclass Traffic Flow Theory and Modelling: With Application to Advanced Driver Assistance Systems in Congestion*, T2004/11, December 2004, TRAIL Thesis Series, The Netherlands
- Rooij, R.M., *The Mobile City. The planning and design of the Network City from a mobility point of view*, T2005/1, February 2005, TRAIL Thesis Series, The Netherlands
- Le-Anh, T., *Intelligent Control of Vehicle-Based Internal Transport Systems*, T2005/2, April 2005, TRAIL Thesis Series, The Netherlands
- Zuidgeest, M.H.P., *Sustainable Urban Transport Development: a Dynamic Optimization Approach*, T2005/3, April 2005, TRAIL Thesis Series, The Netherlands
- Hoogendoorn-Lanser, S., *Modelling Travel Behaviour in Multimodal Networks*, T2005/4, May 2005, TRAIL Thesis Series, The Netherlands
- Dekker, S., *Port Investment - Towards an integrated planning of port capacity*, T2005/5, June 2005, TRAIL Thesis Series, The Netherlands
- Koolstra, K., *Transport Infrastructure Slot Allocation*, T2005/6, June 2005, TRAIL Thesis Series, The Netherlands
- Vromans, M., *Reliability of Railway Systems*, T2005/7, July 2005, TRAIL Thesis Series, The Netherlands
- Oosten, W., *Ruimte voor een democratische rechtsstaat. Geschakelde sturing bij ruimtelijke investeringen*, T2005/8, September 2005, TRAIL Thesis Series, Sociotext, The Netherlands
- Le-Duc, T., *Design and control of efficient order picking*, T2005/9, September 2005, TRAIL Thesis Series, The Netherlands
- Goverde, R., *Punctuality of Railway Operations and Timetable Stability Analysis*, T2005/10, October 2005, TRAIL Thesis Series, The Netherlands
- Kager, R.M., *Design and implementation of a method for the synthesis of travel diary data*, T2005/11, October 2005, TRAIL Thesis Series, The Netherlands
- Boer, C., *Distributed Simulation in Industry*, T2005/12, October 2005, TRAIL Thesis Series, The Netherlands
- Pielage, B.A., *Conceptual Design of Automated Freight Transport Systems*, T2005/14, November 2005, TRAIL Thesis Series, The Netherlands
- Groothedde, B., *Collaborative Logistics and Transportation Networks, a modeling approach to network design*, T2005/15, November 2005, TRAIL Thesis Series, The Netherlands
- Valk, J.M., *Coordination among Autonomous Planners*, T2005/16, December 2005, TRAIL Thesis Series, The Netherlands
- Krogt, R.P.J. van der, *Plan Repair in Single-Agent and Multi-Agent Systems*, T2005/17, December 2005, TRAIL Thesis Series, The Netherlands
- Bontekoning, Y.M., *Hub exchange operations in intermodal hub-and-spoke networks. A performance comparison of four types of rail-rail exchange facilities*, T2006/1, February 2006, TRAIL Thesis Series, The Netherlands

- Lentink, R., *Algorithmic Decision Support for Shunt Planning*, T2006/2, February 2006, TRAIL Thesis Series, The Netherlands
- Ngoduy, D., *Macroscopic Discontinuity Modeling for Multiclass Multilane Traffic Flow Operations*, T2006/3, April 2006, TRAIL Thesis Series, The Netherlands
- Vanderschuren, M.J.W.A., *Intelligent Transport Systems for South Africa. Impact assessment through microscopic simulation in the South African context*, T2006/4, August 2006, TRAIL Thesis Series, The Netherlands
- Ongkittikul, S., *Innovation and Regulatory Reform in Public Transport*, T2006/5, September 2006, TRAIL Thesis Series, The Netherlands
- Yuan, J., *Stochastic Modelling of Train Delays and Delay Propagation in Stations*, T2006/6, October 2006, TRAIL Thesis Series, The Netherlands
- Viti, F., *The Dynamics and the Uncertainty of Delays at Signals*, T2006/7, November 2006, TRAIL Thesis Series, The Netherlands
- Huisken, G., *Inter-Urban Short-Term Traffic Congestion Prediction*, T2006/8, December 2006, TRAIL Thesis Series, The Netherlands
- Feijter, R. de, *Controlling High Speed Automated Transport Network Operations*, T2006/9, December 2006, TRAIL Thesis Series, The Netherlands
- Makoriwa, C., *Performance of Traffic Networks. A mosaic of measures*, T2006/10, December 2006, TRAIL Thesis Series, The Netherlands
- Miska, M., *Microscopic Online Simulation for Real time Traffic Management*, T2007/1, January 2007, TRAIL Thesis Series, The Netherlands
- Chorus, C., *Traveler Response to Information*, T2007/2, February 2007, TRAIL Thesis Series, The Netherlands
- Weijermars, W.A.M., *Analysis of Urban Traffic Patterns Using Clustering*, T2007/3, April 2007, TRAIL Thesis Series, The Netherlands
- Zondag, B., *Joined Modeling of Land-use, Transport and Economy*, T2007/4, April 2007, TRAIL Thesis Series, The Netherlands
- Bok, M.A. de, *Infrastructure and Firm Dynamics: A micro-simulation approach*, T2007/5, May 2007, TRAIL Thesis Series, The Netherlands
- Fiorenzo-Catalano, M.S., *Choice Set Generation in Multi-Modal Transportation Networks*, T2007/6, June 2007, TRAIL Thesis Series, The Netherlands
- Vlist, P. van der, *Synchronizing the retail supply chain*, T2007/7, June 2007, TRAIL Thesis Series, The Netherlands
- Joksimovic, D., *Dynamic Bi-Level Optimal Toll Design Approach for Dynamic Traffic Networks*, T2007/8, September 2007, TRAIL Thesis Series, The Netherlands
- Warffemius, P.M.J., *Modeling the Clustering of Distribution Centers around Amsterdam Airport Schiphol. Location Endowments, Economies of Agglomeration, Locked-in Logistics and Policy Implications*, T2007/9, September 2007, TRAIL Thesis Series, The Netherlands
- Driel, C.J.G. van, *Driver Support in Congestion: an assessment of user needs and impacts on driver and traffic flow*, T2007/10, November 2007, TRAIL Thesis Series, The Netherlands
- Gietelink, O.J., *Design and Validation of Advanced Driver Assistance Systems*, T2007/11, November 2007, TRAIL Thesis Series, The Netherlands